Real-Time Data Aggregation in Distributed Enterprise Social Platforms

Keith Griffin[1], Maciej Dabrowski[2], and John Breslin[2]

[1] Cisco Systems Galway, Ireland
kegriffi@cisco.com
[2] National University of Ireland, Galway, Ireland
maciej.dabrowski@deri.org

Abstract. The widespread use of social platforms in contemporary organizations leads to the generation of large amount of content shared through various social tools. This information is distributed and often unstructured, making it difficult to fully exploit its value in enterprise context. While Semantic Web technologies allow for publishing meaningful and structured data, major challenges include: (1) real-time integration of distributed social data, and (2) content personalization to identify relevant pieces of information and present them to users to limit the information overload. This research in progress paper draws from an enterprise use case and discussed practices in real-time integration of social data in distributed social platforms. We propose to combine Semantic Web technologies with standardized transport protocols to provide efficient and open source layer for aggregation of distributed social data in an enterprise. We also show how our component can facilitate development of personalised social platforms.

Keywords: Enterprise Social Networks, Real-time systems, Personalization.

1 Introduction

The employees of contemporary enterprises are often distributed across departments and geographical locations, use different information systems, and offer a wide variety of skills and expertise to the organisation. In such environment, efficient delivery of relevant information to interested peers can be challenging. To help alleviate this problem, Semantic Web technologies [1] have gained popularity in the corporate world, enabling transparent provision of structured and meaningful data with the use of lightweight and standardized vocabularies such as FOAF[1] or SIOC[2], allowing to model enterprise data in RDF[3]. The combination of Semantic Web technologies with social platforms in organisations led to the "Social Semantic Enterprise" [13] that can profit from these trends by efficient, close to real-time aggregation of structured and meaningful data.

[1] http://foaf-project.org/
[2] http://sioc-project.org/
[3] Resource Description Framework: http://www.w3.org/RDF/

L.M. Camarinha-Matos and R.J. Scherer (Eds.): PRO-VE 2013, IFIP AICT 408, pp. 711–718, 2013.

Social platforms that enable personalized information access in an enterprise environment require an infrastructure and algorithms that address many complex challenges: (1) aggregation and integration of data from different information systems/social platforms in the IT landscape of the enterprise is required, (2) provision of new algorithms that are independent of the domain and/or the source of the data to allow efficient personalization, (3) effective presentation and feedback mechanism to improve acceptance and quality of delivered information. This paper focuses mainly on the first challenge, and discusses how the contribution presented here can facilitate solutions to the second and third problems. We argue that while Semantic Web technologies provide rich data integration capabilities [18], there is a need to ensure that the integration of information from the vital sources available in an organization can be performed in, or close to, real-time. This is especially relevant in large organizations, where social platforms can be deployed in different geographical localisations nevertheless they need to interact instantaneously. Current approaches may not only delay the information flow but also negatively impact overall efficiency in many scenarios.

This paper gives an overview of work focused on enabling real-time aggregation and integration of social content published through distributed social platforms across and beyond an organization. We discuss how Semantic Web technologies and real-time instant messaging protocols can be used for efficient and quick aggregation of content between distributed social platforms according to corporate information management policies and with the use of existing real-time communication protocols.

2 Distributed Social Platforms

Organizations build and maintain many information systems to manage large volume of content published and consumed by knowledge-intensive workers [8]. Such environments involve many actors sharing and consuming information within a large network that is often distributed across various departments (see Figure 1) or even geographically. This shift requires new approaches for delivery of timely and relevant information in a close-to-real-time manner across such peer-to-peer [12] networks. Many initiatives [24,5,11] focus on building collaborative tools (e.g. wikis) that combine the benefits of mass collaboration with the intrinsic qualities of peer-to-peer networks, such as scalability or fault-tolerance. Although knowledge workers utilize many collaboration tools (e.g. blogs, wikis), crucial information is often not managed effectively [16] what affects efficiency and generates additional spending. To address this concern, organizations attempt to sustain information exchange through utilization of social networking tools both internally and externally. Once the social platform is operational and social connections are established, it is important to gather and reuse information available in this network. Thus, a distributed social network requires efficient information aggregation and delivery tools to allow for timely updates and retrieval of relevant content [21]. To address this challenge, the Semantic Web practitioners propose to use RDF to capture social data [20] and integrate RDF content across the organization. This is possible thanks to tools that provide means for building scalable distributed RDF repositories based on P2P networks, for example RDFPeers [3]. Indeed, the growing amount of information available in the form of

RDF that needs to be accessible by distributed peers in an organizational network requires mechanism for distributed querying [15] or replication [17]. Consequently, there is a need to tackle many aspects related to the dynamism of RDF data ("dataset dynamics" [23]), including change management. These issues are addressed in approaches for synchronization of RDF data (e.g. RDFSync [22]) that model and distribute changes in RDF models to the peers in large networks.

