

CIMBA - Client-Integrated MicroBlogging Architecture

Andrei Vlad Sambra¹, Sandro Hawke¹, Tim Berners-Lee¹, Lalana Kagal¹, and Ashraf Abounaga²

¹ Decentralized Information Group,
MIT CSAIL

² Qatar Computing Research Institute
asambra@mit.edu, sandro@w3.org, timbl@w3.org, lkagal@csail.mit.edu,
aabounaga@qf.org.qa

Abstract. Personal data ownership and interoperability for decentralized social Web applications are currently two debated topics, especially when taking into consideration the aspects of privacy and access control. To increase data ownership, users should have the freedom to choose where their data resides and who is allowed access to it by decoupling data storage from the application that consumes it. Through CIMBA, we propose a decentralized architecture based on Web standards, which puts users back in control of their own data.

Keywords: decentralization, Linked Data, social Web, privacy, Web apps

1 Introduction

Recently, we have witnessed a dramatic increase in the number of social Web applications. These applications come in different forms and offer different services such as social networks, content management systems (CMS), bug trackers, blogging tools, or collaboration services in general.

A common practice, specific to most Web services, is to centralize user resources thus becoming so-called *data silos*. Often when adhering to online services people usually end up creating dedicated local accounts, which ties and limits users to particular services and/or resources. A solution to *data silos* can be achieved through decentralization, where users are free to host their data wherever they want, and then use *several* Web apps to consume and manage the data. In the following section we will discuss how our decentralized architecture plays an important role in achieving true data ownership for users.

2 Architecture

Today, more and more software is built around an application-specific back-end database. This makes switching applications problematic, since data are

structured according to each specific application and it only has meaning within the context of those applications. Moreover, this practice also forces a tight coupling between backends and applications consuming the data (cf. Fig.1 a).

The proposed architecture uses the Resource Description Framework (RDF) [1] to achieve greater interoperability between servers and applications as well as to ensure the data structure remains the same, regardless of the server on which the data are stored. In our case, CIMBA is a simple microblogging client that is completely decoupled from the backend, which in turn is a generic storage server (cf. Fig.1 b).

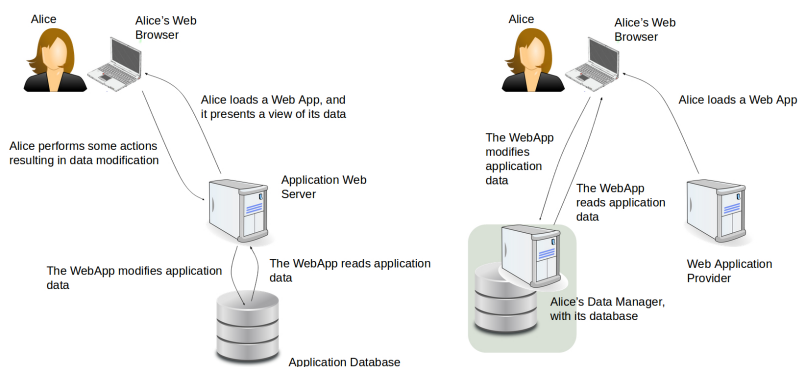


Fig. 1. a) Current architecture; b) Proposed decentralized architecture

By fully decoupling the server from the Web app, developers will be able to produce large scale Web apps without having to also manage the backend servers, making it very simple to switch from one backend to another, as well as from one Web app to another one without losing any data (cf. Fig.2 a).

Another advantage is that users are no longer locked into a *silo* because of their social connections (cf. Fig.2 b). Web apps reuse the user's social graph, which is also located on the data manager. The data manager is a generic Linked Data personal data server, which implements the Linked Data Platform specification [2] (currently on REC track at W3C³), as well as the Web Access Control [3] ontology (to enforce privacy policies).

Our architecture uses WebID [4] as the main mechanism to identify people at the Web scale, together with WebID-TLS [4] (to authenticate requests to restricted resources), a fully decentralized authentication scheme based on WebID.

