

Towards Semantic Social Networks

Ronald Chenu-Abente y Fausto Giunchiglia
 Dept. of Information Engineering and Computer Science
 University of Trento
 Trento, Italy
 Email: chenu@disi.unitn.it, fausto@disi.unitn.it

Luca Cernuzzi
 Departamento Electrónica e Informática
 Universidad Católica “Nuestra Señora de la Asunción”
 Asunción, Paraguay
 Email: lcernuzz@uca.edu.py

Abstract—Computer-enabled social services like tagging or sharing are ubiquitous in current web applications that are aimed to a group of users. These services do not only add value and new functionalities to their applications but also create a network of users and services that interconnect them to a wider on-line ecosystem. Currently these social networks mainly use the vast amount of user-created content, and activity logs to apply to provide recommendations and more complex services. This paper presents the Social Core a social network engine that implements semantic-based functionalities like semantic annotations, semantic search semantic-enhanced access control and user privacy protection. The Social Core was integrated as part of the SmartCampus mobile platform, which was tested by around one hundred students, and it is currently being further developed as part of European FP7 project SmartSociety.

Index Terms—Semantic Web, Social Networks, Semantic Search, Semantic Access Control, Linked Data, Knowledge Bases, Data Privacy and Security.

I. INTRODUCCIÓN

Las redes sociales online son servicios basados en Internet que la gente usa para manejar sus relaciones sociales, para compartir el contenido que ellos producen y para tener acceso al contenido producido por otros. Según [4], las redes sociales están definidas por tener: i) perfiles públicos o semi-públicos para sus usuarios, ii) una lista de otros usuarios con los cuales se comparte una conexión, y iii) una lista de conexiones hechas por otros dentro del sistema. En adición a esto la mayoría de las redes sociales también ofrecen mecanismos para mensajes, tags y/o comentarios (aunque con diversos nombres y presentaciones según el sitio específico). Más allá de estas funcionalidades comunes, cada red social también ofrece sus funcionalidades específicas como blogging, métodos para compartir o re-compartir contenido, creación y/o edición de documentos y otros contenidos. Los resultados en [6], [22], [24], [27] van más allá sugiriendo que las redes sociales actuales son exitosas porque ofrecen los siguientes servicios que gratifican a sus usuarios: conexión social, identidades compartidas, acceso a contenido, actualización de estados, búsqueda/navegación social, e investigación social. Varias redes sociales de propósito general tales como Facebook¹ y Twitter² han sido tremendamente exitosas en atraer usuarios y mantener niveles de actividad/interacción considerables. Redes

sociales más enfocadas en los negocios y la academia [7] han sido también propuestas aunque ellas no son generalmente consideradas tan exitosas como las anteriores [25] y todavía enfrentan varios desafíos para su adopción y uso. Al mismo tiempo el uso de semántica ha sido exitosamente implementada en varias plataformas orientadas a datos offline (como por ejemplo, en sistemas de procesamiento de lenguaje natural y Bigdata); mejorando así varios aspectos relacionados con la representación de conocimiento, clasificación, matching y búsqueda. El principal objetivo de este trabajo es el mejoramiento de los motores de redes sociales mediante el uso de enfoques semánticos, de manera a introducir servicios tales como anotaciones semánticas, búsqueda semántica social y control de acceso mejorado por el uso de semántica. El resultante motor de redes sociales, llamado Social Core, y sus servicios permitirán la exploración de mejoras a las redes sociales actuales y ayudarán a disminuir algunas de las limitaciones de las mismas. La idea clave de este trabajo es el empezar desde entidades que representan los metadatos y la información semántica de todos los objetos en la red social, y luego enriquecer algunas de estas entidades para permitirle el convertirse en sujetos de las interacciones sociales. El permitir que estas entidades semánticas se comporten tanto como objetos y sujetos es lo que nos permite el ofrecer estos servicios sociales semánticos. El trabajo está estructurado como sigue; la Sección II introducirá y detallará los sub-sistemas principales de Social Core, mientras que la Sección III describirá los servicios principales que Social Core ofrece. La Sección IV introducirá los casos de usos principales y como Social Core contribuye, junto con información sobre uso y testeo del sistema. La Sección V analiza los trabajos relacionados y finalmente la Sección VI enumerará como Social Core enfrenta los objetivos propuestos y examinará conclusiones y trabajo futuro.

II. SOCIAL CORE

Los principales objetivos de Social Core son el proveer almacenamiento de metadatos, manejo de usuario, control de acceso y servicios semánticos para así asistir a la creación de una red social semántica. La Figura 1 muestra una visión arquitectural de alto nivel para los componentes internos de Social Core.

En la Figura 1 los tres elementos principales son el Almacenamiento Semántico (Semantic Storage), donde los conceptos

¹<http://www.facebook.com/>

²<http://twitter.com/>

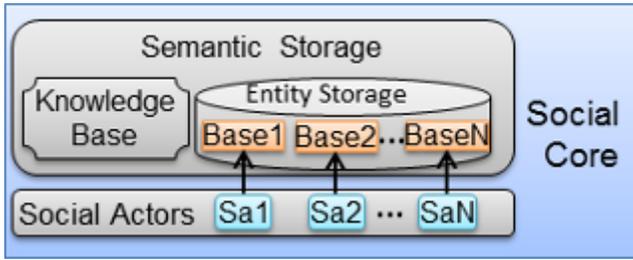


Fig. 1. Arquitectura interna de Social Core.

y las entidades son definidos (subsección II-A) y Actores Sociales (Social Actors), que actúan como sujetos en las interacciones (subsección II-B).

A. Entidades, Tipos de Entidades y Bases de Entidades

Para describir y representar los datos del usuario (que se encuentran potencialmente compartidos en la red), Social Core define un conjunto de tipos de entidad comúnmente usadas tales como personas, lugares, eventos, medios y otros. Cada tipo de entidad (entity type) provee una definición estructurada de los metadatos y conceptos usados para describir a estas entidades (por ejemplo las personas tienen 'nombre' y 'edad'; los lugares tienen 'coordenadas'; etc). A grandes rasgos este enfoque es similar al presentado por la iniciativa schema.org³, mientras el trabajo en [18] provee una descripción detallada de las ideas y los algoritmos involucrados. Sin embargo, a diferencia de schema.org (que intenta proveer un estándar para las anotaciones en una página web), los tipos de entidad de Social Core están mejor adaptados para ser usados en un contexto social y son más flexibles en el sentido que los usuarios son capaces de agregar metadatos personalizados a las entidades instanciadas. Los conceptos usados para describir las entidades están basados en una ontología que se encuentra en la base de conocimientos (knowledge base), mientras que los metadatos para la entidad están contenidos en el almacenamiento de entidades (entity store); ambos presentes en la Figura 1. En particular los metadatos de las entidades son representados por En tal que:

$$En = \langle id; type; Attr; Rel \rangle \quad (1)$$

En donde:

- id : es un identificador único (por ejemplo un URI);
- $type$: es el tipo de la entidad, es decir, la categoría a la cual pertenece (por ejemplo, la entidad 'John' es del tipo persona);
- $Attr$: es un conjunto de atributos compuestos de pares $attr = \langle attr_{name}; attr_{value} \rangle$ que describen las propiedades de esa entidad particular (por ejemplo, la fecha de nacimiento de John es 02/01/88);
- Rel : es un conjunto de atributos relaciones compuestos de pares $rel = \langle rel_{name}; rel_{value} \rangle$ que describen las relaciones de esa entidad en particular con otras entidades (por ejemplo, John es un 'amigoDe' Paul);

³<http://schema.org/>

Por ejemplo, el tipo de entidad 'documento' define que todas las instancias de esta entidad usada para representar documentos de texto tendrán el atributo 'título' y la relación 'autor'. Las verdaderas entidades que instancian los tipos de entidades son almacenadas en diferentes bases de entidades (Entity Bases), representadas en la Figura 1 como Base1 a BaseN, dentro del almacenamiento de entidades; estas bases de entidades son usadas como contenedores separados para las entidades que pertenecen a usuarios particulares o, hablando más generalmente, actores sociales (social actors) como se definen en la siguiente subsección. A pesar que la especificación completa de estas estructuras va más allá del alcance de este trabajo, es importante recalcar que estas entidades, tipos de entidades y bases de entidades son todas usadas para representar los metadatos y conceptos almacenados en la plataforma.