Distributed social platforms pose challenges related to efficient aggregation and delivery of information to relevant employees. The generic categorizations of models for communication and content/event exchange in a distributed environment differentiates pull and push approaches. The pull model involves an initial request from the (active) client that is responded by a (passive) server and is one of the most commonly used communication patterns in distributed networks. Polling is a mechanism related to pull model, which relies on clients actively sampling the server status through repetitive requests. Polling is considered resource expensive and scales poorly [6] as frequent polling may lead to inefficient usage of resources, but infrequent requests "may result in delayed responses to critical situations" [6]. Further, many scenarios require asynchronous delivery of events for better performance and scalability. Long Polling, introduced to address these limitations, is an approach based on the request-response model in which the server holds the request open until the response is available (or when the set timeout is reached) [9]. In contrast to the pull approach, the push model assumes a passive client that is notified of the occurrence of specific events upon a subscription to the server. The publish/subscribe (PubSub) interaction paradigm exploits the push model as it enables agents to subscribe to a particular event (e.g. content update), and to receive asynchronous notifications from the server/publisher when the event occurs [7]. The advantages of the PubSub paradigm over the Polling approach lie in the optimization of the number of request, the required network traffic, and in the full decoupling in "time, space, and synchronization between publishers and subscribers" [7]. Although the push approaches are gaining more popularity, the tools built using the pull paradigm are prevalent (see [2] for a detailed discussion). However, with the expansion of the Semantic Web technologies, more focus is put on applications implementing the push interaction model.

The task of real-time synchronization of RDF data requires both protocols and RDF description formats that allow concise data modelling and generate minimum amount of network traffic. In the next section, we present the scenario based on the problems faced by organizations with distributed social platforms and the generic requirements for integration of semantic social content and discuss these formats.

2.1 Use Case

Real-time aggregation of information published through various social platforms and collaboration tools in an organization lead to a number of problems and technological challenges. This is resembled in the use case described in this article (see Figure 1). Andrew, Bob and Cecilia are knowledge workers employed by a large organisation, however they work in different departments (CTO, IT, and marketing respectively) and use different social tools and platforms. In the current environment, they often must follow updates through various collaboration tools (e.g. wikis, confluence,

enterprise microblogging) and corporate blogs of other co-workers to access content relevant to a given topic of their interest. While the use of RSS in this scenario is possible it implies regular querying of the information sources for updates, this approach does not allow for efficient integration of data across separated sub-divisional networks with restricted access policies. Finally, modifications of content that has already been distributed are not possible.

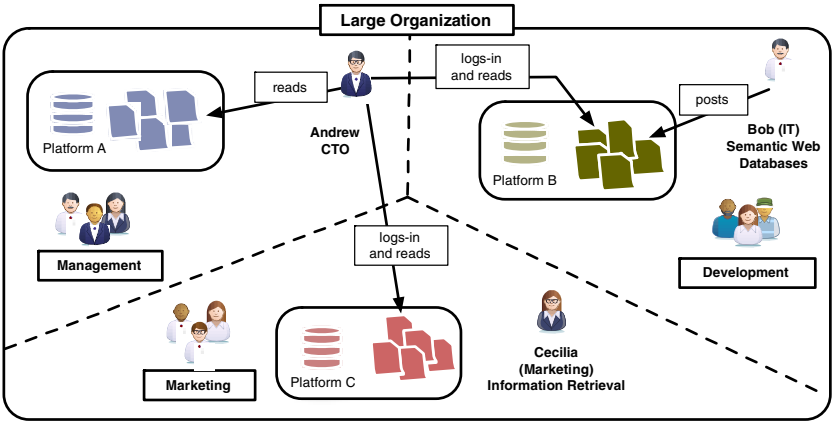

Fig. 1. Users publish large volumes of information through disconnected enterprise social platforms. This valuable information may be hard to discover by other users lowering the value of knowledge capital in an organization.