³ <http://www.w3.org>

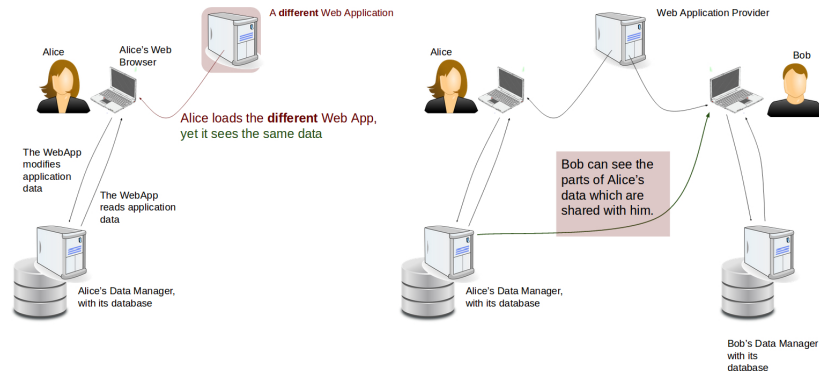


Fig. 2. a) users can easily switch software; b) social connections stay with the user

3 CIMBA

The name CIMBA stands for Client-Integrated MicroBlogging Application. It provides users with the power of having their own blog combined with the ease of using Twitter. CIMBA was written in Javascript, using the AngularJS⁴ framework. The source code is publicly available on GitHub⁵, and a running online demo can be accessed by visiting <http://cimba.co>.

Compared to Twitter, CIMBA users are not stuck with a single feed or timeline, but instead they can create multiple *Channels* and use them as categories for their posts (i.e. main, work, family, etc.). Access control can be set per channel as well as per post, though policies for posts will override those set per channel (i.e. having a private post in a public channel).

Figure 3 displays an overview of the architecture behind CIMBA. Users Alice and Bob each have their own personal data managers, which hold the posts data, configuration files as well as their personal WebID profiles.

Accessing CIMBA simply means loading all the necessary HTML, Javascript and CSS files from the application server into the user's browser. From that moment on, the application which now runs in the browser will communicate directly with the user's personal data manager. The location of the personal data manager is found by "faking" a WebID-TLS authentication process, with the purpose of finding the user's WebID and implicitly, the WebID profile. There is no actual need to authenticate the user to CIMBA, since all requests for data are authenticated by the user's personal data store.

Once the WebID profile is found, the app follows a series of links to discover useful information about the user, such as a generic Linked Data server from where the app can store and retrieve resources. CIMBA stores all the application data on the user's personal data manager, in a workspace dedicated to microblogging. Microblogging data are stored using the SIOC ontology.

⁴ <https://angularjs.org/>

⁵ <https://github.com/linkedata/cimba>

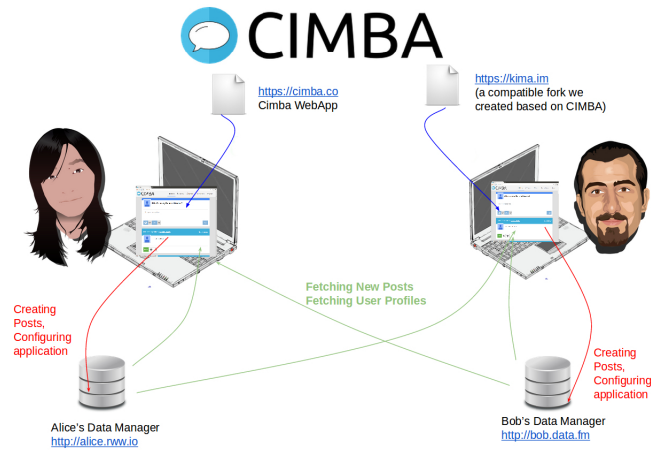


Fig. 3. Overview of the architecture for CIMBA

To read what other people write, users can subscribe to their channels. The list of subscriptions is also expressed using the SIOC vocabulary and it is stored on the user's server.

4 Conclusions and future work

Our proposed decentralized architecture offers significant benefits compared to current Web apps, both in terms of data ownership, privacy, as well as interoperability. Being fully decoupled from the backend, Web apps can be easily forked and improved by anyone with access to the source code, thus spurring innovation and creativity. At its current stage, CIMBA suffers from scalability issues, though we are very close to overcoming these issues.

References

1. Graham K., Carroll J., McBride B.: Resource description framework (RDF): Concepts and abstract syntax. In: W3C recommendation. (2004)
2. Speicher S., Arwe J., Malhotra A.: Linked Data Platform 1.0. <http://www.w3.org/TR/1dp/> (2014)
3. Hollenback J., Presbrey J., Berners-Lee T.: Using RDF metadata to enable access control on the social semantic web. In: Proceedings of the Workshop on Collaborative Construction, Management and Linking of Structured Knowledge (CK2009), vol. 514. (2009)
4. Sambra A., Henry S., Berners-Lee T.: WebID Specifications. <http://www.w3.org/2005/Incubator/webid/spec/> (2014)
5. Breslin J.G., Harth A., Bojars U., Decker, S.: Towards semantically-interlinked online communities. In: The Semantic Web: Research and Applications, pages 500-514. (2005)