B. Actores Sociales: Entidades como Sujetos

Además de la representación y manejo de contenido (es decir, objetos) las redes sociales también necesitan definir los sujetos que son capaces de crear, modificar y compartir este contenido. Para este objetivo introducimos la estructura actor social (Social Actor) que se define sobre, es decir como una extensión, de las entidades (como se muestra en la Figura 2). El propósito de la separación entre entidad y actor social tiene dos justificaciones: la principal y más inmediata razón es que más de una estructura de actor social podría ser definida para la misma entidad (como también se muestra en la Figura 3); la otra razón es que el tipo de información en la entidad se refiere a propiedades en el mundo real (por ejemplo, nombre y altura), mientras que la información contenida en la estructura actor social se refiere a las propiedades y actividades dentro de la red social (por ejemplo, nombre de usuario, tags definidos, comentarios hechos). Más específicamente la fórmula 2 muestra la definición de la estructura actor social:

$$Sa = \langle id; target_entity; name; entity_base; Group \rangle \quad (2)$$

Donde:

- id : el identificador único del actor social, esto es usado para identificar el sujeto o sujetos para todas las interacciones sociales del sistema;
- $target_entity$: el identificador de la entidad que es extendida por el actor social actual. En el caso de un usuario, este atributo contiene el id de su correspondiente entidad del tipo persona;
- $name$: es un identificador único legible para humanos que individualiza al actor social, esto es análogo al nombre de usuario (username) que es comúnmente encontrado en redes sociales existentes. Para proveer seguridad adicional, y para estar en línea con los requerimientos de autorización de las redes sociales actuales, la estructura actor social puede ser extendida con atributos como 'password' y 'email' (que por simplicidad son omitidas de esta definición);
- $entity_base$: especifica la base de las entidades que es asignada y pertenece al actor social actual. Esta base de entidades (tal cual como fue definida en la subsección

anterior) es usada para almacenar todas las entidades que fueron y serán creadas por el actor social actual. Como dueño, el mismo puede conceder permisos para que otros también tengan acceso a las entidades que se encuentran almacenadas en su base de entidades;

- *Group*: conjuntos de otros usuarios a los cuales se les asigna un nombre. La aplicación más común de esto es la creación de grupos relativos al usuario actual, tales como ‘amigos’ o ‘familia’, que luego son usados para clasificar otros usuarios.

La Figura 2 muestra la representación de como un actor social extiende una entidad. Las estructuras de actor social están

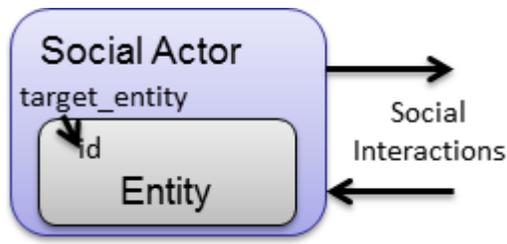


Fig. 2. La estructura Social Actor extiende una entidad.

conectadas a las entidades mediante su atributo `target_entity` (como puede verse en la Figura 2). Los actores sociales que están así conectados a una entidad son habilitados a actuar como los representantes de esa entidad dentro de la red social (por ejemplo, un usuario representando a la persona real y un gerente de comunidad hablando en representación de una empresa) y como tales pueden iniciar o participar en interacción sociales como si fueran las entidades que representan. Algunos ejemplos de estas interacciones sociales incluyen la compartición (por ejemplo, un actor social concediendo acceso a un recurso a otro), tagueo, comentarios y mensajería personal (por ejemplo, un actor social enviando un mensaje a otro). Más detalles de los servicios basados en estas interacciones sociales serán presentados en la próxima sección. Antes de presentar estas funcionalidades a los usuarios finales, Social Core especializa a los actores sociales en dos estructuras adicionales:

- *Users*: en los usuarios el atributo `target_entity` solo puede apuntar a entidades del tipo persona (Person Entity). De esta manera, mientras que la entidad del tipo persona contiene toda la información relacionada con la persona física (por ejemplo, nombre, apellido, dirección), la estructura usuario por su parte, es usada para crear y participar en las interacciones sociales.
- *Community*: la estructura comunidad representa un grupo de personas que se reúne para discutir, crear y compartir sobre un concepto, tema o entidad. En lugar de representar a un individuo como en el caso de la estructura usuario, la estructura comunidad es usada para crear grupos de interés.

Es importante mencionar que ninguna de las definiciones anteriores proponen limitación alguna que evite que múltiples

estructuras de usuario extiendan a la misma persona. De esta manera, la plataforma permite una situación como la que se ejemplifica en la Figura 3. Estos múltiples usuarios pueden

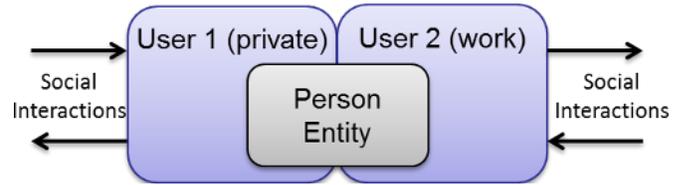


Fig. 3. Dos o más usuarios pueden estar conectados a la misma entidad de persona.

representar facetas independientes y diferentes de la misma persona. El ejemplo de la Figura 3 muestra que el User1 es un usuario privado (que podría, por ejemplo, compartir las fotografías de su familia) mientras que User2 es un usuario de la misma persona relacionado al ambiente laboral (que podría, por ejemplo, colaborar en la creación de documentos con sus colegas).

C. Pares y Perfiles

La mayoría de las redes sociales actuales adoptan un arquitectura centralizada (es decir, donde toda la información de todos los usuarios se almacena en un conjunto de servidores que actúan lógicamente como una unidad). Sin embargo, la arquitectura peer-to-peer (red de pares), en la que la información perteneciente a distintos usuarios es lógica o físicamente contenida en distintas unidades de almacenamiento, también es considerada interesante por las implicaciones que acarrea para la protección de la privacidad de los usuarios. Por estas razones (elaboradas en mejor detalles en la sub-sección III-D) Social Core define dos estructuras adicionales: el par y el perfil. Éstas son usadas junto a las ya definidas, actor social y usuario, para proteger la privacidad de los usuarios.

1) *Estructura de Pares para almacenamiento distribuido*: las estructuras de pares (peers) son unidades de almacenamiento manejadas por Social Core y por los sujetos que participan en el mismo. Cada una de estas estructuras de pares tiene una base de conocimientos y base de entidades (previamente definidas en la subsección II-A) independientes de la plataforma y bajo el control exclusivo de un usuario dueño de la misma. Como se muestra en la Figura 4, esta información se encuentra almacenada separadamente de la información de otros usuarios y de la plataforma e inicialmente solo el usuario dueño tiene acceso a la misma.

La Figura 4 muestra que cada uno de los pares tiene su propia KB (Knowledge Base) y EB (Entity Base) asignada y separada del resto de la plataforma. El diseño de la infraestructura de Social Core permite que cada uno de estos pares sean almacenados en servidores lógicamente distintos (por ejemplo en distintos contenedores de un servidor cloud) o físicamente en distintos servidores. Gracias a esto la plataforma garantiza que cada sujeto estará siempre en control de la información almacenada en su par asignado y que nadie podrá acceder a

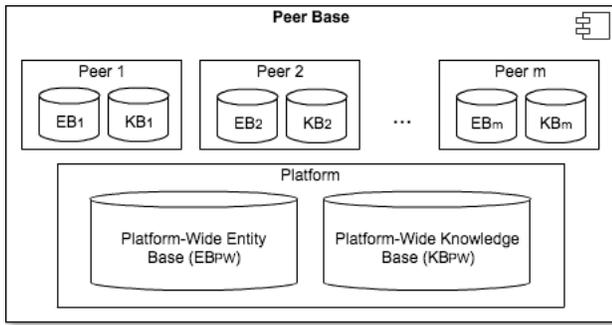


Fig. 4. Organización en Pares con bases de conocimiento y entidades independientes.

esta información a no ser de haber sido habilitado (mediante el uso de control de acceso) por este mismo sujeto.