This example highlights the need for techniques enabling aggregation of social data with operations on data with fine granularity. Therefore, we argue that operations on data such as *Create*, *Read*, *Update*, or *Delete* (CRUD) [10] should be possible to execute not only on the document, but as well on the single-statement level. The support for, and efficiency of, the execution of these operations form the basic requirements for social platforms aggregating data across distributed organizations in real-time. Further, social platforms should be capable of delivery of content to interested users independently of the platform they are using, as well as limit the content delivery to only subscribed users as well as through filtering. In the next section we give an overview of the existing RDF update formats.

3 Approach

3.1 Describing Social Data with Semantic Web

For some applications (e.g. social content), in particular when aggregation from various sources is required, content update can be represented as streams of RDF triples [19]. Many applications in the Enterprise 2.0 are more stateful (e.g. presence management), thus require not only addition of the new content but also deletion/editing of existing information. Thus, RDF data integration techniques deployed in Enterprise 2.0 platforms should support not only addition, but also

deletion and editing of existing content. Further, it is essential that the update operations should be done on the lowest possible level that is triple-level. Hence, we investigate a number of formats that provide social data modelling capabilities and fine-grained CRUD. Here we compare variety of existing formats including: SPARQL Update[4], Talis Changeset, the Graph Update Ontology[5], Guaranteed RDF Update format[6], and Sesame RDF transactions.

The results of our previous work [4] indicate that Changeset and GUO consume significantly more network resources than the efficient formats (SPARQL Update and GRUF). Talis Changeset required roughly twice more resources (a 70% increase in case of GUO) to remove a tag from the description of an already distributed post. However, when more data is concerned in a single request (e.g. distribution of a newly published post), SPARQL Update provides the highest efficiency, followed closely by GUO and GRUF. Overall, the results indicate that, for the data provided, SPARQL Update combines many advantages with very low network usage and flexibility and as a W3C standard, SPARQL Update seems an appropriate format for describing RDF updates for real-time synchronization of distributed platforms.

Table 1. A comparison of the selected push protocols

Characteristics	Bayeux/Comet	Websockets	SLAP	SUP	XMPP PubSub	PUSH
Transport Layer	HTTP	HTTP	UDP	HTTP(S)	TCP/XMPP	HTTP(S)
Category	Both	Fat ping	Light ping	Light Ping	Fat ping	Fat ping
Interaction Style	Long Polling	Push	Ping/Poll	Poling	Push	Push
Latency	Low	Min.	Low	Low	Min.	Min.
Secure	Yes	Yes	Yes	Somewhat	Yes	Yes
Notifications						

3.2 Aggregating Social Data

The RDF update formats require a particular communication protocol [19] to deliver content to appropriate recipients. For example, SPARQL/Update utilizes the SPARQL 1.1 protocol[7] (with a HTTP binding) for managing RDF graphs, Changesets messages are distributed using a HTTP-based protocol[8], and Sparql-PuSH [14] uses the PubSubHubbub protocol[9] to notify subscribers about content updates using HTTP POST. Although some protocols enable maintenance of sessions between endpoints willing to exchange data (e.g. Session Initiation Protocol[10]), most are connectionless and based on HTTP, with exceptions such as XMPP PubSub that uses TCP connections maintained between the requests. These characteristics impact reliability of protocols and/or their ability to handle notifications to subscribers temporarily

[4] http://www.w3.org/TR/sparql11-update/
[5] http://webr3.org/specs/guo/
[6] http://websub.org/wiki/GRUF
[7] http://www.w3.org/TR/2009/WD-sparql11-protocol-20091022/
[8] http://n2.talis.com/wiki/Changeset_Protocol
[9] http://code.google.com/p/pubsubhubbub/
[10] See RFC 3261 at http://www.rfc-editor.org/rfc/rfc3261.txt

offline or over unreliable networks. A detailed comparison of various features (see Table 1) suggests that reliability, efficiency, security and extensibility of XMPP PubSub combined with its adoption in contemporary enterprises make this protocol an ideal choice for aggregation of social content modelled in RDF. Thus, XMPP PubSub and SPARQL Update are selected as a solution to the above-mentioned problem.