2) *Perfiles como indirección a la información*: en lugar de habilitar a acceder directamente a la información contenida dentro de las bases de conocimiento y entidades, Social Core permite la creación de estructuras de perfil (profile) para dar respuestas controladas (que solo revelan información parcial o oscurecida) a los pedidos de información que llegan a una determinada estructura de par.

La estructura de definición de perfil PD es usada para definir el subconjunto de información que se incluirá en el perfil compartido en lugar de la entidad perfilada PE . Así la estructura de definición de perfil está dada por la tupla $PD = \langle ID, PE, \{PP\}, \{GP\}, \{NR\} \rangle$ donde:

- ID es un identificador numérico único que se utiliza como referencia principal a PD ;
- PE (Profiled Entity) es el id de la entidad perfilada, es decir, la entidad a la cual el perfil actual se refiere;
- $\{PP\}$ es un conjunto que especifica los distintos parámetros que son usados por los algoritmos especializados para oscurecer y parcializar la información contenida en PE , la especificación de estos parámetros así como de los algoritmos usados depende del tipo de entidad al que se aplican y van más allá del alcance del trabajo actual;

Aplicando la previa estructura de definición del perfil a la entidad a la cual se refiere materializa la información en una estructura de perfil de entidad. La estructura de perfil de entidad está definida como la tupla $EP = \langle ID, U, S, \{A\}, \{R\}, \{AR\} \rangle$ donde:

- ID es un identificador numérico único;
- U es el id del usuario que materializó el perfil y es ahora su dueño;
- $\{A\}$ es un conjunto de atributos tal como fueron definidos anteriormente para las entidades, sin embargo, debido a los procesos usados para oscurecer y parcializar la información, estos atributos podrán ser diferentes a los que se encuentran en la entidad perfilada;
- $\{R\}$ es un conjunto de relaciones tal como fueron definidas anteriormente para las entidades, al igual que con los atributos, éstas pueden ser distintas a las relaciones de la entidad perfilada;

- $\{AR\}$ (Agreed Requirements) conjunto de los requerimientos acordados entre el dueño de la entidad perfilada y el creador del perfil actual para el manejo de la información contenida en el perfil, esta propiedad puede ser controlada para comprobar que el uso que se le da al perfil actual se adecua a los términos del acuerdo con el dueño original de la entidad perfilada.

Es importante notar que mientras que la estructura de definición de perfil (es decir, el filtro que se aplica a la información original antes de compartirla) es almacenada en el par que es origen de esta información, el perfil de entidad materializado (que contiene la información compartida ya oscurecida o parcializada) se almacena en el par del usuario que hizo el pedido original de esta información. Esta disposición de datos puede verse claramente en la Figura 5. La parte

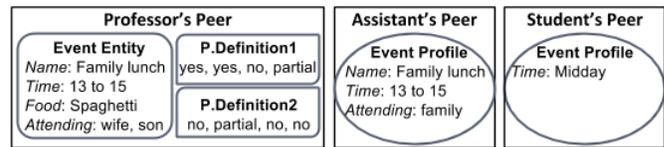


Fig. 5. Ejemplo de uso de perfiles preservadores de la privacidad.

izquierda de la Figura 5 muestra un evento (family lunch o almuerzo familiar) que pertenece al par del profesor. Este profesor ha creado dos estructuras de definición de perfil, que se presentan en la figura de manera simplificada, para definir como este evento es compartido con su secretaria y sus estudiantes. El resto de la figura muestra las estructuras de par que pertenecen a la secretaria y a un estudiante con las estructuras de perfil de entidad materializadas creadas por la aplicación a partir de las restricciones definidas por el profesor. Nótese que estas materializaciones incluyen ejemplos de información omitida (el atributo 'food' o comida no es compartida a ningún perfil), así como ejemplos de información oscurecida o parcializada (el tiempo del evento se convierte en 'MIDDAY' o medio día). La realización de las materializaciones del ejemplo anterior requieren la formalización de funciones que pueden proveer diferentes niveles de abstracción a partir de la información original, permitiendo a los dueños elegir un particular nivel de privacidad durante la creación de la estructura de definición de perfil. Sin embargo, esta definición formal está más allá del alcance de el trabajo actual y será explorada en trabajos futuros.

III. SERVICIOS

Esta sección introduce los principales servicios de usuario final ofrecidos por Social Core.

A. *Compartición de Entidades Semánticas y Control de Acceso*

Habiendo ya definido los sujetos y los objetos en la sección anterior, necesitamos ahora una forma de proteger la privacidad de los usuarios denegando el acceso no autorizado a los recursos pero al mismo tiempo permitiendo el compartir y la creación de enlaces para las partes debidamente autorizadas. El

componente de control de acceso de Social Core está basado en la teoría RelBAC (Relation Based Access Control). RelBAC es un nuevo modelo y una lógica que ha sido introducida en [14], que se propone como objetivo principal el enfrentar el problema del control de acceso en aplicaciones web modernas. La mayoría de los sistemas que manejan grupos de usuarios (por ejemplo, lista de amigos), tags/comentarios, eventos o colecciones de foto tienen en común la tendencia de organizar tanto sus recursos como sus relaciones sociales en estructuras similares a los árboles; donde cuanto más profunda una categoría dentro del árbol, más específico será el recurso contenido en la misma. Esto también se cumple en Social Core donde, explotando la traducción de clasificaciones como las anteriormente mencionadas y directorios web a ontologías ligeras (lightweight ontologies, introducidas en [19]) podemos representar (como se muestra en [17] y [2]) ambas redes de usuarios (sujetos) y organizaciones de entidades (objetos) como ontologías ligeras. Esto nos permite cumplir con los principales requerimientos del modelo RelBAC que define sus principales elementos como:

- *Subjects*: un sujeto es un usuario o un conjunto de usuarios que se propone acceder a algunas entidades. La relación 'IS-A' es usada entre los conjuntos de sujetos, creando así un jerarquía que va desde los grupos de sujetos más numerosos y generales (es decir, el grupo "all" o "todos") hasta sujetos menos numerosos y más específicos (por ejemplo, "work" o "trabajo", "family" o "familia").
- *Objects*: un objeto es una entidad o un conjunto de entidades a las cuales los sujetos se proponen acceder. Similarmente a los sujetos la relación 'IS-A' es usada entre los conjuntos de objetos.
- *Permissions*: los permisos son representados por la tripla $\langle Subject, Object, Operation \rangle$, donde Operation es el nombre de la operación que se pretende hacer (por ejemplo, Leer, Escribir, Borrar). Es usada para permitir a uno o más sujetos el realizar la operación dada sobre uno o más objetos.

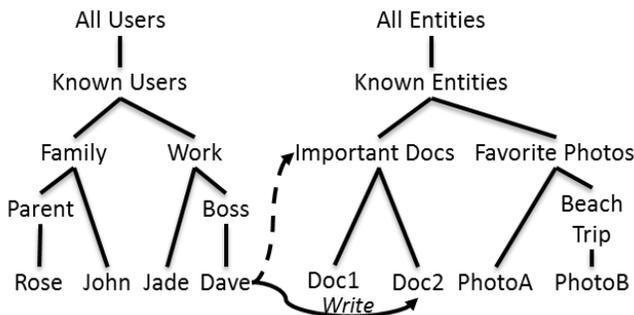


Fig. 6. Jerarquías de sujetos y objetos RelBAC.

En RelBAC todos los mencionados elementos pueden ser directamente expresados en una Lógica Simbólica (Description Logic - DL). Como se explica en [17] los sujetos y objetos son

formalizados como conceptos, mientras que los permisos son formalizados como roles DL (es decir, relaciones binarias). Así las reglas de control de acceso que expresan el tipo de derechos de acceso que los sujetos habrán sobre los objetos basándose en los permisos definidos, son también formalizadas como se presenta en [16] y [14]; lo que permite el cómputo del control de acceso y la sugerencia de nuevas reglas de control de acceso mediante la aplicación de razonadores DL (DL reasoners) estándar [2] sobre las estructuras mencionadas del modelo. Esto permite al Social Core el ofrecer las siguientes ventajas relativas al control de acceso:

- En los métodos convencionales de control de acceso basados en roles (Role-Based action control o RBAC), un particular conjunto de permisos es asignado a un rol. Cuando se necesita introducir cambios, es muchas veces necesario crear nuevos roles o borrar roles que ya no son usados (lo que puede potencialmente dejar al sistema de control de acceso en un estado inconsistente o inseguro). En RelBAC los permisos se encuentran totalmente desacoplados de los sujetos y objetos por lo que todos los cambios deben satisfacer las condiciones impuestas por el modelo semántico subyacente, lo que permite que el sistema evite entrar en los mencionados estados problemáticos;
- Los enlaces relacionales entre sujetos y objetos dentro de la red social pueden ser explotados para sugerir caminos candidatos para la propagación de permisos. Por ejemplo en la Figura 6) si el permiso es concedido al grupo de usuarios 'Work' o 'Trabajo', esto significa que ambos los usuarios con el nombre 'Jade' y 'Dave' recibirán también el permiso. Adicionalmente, analizando los sujetos, objetos y permisos definidos se vuelve posible inferir y sugerir permisos adicionales al usuario. Por ejemplo, si el usuario 'Dave' tiene acceso y usa regularmente varios documentos en la categoría 'Important Docs' o 'Documentos Importantes' (por ejemplo 'Doc1' y 'Doc2' en la Figura 6) entonces el sistema podría sugerir conceder los mismos permisos sobre las categorías más generales que incluyan esos documentos (esto es mostrado como una línea punteada en la Figura 6). Más detalles sobre la inferencia de nuevas reglas mediante el uso de este modelo y lógica modal pueden ser encontrados en [12] y [28].
- Cada usuario es capaz de definir sus propias jerarquías de sujetos (incluyendo por ejemplo, amigos, compañeros de trabajo, familia) y objetos personalizadas (incluyendo por ejemplo, documentos de trabajo, fotos de mi tiempo libre). Así, diversos usuarios pueden tener jerarquías funcionalmente similares pero con diferencias en los nombres y la estructura de cada elemento. Para poder hacer estas jerarquías interoperables y superar el problema de la heterogeneidad semántica Social Core es capaz de utilizar el pareo semántico (semantic matching, detallado en [13] y [11]) sobre las jerarquías creadas por los distintos usuarios.

- Gracias al procedimiento usado para convertir las jerarquías sujetos y objetos en ontologías ligeras, RelBAC es capaz de razonar sobre ellas (por ejemplo usando, [F2] y [F3]) y así es posible agregar clasificaciones para objetos y sujetos bien estudiadas y probadas según el caso lo requiera.

B. Anotaciones Semánticas

Social Core permite a sus usuarios el anotar a sus entidades, así como a las entidades de otros usuarios siempre y cuando hayan sido concedidos los permisos necesarios para hacerlo. Las anotaciones y los tags son adjuntados a las entidades almacenadas en almacenamiento semántico (semantic storage) de la Figura 1), y gracias a esto las operaciones como búsqueda y pareo también pueden ser ejecutadas sobre estas anotaciones. La plataforma soporta anotaciones de tres tipos:

- Free text annotations: o anotaciones de texto; estas anotaciones están compuestas por una secuencia arbitraria de caracteres introducidos por el usuario (por ejemplo, “java”, “party”, “xyz”). Esta es probablemente el tipo de anotación más comúnmente usada en redes sociales (por ejemplo, Flickr⁴). El significado que este tipo de anotaciones lleva es definido por el usuario durante su creación; sin embargo, a este tipo de anotaciones no se asigna ningún tipo de semántica que sea directamente procesable por el sistema.;
- Semantic annotations: o anotaciones semánticas, este tipo de anotaciones es una tupla que consiste de una cadena de caracteres junto con el concepto que esta cadena representa, por ejemplo $\langle \textit{“perro”}, \textit{PERRO} \rangle$ ⁵. Una ventaja de las anotaciones semánticas es que son representadas en un lenguaje formal y por lo tanto permiten el razonamiento automatizado sobre ellas, por ejemplo la búsqueda del concepto ANIMAL encontraría entidades anotadas con los conceptos de PERRO o GATO. Ejemplos de este enfoque pueden ser encontrados en Tagasauris⁶ y L!NKs⁷;
- Entity annotations: o anotaciones de entidades, tales anotaciones están definidas por una tupla que consiste de una cadena de caracteres y una entidad representado por la anterior cadena, por ejemplo, $\langle \textit{“Trento”}, \textit{Trento} \rangle$ ⁸. Tales anotaciones pueden ser usadas, por ejemplo, cuando una entidad del tipo imagen es anotada con la gente que se encuentra en ella o con el lugar donde la imagen ha sido tomada. Una ventaja de las anotaciones de entidad es que permiten la interconexión de recursos relaciones y permiten así que se ofrezcan mejores servicios de navegación y búsqueda sobre ellas, por ejemplo, el encontrar imágenes de un lugar en particular o de una persona en particular.

⁴<http://www.flickr.com/>

⁵Para mantener la claridad, los conceptos serán escritos con letras mayúsculas

⁶<http://www.tagasauris.com/>

⁷www.insemtives-links.net

⁸Para mantener la claridad, las entidades son escritas en itálicas e inician con una letra mayúscula

El lector interesado puede referirse a [1] para una visión más completa de los tipos varios de anotaciones, sus relativas ventajas y debilidades y las aplicaciones que permiten.

C. Siguiendo Live Topics

Usando los tres componentes principales de Social Core (Sujetos, Conceptos y Objetos) como pilares, el servicio llamado Live Topics (temas en vivo) provee un mecanismo que permite a sus usuarios el subscribirse y seguir noticias/actualizaciones. Cada Live Topic consiste de los siguientes tres componentes:

- La fuente: los sujetos que son la fuente de las actualizaciones (por ejemplo usuarios individuales, grupos de usuarios, o todos los usuarios en el sistema);
- El/los temas: el tema de Live Topic se define basado en las anotaciones con la que el contenido ha sido marcado (estas anotaciones pueden ser del tipo texto, semántica o de entidad). Por ejemplo, un Live Topic podría usar el concepto ANIMAL como tema y entonces una entidad marcada con la anotación GATO sería relevante para el mismo Live Topic;
- El/los tipos: uno o más tipos de entidades al cual las noticias retornadas por el Live Topic deben pertenecer (por ejemplo, imágenes, lugares y personas). Es también posible especificar que entidades de cualquier tipo pueden ser retornadas.

El servicio de Live Topics considera el control de acceso, en el sentido que todos los Live Topics creados por un usuario retornarán solo las entidades a la cuales ese usuario tiene acceso (es decir son entidades públicas o fueron compartidas con ese usuario). Para proveer esta funcionalidad se hace uso de búsqueda semántica (reportada en [15]); más precisamente, para los conceptos asociados con un Live Topic con aquellos usados a la hora de anotar las entidades. Por ejemplo, para el Live Topic definido como “seguir imágenes relacionados con el concepto DEPORTE y la entidad *Argentina*”, el sistema es capaz de retornar imágenes compartidas de un evento de basketball que tuvo lugar en Buenos Aires. Más detalles sobre como la búsqueda semántica está implementada pueden ser encontrados en [15]. Las noticias así retornadas a un usuario por un Live Topics también almacenan un indicador de si ya fueron leídas por el usuario, lo que permite al sistema ordenarlas según su estado de lectura y eventualmente al usuario archivarlas o borrarlas.