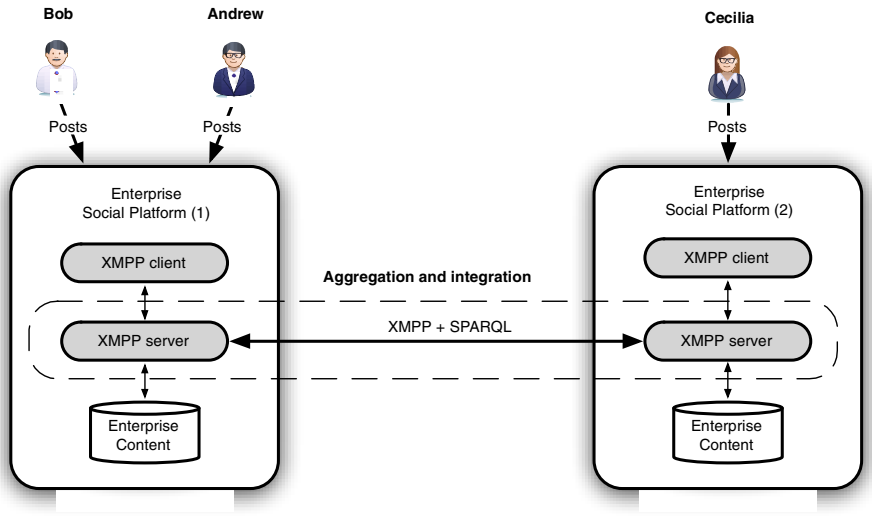

Fig. 2. Aggregation of content through distributed XMPP infrastructure

3.3 Proposed Solution

Based on the presented use case and the requirements we describe the architecture of the proposed solution. The XMPP server has the function of routing of the messages between the connected social platforms. In addition it provides an extensible platform that can process all information that it receives, for example to provide content personalisation through pluggable personalization components. As one or more platforms may be connected to a given XMPP server, the servers communicate with each other sending SPARQL Update messages embedded in XMPP PubSub stanzas. XMPP provides the infrastructure for connecting social platforms in a decentralised way and for sharing of knowledge between those social platforms. The server provides a central point for the connected social platforms to exchange XMPP messages. In case an existing XMPP server is used, it must implement the XMPP Publish-Subscribe extension (XEP-0060)[11], which allows XMPP clients to subscribe to updates. In order to publish and receive aggregated content social platforms need to connect to the XMPP server via the XMPP client, a task that can be accomplished using an open source component[12] that allows easy integration.

[11] http://xmpp.org/extensions/xep-0060.html
[12] https://github.com/derixmpppubsub/derixmpppubsub/

3.4 Personalization

Our approach is designed having in mind the need for personalisation in enterprise social platforms. Being aware of the overwhelming amount of content published daily within the organization, users of social platforms require personalised access to information to limit the information overload. First the recommendation approach should take into account data from multiple sources. In addition, the approach should be applicable to any kind of social content: blog posts, microblog posts, wiki pages or even office documents could be recommendable items. The recommendation results should take relations between concepts/terms into account, so that slightly different views on the same concept can be handled. Further, the cold-start problem related to the initial lack of data for computation of recommendation should be avoided (e.g. in collaborative filtering, items which did not get any user ratings should be available as part of the recommendations). Such requirements make the selection of appropriate personalisation algorithm very difficult. The proposed solution allows for design and deployment of various personalization components that extend the XMPP server either through plugin mechanism, or using external components that process received messages. Our initial experiments (not in scope of this paper) prove that this approach is scalable, efficient and flexible enabling close to real-time personalised aggregation of social content from distributed social platforms.

4 Discussion

This article discussed challenges for efficient aggregation of cross-domain, multi source content in distributed enterprise social platforms. We argue that while Semantic Web technologies allow for publishing meaningful and structured data, major remaining challenges include: (1) real-time integration of distributed social data, and (2) content personalization to identify relevant pieces of information and present them to users to limit the information overload. This research in progress paper draws from an enterprise use case and discussed practices in real-time integration of social data in distributed social platform. We proposed to combine Semantic Web technologies and standard protocols already used in organisations to deploy instant messaging solutions to provide efficient and open source layer for aggregation of distributed social content. We also show how our component facilitates development of personalised social platforms.

References

1. Berners-Lee, T., Hendler, J., Lessila, O.: The Semantic Web. Scientific American (May 2001)
2. Bhide, M., Deolasee, P., Katkar, A., Panchbudhe, A., Ramamritham, K., Shenoy, P.: Adaptive push-pull: Disseminating dynamic web data. IEEE Transactions on Computers 51, 652–668 (2002)
3. Cai, M., Frank, M.: Rdfpeers: A scalable distributed rdf repository based on a structured peer-to-peer network (2004)