Utilizando las anotaciones semánticas creadas por sus usuarios y actores sociales, Social Core provee un mecanismo para computar los actores sociales con intereses y gustos similares al usuario origen. El mecanismo para lograr esto esta basado en los Live Topics descritos anteriormente; es decir, el sistema asume que dos actores sociales son más similares cuanto más Live Topics y sus resultados tengan en común. El re-usar los mismo mecanismos del Live Topics implica que la sugestión de actores sociales también toma ventaja del razonamiento automatizado. Por ejemplo si dos actores sociales han definidos sus Live Topics con los conceptos LENGUAJE_DE_PROGRAMACIÓN y JAVA

respectivamente, entonces serán considerados con intereses similares entre sí. Nótese que este servicio está definido para ambos subtipos de actores sociales: usuarios y comunidades. Y como tal puede ser usado para sugerir a usuarios el unirse a nuevas comunidades, para sugerir a los administradores de las comunidades a invitar nuevos miembros, o para construir contactos nuevos de usuario a usuario.

D. Protección de la Privacidad por Diseño

La protección de la privacidad y la regulación del uso de los datos personales de los usuarios de sistemas informáticos es un problema que, pese a ser frecuentemente subestimado por muchas de las personas que rutinariamente interactúan con redes sociales y sistemas web, actualmente es frecuente objetivo de la atención de varios gobiernos y entes reguladores. Como el consejo europeo ha discutido en su recomendación CM/REC(2010)13 sobre el almacenamiento de información de los usuarios [9], la recolección, enlace, cálculo, comparación y procesamiento estadístico de la información de los usuarios puede impactar a la privacidad de los ciudadanos significativamente. Esto es debido a que a partir de estos procesos se pueden obtener no solo detalles de la personalidad, comportamiento, intereses y hábitos de los usuarios sino que también permiten el predecir y derivar estadísticamente (big data analytics) detalles de los usuarios que no fueron capturados directamente por el sistema. Por ejemplo detalles sobre las opiniones políticas, creencias religiosas, inteligencia u orientación sexual pueden ser automáticamente predichos a partir de 'likes' de Facebook [23]. Para cumplir con estas recomendaciones y evitar incurrir en problemas éticos relacionados con la privacidad y el manejo de los datos de los usuarios, Social Core ha decidido seguir los siguientes requerimientos:

- **Legitimidad y consentimiento informado:** la recolección y procesamiento de información personal debe ser legítima, lo que normalmente implica que los sujetos de los datos deben haber dado su consentimiento informado.
- **Especificación y cumplimiento de propósitos:** los datos personales deben ser usados y recolectados para propósitos legítimos y deben ser usados a posteriori exclusivamente para estos propósitos.
- **Minimización de datos:** la cantidad de los datos personales y la extensión a la que son procesados debe ser minimizada, lo que implica que mientras sea posible los perfiles de usuarios deben ser anonimizados o seudonimizados.
- **No a los datos sensibles:** la recolección y procesamiento de las así llamadas categorías especiales de datos debería estar prohibida como principio.
- **Transparencia y derechos del sujeto de los datos:** los sujetos de los datos deben tener derecho de acceder a su información en todo momento, así como el derecho de ser informados sobre la lógica involucrada durante el procesamiento de estos datos. Por sobre todo, los sujetos

de estos datos deben tener derecho a corregir, borrar o bloquear sus datos.

- **Seguridad:** el controlador de los datos debe implementar medidas técnicas y organizacionales apropiadas para garantizar la protección de la información personal de sus usuarios.

Para responder a estos requerimientos Social Core ha decidido seguir un enfoque de privacidad desde el diseño (privacy by design) que implica que los detalles relativos a la privacidad de los usuarios son considerados por los desarrolladores desde la etapa de diseño del sistema. Más específicamente, Social Core implementa, utilizando las estructuras definidas en la sección II, tres principios guía para la protección de la privacidad:

- 1) *Separación de la información bien definida entre la información perteneciente a diferentes sujetos.* Para garantizar que los sujetos controlan su propia información, Social Core crea ambientes distribuidos para almacenar el conocimiento y la información de cada sujeto. Así, como estas unidades de almacenamiento pueden ser distribuidas lógicamente o físicamente, la información personal de un sujeto es aislada del resto de la información en la plataforma (como se explica en la subsección II-C).
- 2) *Manejo de la identidad centrado en el usuario.* Social Core da a cada sujeto el control del flujo de la información que puede identificarlo personalmente. Adicionalmente, mediante la creación de varias estructuras de usuario para el mismo par (como se explica en la subsección II-B), Social Core permite la definición de seudónimos para distintos tipos de interacciones.
- 3) *Una de-construcción y re-imaginación de el perfilamiento de información.* Mientras que los objetivos de alto nivel de los perfiles de Social Core (definidos en la subsección II-C) son los mismos que los encontrados en los perfiles convencionales, es decir ambos son estructuras de almacenamiento de información mantenidas y actualizadas autónomamente, los perfiles de Social Core fueron enfocados en revertir las prácticas convencionales de perfilamiento haciéndolas transparentes y controlables por los sujetos a los cuales se refieren.

De esta manera, Social Core permite a las personas (es decir a los usuarios) definir perfiles que contienen y revelan solo información oscurecida o parcializada. Adicionalmente, los propósitos y los usos de los datos personales son especificados y limitados mediante el uso de una póliza de privacidad, con la promesa que estos requerimientos serán hechos cumplir por la plataforma y que todos los datos compartidos no serán mal usados ni intercambiados en contra de la voluntad de sus sujetos. Un ejemplo del funcionamiento de los perfiles y sus limitaciones es mostrado en la Figura 7.

El uso de la información contenida en los perfiles materializados está restringido por los requerimientos acordados (agreed requirements, introducido en la subsección II-C) que se encuentran especificados usando una póliza de privacidad PPL [29]. El hacer cumplir estas pólizas al nivel de

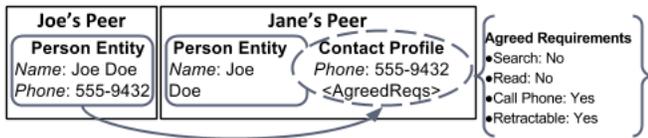


Fig. 7. Ejemplo de compartición privada mediante el uso de perfiles.

la plataforma garantiza que la información compartida sea solo usada para los propósitos acordados. Como se muestra en la Figura 7, si el doctor utiliza perfiles para compartir su número de teléfono con sus asistente entonces el doctor podría restringir el uso de esta información a solo “llamar por teléfono”. De esta manera cualquier otra operación sobre este dato, como compartir el número de teléfono con otro usuario o incluso leer para mostrar el número de teléfono en la pantalla, serían restringidas. El asistente puede llamar por teléfono al doctor a través del sistema pero el verdadero número de teléfono del doctor no es revelado al asistente. De hecho, una vez despedido el asistente el doctor puede elegir retractar el perfil con su información de contacto y al (ex)asistente ya no le será posible llamar por teléfono al doctor. Detalles sobre la negociación de requerimientos y la especificación de los requerimientos acordados de los perfiles serán clarificados en trabajos futuros.

IV. CASOS DE USO

Esta sección introduce los principales casos de uso que serán usados para la validación de los servicios de Social Core.

A. SmartCampus: Universidad Amiga

SmartCampus es un proyecto finalizado a fines del año 2014 y que fue financiado y llevado a cabo mediante la colaboración del consorcio de investigación TrentoRise⁹ y la Universidad de Trento. El proyecto estuvo principalmente orientado al soporte de los estudiantes de la Universidad de Trento.

1) *Descripción del Proyecto:* La idea principal del proyecto es el armonizar la gran variedad de distintos sistemas, infraestructuras, servicios y dispositivos que están disponibles y relacionados a los estudiantes de la universidad y sus profesores. En particular, uno de los enfoques principales fue el de dar soporte a los estudiantes universitarios que se mudaban a una nueva ciudad para estudiar (Trento en este caso) proveyendo servicios personalizados y contextualizados a través de sus dispositivos móviles (por ejemplo celulares y tabletas). Un aspecto clave es que los estudiantes no son simplemente considerados “consumidores” de estos servicios sino también, mediante el uso de tecnologías colaborativas, son considerados proveedores y curadores de la información en el sistema. SmartCampus toma el enfoque “sistema de sistemas” y como tal ofrece servicios relacionados al transporte, perfiles de estudiantes, lugares interesantes, información sobre cursos, entre otros. Todos estos servicios son ofrecidos como parte de los servicios normales que la universidad ofrece a

sus estudiantes, por lo que los estudiantes tienen acceso automático a ellos sin tener que pasar por un proceso adicional de inscripción. Para dar una mejor idea del tipo de servicios

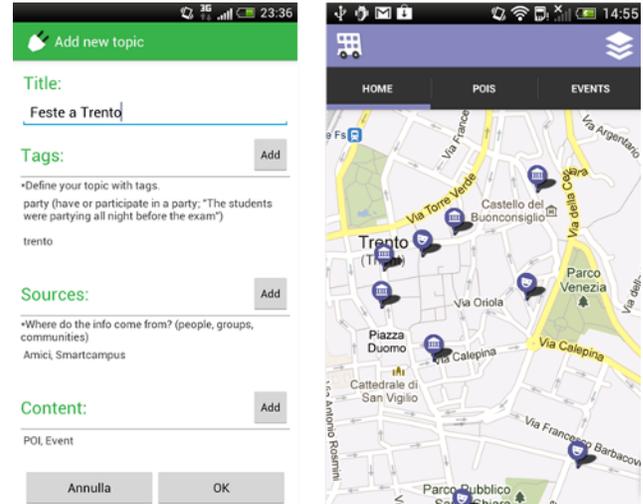


Fig. 8. Imágenes en pantalla de la aplicación para Android SmartCampus

ofrecidos por la plataforma, los siguientes son algunos de sus principales sub-sistemas:

- *Community Manager:* el manejador de comunidad permite la creación y manejo de las conexiones sociales (representadas como una ontología ligera de sujetos). Esta aplicación provee servicios para el descubrimiento de nuevos usuarios, para la organización de usuarios en grupos y para compartir entidades con los mismos (por ejemplo el compartir eventos, lugares y el historial académico son actualmente soportados). Finalmente, esta aplicación también permite controlar lo que los otros están compartiendo y el seguir (mediante la creación de Live Topics) eventos, lugares y otras entidades interesantes. La imagen de la izquierda en la Figura 8 muestra un ejemplo de la pantalla de creación de un Live Topic;
- *Discover Trento:* descubre a Trento provee información y recorridos virtuales de la ciudad de Trento. Los usuarios pueden buscar lugares y eventos en la ciudad usando información pública o información compartida por otros usuarios; así como seguir y recibir notificaciones (a través de los Live Topics) de, por ejemplo, “happy hours” de los establecimientos locales. La imagen derecha de la Figura 8 muestra el resultado de una búsqueda de proximidad sobre los eventos que actualmente están sucediendo alrededor del usuario;
- *Portfolio:* el historial académico permite al usuario crear y compartir múltiples curriculum vitae y perfiles editables usando como base la información certificada y actualizada de la universidad. Estos perfiles pueden ser posteriormente compartidos (mediante el uso de las funcionalidades de control de acceso proveídas por Social Core) y los usuarios que se subscriben a estos perfiles son capaces de recibir notificaciones de actualizaciones

⁹<http://www.trentorise.eu/>

al perfil;

- *Journey Planner*: el planificador de viajes provee mapas y herramientas para la planificación de viajes usando los distintos métodos de transporte disponibles en la ciudad (por ejemplo, transporte público, bicicletas, caminar, etc.). Adicionalmente los usuarios pueden también recibir notificaciones en tiempo real sobre las mejores rutas que tomar para llegar a su destino y sobre informaciones de retrasos en los medios de transporte público y congestiones.

2) *Arquitectura de Integración*: Social Core provee a la plataforma SmartCampus con los servicios descritos en las secciones previas. Estos incluyen el almacenamiento de entidades, manejo de los usuarios y de sus interacciones, servicios semánticos como la búsqueda, la creación de Live Topics y las noticias que estos generan. La Figura 9 muestra una representación arquitectural de alto nivel de la plataforma Smartcampus y de cómo Social Core está integrado en la misma. Como puede apreciarse en la Figura 9, estas funcional-

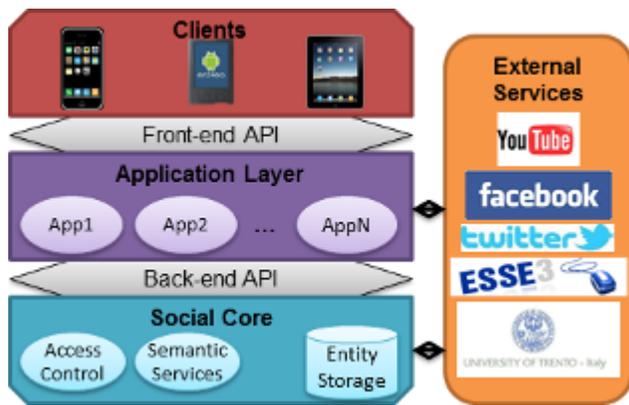


Fig. 9. Arquitectura de la plataforma SmartCampus

idades son proveídas a través de APIS HTTP que son luego usadas por las aplicaciones de la plataforma (por ejemplo community manager, discover Trento, etc.), las cuales hacen llegar estos servicios a los usuarios finales por medio de interfaces implementadas en el código nativo de las plataformas móviles. Esta arquitectura permite además la integración y enlace con servicios externos a la plataforma tanto al nivel de servidor como el de aplicaciones.

3) *Primeros experimentos*: El testeo real de la plataforma SmartCampus fue iniciado en Octubre del 2012 con una población de casi cien estudiantes universitarios. Cada estudiante fue proveído con un smartphone con las aplicaciones de SmartCampus ya instaladas en el mismo y fue brevemente introducido a algunas de sus funcionalidades. Los datos reportados en este trabajo representan el feedback obtenido por los estudiantes a lo largo de seis meses de uso de la plataforma y como resultado de ellos varias mejoras a la plataforma fueron planeadas e implementadas. Mientras que estudios más acabados sobre el uso de Social Core serán llevados a cabo en el futuro, la Tabla I contiene valores de uso por parte de los usuarios de prueba, relevantes para obtener indicios de los

TABLE I
ESTADÍSTICAS DE USO DEL SISTEMA SMARTCAMPUS.

Usuarios	98	Live Topics	155
Grupos de Usuarios	119	Actualizaciones Live Topics	6253
Entidades	2309	Entidades compartidas	203
Entidades Community	2065	Anotaciones texto	145
Entidades User	253	Anotaciones semánticas	29

posibles resultados finales. Un breve análisis de los valores presentados en la Tabla I es presentado a continuación:

- *Usuarios*: esto representa el número de estudiantes que ayudaron con la prueba de la plataforma. Nótese que los usuarios creados por los desarrolladores del sistema no fueron incluidos en estos números;
- *Grupos de Usuarios*: estos representan los subgrupos creados por los usuarios para categorizar a otros usuarios, como por ejemplo ‘amigo’ entre otros. De los 98 usuarios, 59 de ellos agregaron al menos un amigo.
- *Entidades*: esto incluye la entidad del tipo persona para cada uno de los usuarios, además de los lugares, eventos y entidades de portafolio usadas en la plataforma.
 - *Entidades Community*: subgrupo de entidades que fueron creadas por tantos por los administradores de la plataforma (para iniciarla ya con contenido útil), o bien fueron creadas por los usuarios. Éstas representan un gran porcentaje de las entidades creadas, lo que es consistente con las etapas iniciales de prueba de varios sistemas;
 - *Entidades User*: subgrupo de las entidades creadas por los propios usuarios. En la actualización probada para SmartCampus, estas fueron principalmente entidades del tipo portafolio.
- *Live topics*: este número representa a los Live Topics creados explícitamente por los usuarios (mediante el uso de la interfaz demostrada en la Figura 8) o bien generados por el sistema como respuesta de pedidos de ‘seguir’ o ‘recibir actualizaciones’ de ciertas entidades.
- *Actualizaciones Live Topics*: notificaciones generadas por el sistema y enviadas a los usuarios sobre las noticias relacionadas a las entidades que han decidido seguir;
- *Entidades compartidas*: esto representa el número de directivas de control de acceso creadas por los usuarios. La versión probada para SmartCampus solo permitía el compartir entidades del tipo portafolio por lo que el número evidencia un uso interesante de esta funcionalidad.
- *Anotaciones texto*: número de las anotaciones de texto agregadas por los usuarios a otras entidades.
- *Anotaciones semánticas*: número de las anotaciones semánticas agregadas por los usuarios a varias entidades. Un reporte de los usuarios sobre esta funcionalidad revela que los usuarios no eran conscientes de las ventajas de usar una anotación semántica sobre una anotación de texto. Esto causó el reformateo de la interfaz de anotaciones para futuros usos de Social Core.