4. Dabrowski, M., Griffin, K., Passant, A.: Approaches for real-time integration of semantic web data in distributed enterprise systems. In: 2012 IEEE Sixth International Conference on Semantic Computing, pp. 47–50 (2011)
5. Du, B., Brewer, E.A.: Dtwiki: a disconnection and intermittency tolerant wiki. In: Proceedings of the 17th International Conference on World Wide Web, WWW 2008, pp. 945–952. ACM, New York (2008)
6. Eugster, P.T., Guerraoui, R., Sventek, J.: Distributed Asynchronous Collections: Abstractions for Publish/Subscribe Interaction. In: Bertino, E. (ed.) ECOOP 2000. LNCS, vol. 1850, pp. 252–276. Springer, Heidelberg (2000)
7. Eugster, P.T., Felber, P.A., Guerraoui, R., Kermarrec, A.-M.: The many faces of publish/-subscribe. ACM Comput. Surv. 35, 114–131 (2003)
8. Gold, H., Malhotra, A., Segars, A.H.: Knowledge management: An organizational capabilities perspective. J. Manage. Inf. Syst. 18, 185–214 (2001)
9. Griffinand, K., Flanagan, C.: Evaluation of asynchronous event mechanisms for browser-based real-time communication integration. In: Elleithy, K., Sobh, T., Iskander, M., Kapila, V., Karim, M.A., Mahmood, A. (eds.) Technological Developments in Networking, Education and Automation, pp. 461–466. Springer, Netherlands (2010)
10. Martin, J.: Managing the database environment Martin. Prentice Hall (1999)
11. Mukherjee, P., Leng, C., Schurr, A.: Piki - a peer-to-peer based wiki engine. In: IEEE International Conference on Peer-to-Peer Computing, pp. 185–186 (2008)
12. Oram (ed.): Peer-to-Peer: Harnessing the Power of Disruptive Technologies. O'Reilly & Associates, Inc., Sebastopol (2001)
13. Passant, A.: Semantic Web Technologies For Enterprise 2.0. IOS Press (2010)
14. Passant, A., Mendes, P.N.: sparqlpush: Proactive notification of data updates in rdf stores using pubsubhubbub. In: 6th Workshop on Scripting and Development for the Semantic Web (2010)
15. Quilitz, B., Leser, U.: Querying distributed rdf data sources with sparql. In: Bechhofer, S., Hauswirth, M., Hoffmann, J., Koubarakis, M. (eds.) ESWC 2008. LNCS, vol. 5021, pp. 524–538. Springer, Heidelberg (2008)
16. Sabherwal, R., Becerra-Fernandez, I.: Business Intelligence: Practices, Technologies, & Management. Wiley (March 2010)
17. Schandl, B., Zander, S.: A framework for adaptive RDF graph replication for mobile semantic web applications. Technical report, University of Vienna University of Vienna, Department of Distributed and Multimedia Systems (2009)
18. Shadbolt, N., Hall, W., Berners-Lee, T.: The semantic web revisited. IEEE Intelligent Systems 21(3), 96–101 (2006)
19. Shinavier, J.: Optimizing real-time rdf data streams. CoRR, abs/1011.3595 (2010)
20. Shinavier, J.: Real-time semantic web in <=140chars. In: Proceedings of the Third Workshop on Linked Data on the Web (LDOW2010) at WWW (2010)
21. Tramp, S., Frischmuth, P., Arndt, N., Ermilov, T., Auer, S.: Waeving a distributed, semantic social network for mobile users. In: Antoniou, G., Grobelnik, M., Simperl, E., Parsia, B., Plexousakis, D., De Leenheer, P., Pan, J. (eds.) ESWC 2011, Part I. LNCS, vol. 6643, pp. 200–214. Springer, Heidelberg (2011)
22. Tummarello, G., Morbidoni, C., Bachmann-Gmür, R., Erling, O.: Rdfsync: efficient remote synchronization of rdf models. In: Aberer, K., et al. (eds.) ASWC 2007 and ISWC 2007. LNCS, vol. 4825, pp. 537–551. Springer, Heidelberg (2007)
23. Umbrich, J., Villazon-Terrazas, B., Hausenblas, M.: Dataset dynamics compendium: A comparative study. In: First International Workshop on Consuming Linked Data (2010)
24. Weiss, S., Urso, P., Molli, P.: Wooki: a p2p wiki-based collaborative writing tool. In: Benatallah, B., Casati, F., Georgakopoulos, D., Bartolini, C., Sadiq, W., Godart, C. (eds.) WISE 2007. LNCS, vol. 4831, pp. 503–512. Springer, Heidelberg (2007)