En líneas generales, a pesar de haber representado la primera prueba en vivo de las funcionalidades ofrecidas por Social Core, los estudiantes han demostrado un buen nivel de interés y participación. Al mismo tiempo las lecciones aprendidas en este experimento serán aplicadas en trabajos futuros.

B. SmartSociety: Ciudadanos y Máquinas Colaborando

SmartSociety es un proyecto EU FP7 FET que explora como los sistemas colectivos adaptativos (Collective Adaptive Systems o CAS) que son compuestos por personas y máquinas pueden ayudar a solucionar problemas relacionados con la vida urbana. La palabra ‘Smart’ se refiere a como tecnologías innovadoras, sociales, móviles y basadas en sensores pueden dar soporte a sistemas colectivos de personas y máquinas que son capaces de usar recursos compartidos y limitados (como por ejemplo la red de transporte) en maneras más efectivas y sostenibles. Social Core es entonces usado en el contexto del proyecto SmartSociety para proveer una red social de usuarios interconectados donde estos usuarios pueden representar tanto a personas reales como a agentes computacionales. La entidad socio-técnica resultante al corazón del proyecto SmartSociety es conocida como un sistema colectivo adaptativo híbrido consciente de su diversidad (Diversity Aware Collective Adaptive System o (HDA-CAS)). El componente dentro de la plataforma de SmartSociety que implementa Social Core es conocido como Peer Manager (manejador de pares). Así Peer Manager puede ser visto como el almacenamiento principal de información personal dentro de la plataforma de SmartSociety y, como tal, tiene los siguientes objetivos:

- 1) Almacenamiento de metadatos y semántica mediante la implementación del modelo de representación detallado en la subsección II-A;
- 2) Protección de la privacidad a través del uso de las estructuras introducidas en la subsección II-C y los métodos de control de acceso de la subsección III-A;
- 3) Proveer búsqueda y pareo semánticos entre pedidos y ofrecimientos de distintos pares, tal como se explica en la subsección III-C.

La Figura 10 muestra una visión arquitectural de alto nivel de la plataforma SmartSociety y Peer Manager.

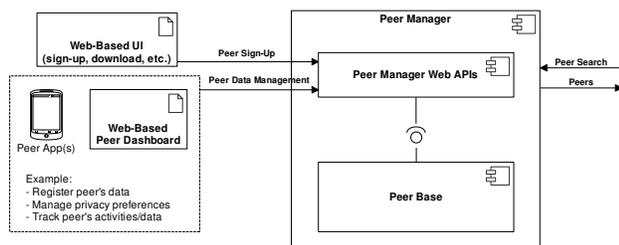


Fig. 10. Arquitectura del Peer Manager implementado a partir de Social Core para SmartSociety.

La parte izquierda de la Figura 10 muestra a dos clientes de usuarios finales interactuando directamente con Peer Manager

(a través de su API web). El primer ejemplo (en la parte superior izquierda) usa un cliente basado en la web para registrar nuevos usuarios a la plataforma SmartSociety, mientras que el segundo ejemplo (en la parte inferior izquierda) muestra un dispositivo móvil con un panel de control donde la persona a cargo puede elegir modificar su configuración de privacidad o controlar distintos tipos de información relacionada a su participación en la plataforma. Por otro lado, en la parte derecha de la Figura 10 se muestra una llamada (a la misma API web de Peer Manager) con un pedido de búsqueda de pares cuya respuesta es una lista de usuarios que cumplen con las condiciones especificadas en el pedido. Este tipo de interacción es usada con otros componentes internos de la plataforma SmartSociety para parear pedidos y soluciones en el sistema HDA-CAS. Pruebas y validaciones de Peer Manager implementado a partir de Social Core para la plataforma SmartSociety iniciarán hacia el final de este año. Esto extenderá los resultados obtenidos en el caso de uso SmartCampus para proveer una visión más acabada de la efectividad de los distintos servicios implementados por Social Core.

V. TRABAJOS RELACIONADOS

Hasta el día de esta publicación hemos encontrado un modesto número de estudios relacionados a redes sociales semánticas. El proyecto FOAF¹⁰ (Friend of Friend) es, sin embargo, usado normalmente como la principal referencia para las ontologías que describen a las personas y sus relaciones en un formato adecuado para el uso de las computadoras. Estudios como [10] han usado FOAD con RDF para capturar la información semántica contenida en las redes sociales. Nótese que este estudio también usa FOAF como una inspiración para definición de metadatos y relaciones para el tipo de entidad persona, sin embargo esta información se encuentra codificada mediante métodos adaptados para proveer servicios como la búsqueda y el taggeo semánticos. Centrándose más específicamente en el estudio de métodos de control de acceso para redes sociales, [5] describe un modelo basado en semántica utilizando OWL y SWRL; y propone una arquitectura para la autorización, administración y filtrado de pólizas basadas en relaciones de confianza. Aunque este enfoque tiene objetivos hasta cierto punto similares, el modelo RelBAC discutido en este trabajo está basado en las relaciones que existen entre sujetos y objetos. De mayor relevancia aún, una primera versión de RelBAC se encuentra ya implementada y siendo testada como parte de plataformas como SmartCampus y SmartSociety. En cuanto respecta a la protección de la privacidad y la privacidad por diseño para redes sociales trabajos, como [3] y [21] han sido considerados para los aspectos relacionados del diseño de Social Core y Peer Manager dentro del contexto de la plataforma SmartSociety. La creación y el manejo de redes sociales orientadas a los negocios/academia (explorada en [26]) es también relevante porque ilustra las ventajas y limitaciones de las redes sociales

¹⁰<http://www.foaf-project.org/>

actualmente existentes. Ejemplos incluyen la red social orientada al ambiente laboral Beehive de IBM estudiada en [6] y las científicamente orientadas Academia.edu¹¹, ResearchGate¹² y IamResearcher¹³ (analizadas en [25]). Estas han sido menos exitosas en atraer y mantener los niveles de actividad de sus usuarios por lo que si bien es cierto que el uso de las redes sociales en contextos no-recreacionales tiene tendencia en aumento durante los últimos años, sus influencias y resultados (por ejemplo, si la productividad es realmente mejorada o no) son mucho menos significativos [27]. Cabe, por último, mencionar que versiones preliminares de la investigación contenida en este trabajo fueron previamente presentados en [20] y [8].

VI. CONCLUSIONES Y TRABAJO FUTURO

Este trabajo ha introducido a Social Core, una infraestructura del lado del servidor orientada a redes sociales móviles, que fue usada como parte de la plataforma SmartCampus y se encuentra en fase de implementación y pruebas como parte de la plataforma SmartSociety. El enfoque principal de Social Core es el introducir tecnologías semánticas para la mejora de los servicios de las redes sociales (como por ejemplo las anotaciones semánticas, búsqueda semántica y el control de acceso semántico) mediante el uso de entidades que contienen metadatos y conceptos para representar a la información social. La gran atención prestada a la protección de la privacidad y los valores sociales por parte de Social Core no implica que todas las dificultades relacionadas fueron superadas. Todavía es necesaria una validación profunda de la efectividad de estas medidas dentro de la plataforma SmartSociety y, aún obteniendo buenos resultados, existe una gran variedad de desafíos futuros. Adicionalmente se deben enfrentar los detalles técnicos relacionados con proveer los exigentes servicios (en cuanto a espacio de almacenamiento y tiempo de ejecución) que Social Core ofrece escalados a un nivel de plataforma comercial y dentro de tiempos de respuesta humanamente aceptables. Otros trabajos futuros incluyen la expansión del testeo de algunas de las funcionalidades de Social Core, junto con una implementación completa de la teoría ReBAC para el control de acceso semántico. Adicionalmente, mientras que algunos de los servicios ofrecidos por Social Core (como Live Topics, control de acceso semántico o manejo transparente de datos personales) han sugerido emocionantes nuevas posibilidades, un estudio de usuarios más comprensivo y refinamiento en sus interfaces es necesario antes que su potencial impacto pueda ser mejor determinado.

RECONOCIMIENTOS

Esta investigación fue parcialmente financiada por SmartSociety, un proyecto del Seventh Framework Programme para la investigación de la comunidad europea bajo el acuerdo de fondos numero 600854.

¹¹<http://academia.edu/>

¹²<http://www.researchgate.net/>

¹³<http://www.iamresearcher.com/>

REFERENCES

- [1] P. Andrews, I. Zaihrayeu, and J. Pane. A classification of semantic annotation systems. *Semantic Web*, 3(3):223–248, 2012.
- [2] F. Baader, D. Calvanese, D. L. McGuinness, D. Nardi, and P. F. Patel-Schneider. *The description logic handbook: theory, implementation, and applications*. Cambridge University Press, New York, NY, USA, 2003.
- [3] Randy Baden, Adam Bender, Neil Spring, Bobby Bhattacharjee, and Daniel Starin. Persona: An online social network with user-defined privacy. In *Proceedings of the ACM SIGCOMM 2009 Conference on Data Communication*, SIGCOMM '09, pages 135–146, New York, NY, USA, 2009. ACM.
- [4] D.M. Boyd and N.B. Ellison. Social network sites: Definition, history, and scholarship. *Journal of Computer-Mediated Communication*, 13(1):210–230, 2007.
- [5] B. Carminati, E. Ferrari, R. Heatherly, M. Kantarcioglu, and B. Thuraisingham. A semantic web based framework for social network access control. In *In proceedings of the 14th ACM symposium on Access control models and technologies*, 2009.
- [6] Y.J. Chang, H.H. Liu, L.D. Chou, Y.W. Chen, and H.Y. Shin. A general architecture of mobile social network services. In *Convergence Information Technology, International Conference on*, pages 151–156, 2007.
- [7] R. Chenu-Abente, M. Menéndez, F. Giunchiglia, and A. De Angeli. An entity-based platform for the integration of social and scientific services. In *8th IEEE International Conference on Collaborative Computing: Networking, Applications and Worksharing*, 2012.
- [8] Ronald Chenu-Abente, Ilya Zaihrayeu, and Fausto Giunchiglia. A semantic-enabled engine for mobile social networks. In *The Semantic Web: ESWC 2013 Satellite Events*, volume 7955 of *Lecture Notes in Computer Science*, pages 298–299. Springer Berlin Heidelberg, 2013.
- [9] Council of Europe. Recommendation CM/Rec(2010)13 of the Committee of Ministers to member states on the protection of individuals with regard to automatic processing of personal data in the context of profiling. Available from <https://wcd.coe.int/ViewDoc.jsp?id=1710949>, 11 2010.
- [10] T. Finin, L. Ding, L. Zhou, and A. Joshi. Social networking on the semantic web. *Learning Organization, The*, 12(5):418–435, 2005.
- [11] A.A. Freihat, F. Giunchiglia, and B. Dutta. Approaching regular polysemy in wordnet. In *In proceedings of eKNOW 2013, The Fifth International Conference on Information, Process, and Knowledge Management*, 2013.
- [12] E. Giunchiglia, F. Giunchiglia, R. Sebastiani, and A. Tacchella. More evaluation of decision procedures for modal logics. In *Proceedings KR'98, 6th International Conference on Principles of Knowledge Representation and Reasoning*, pages 626–635, 1998.
- [13] F. Giunchiglia, A. Autayeu, and J. Pane. S-match: An open source framework for matching lightweight ontologies. *Semantic Web Journal, IOS Press*, 3(3), 2012.
- [14] F. Giunchiglia, B. Crispo, and R. Zhang. Access control via lightweight ontologies. In *Semantic Computing (ICSC), 2011 Fifth IEEE International Conference on*, page 352, 2011.
- [15] F. Giunchiglia, U. Kharkevich, and I. Zaihrayeu. Concept search. In *ESWC 2009*, pages 429–444, 2009.
- [16] F. Giunchiglia, M. Marchese, and I. Zaihrayeu. Towards a theory of formal classification. In *AAAI-05 Workshop on Contexts and Ontologies: Theory, Practice and Applications (C&O-2005)*, 2005.
- [17] F. Giunchiglia, M. Marchese, and I. Zaihrayeu. Encoding classifications into lightweight ontologies. In *ESWC 2006*, page 80–94, 2006.
- [18] F. Giunchiglia and R. Sebastiani. Building decision procedures for modal logics from propositional decision procedures: The case study of modal $k(m)$. *Information and computation, Academic Press*, 162(1/2):158–178, 2000.
- [19] F. Giunchiglia and I. Zaihrayeu. Lightweight ontologies. *Encyclopedia of Database Systems*, pages 1613–1619, 2009.
- [20] Mark Hartswood, Marina Jirotko, Ronald Chenu-Abente, Alethia Hume, Fausto Giunchiglia, Leonardo A. Martucci, and Simone Fischer-Hübner. Privacy for peer profiling in collective adaptive systems. In *Privacy and Identity Management for the Future Internet in the Age of Globalisation*, volume 457 of *IFIP Advances in Information and Communication Technology*, pages 237–252. Springer Berlin Heidelberg, 2015.
- [21] MohammadBadiul Islam and Renato Iannella. Privacy by design: Does it matter for social networks? In Jan Camenisch, Bruno Crispo, Simone Fischer-Hübner, Ronald Leenes, and Giovanni Russello, editors, *Privacy*

and Identity Management for Life, volume 375 of *IFIP Advances in Information and Communication Technology*, pages 207–220. Springer Berlin Heidelberg, 2012.

- [22] A.N. Joinson. ‘looking at’, ‘looking up’ or ‘keeping up with’ people? motives and uses of facebook. In *CHI 2008*, 2008.
- [23] M. Kosinski, D. Stillwell, and T. Graepel. Private traits and attributes are predictable from digital records of human behavior. *Proceedings of the National Academy of Sciences*, 110(15):5802–5805, March 2013.
- [24] C. Lampe, R. Wash, A. Velasquez, and E. Ozkaya. Motivations to participate in online communities. In *CHI 2010*, 2010.
- [25] M. Menendez, A. De Angeli, and Z. Menestrina. Exploring the virtual space of academia. In *From Research to Practice in the Design of Cooperative Systems: Results and Open Challenges*, pages 49–63, 2012.
- [26] M. Motani, V. Srinivasn, and P.S. Nuggehalli. Peoplenet: Engineering a wireless virtual social network. In *Proceedings of ACM MobiCom*, 2005.
- [27] M.M. Skeels and J. Grudin. When social networks cross boundaries: a case study of workplace use of facebook and linkedin. In *Proceedings of the ACM 2009 International Conference on Supporting Group Work (GROUP '09)*, page 95–104, 2009.
- [28] Reasoning About Theory Adequacy. A New Solution to the Qualification Problem. More evaluation of decision procedures for modal logics. *Fundamenta Informaticae*, 2/3/4:247–262, 1995.
- [29] Slim Trabelsi, Gregory Neven, and Dave Raggett, editors. *PrimeLife Public Deliverable D5.3.4 – Report on design and implementation*. PrimeLife Project, 20 May 2011.