

# Ajax for Mobility: MobileWeaver Ajax Framework

Louenas Hamdi

SAP Research, SAP Labs Canada  
111 Duke st. suite 1500,  
Montreal, QC, Canada  
15145763181

[louenas.hamdi@sap.com](mailto:louenas.hamdi@sap.com)

Huaigu Wu

SAP Research, SAP Labs Canada  
111 Duke st. suite 1500,  
Montreal, QC, Canada  
15148796743

[huaigu.wu@sap.com](mailto:huaigu.wu@sap.com)

Serhan Dagtas

Dept. of Information Science  
2801 S. University Dr.  
Little Rock AR 72204  
15016837267

[serhan\\_dagtas@yahoo.com](mailto:serhan_dagtas@yahoo.com)

## ABSTRACT

This paper provides a brief description of a research project on using Ajax to enhance mobile Web applications for enterprise use. The project, known as MobileWeaver Ajax Framework, leverages enterprise SOA (Service Oriented Architecture) and the latest web technologies on mobile devices.

## Categories and Subject Descriptors

H.5.4 [Information Interfaces and Presentation]

Hypertext/Hypermedia: Architectures. H.5.3 [Information Interfaces and Presentation] Group and Organization Interfaces: Web-based interaction

## General Terms

Performance, Design, Languages.

## Keywords

Enterprise SOA, Web 2.0, Mobility, Ajax, SOAP, REST.

## 1. INTRODUCTION

Traditionally, developing Web applications and consuming web services on mobile devices is very hard or impossible. Nowadays, web browsers and internet connections that these devices are equipped with are increasingly used to make the development and consumption of mobile services much easier. However, mobile Web remains very limited in terms of functionality and usability. The web experience is very frustrating for most of the mobile users.

Ajax is one of the delivery mechanisms of mobile Web2.0 that leverages existing Web standards such as HTTP, XHTML, XML, CSS, DOM and the asynchronous nature of XHR (XmlHttpRequest) to avoid whole web page refreshes.

The main objective of our research work is validating some concepts related to mobile Web architectures in the business context. The ultimate goal is to define an architecture that can reduce the load on the Web servers and ease the consumption of web services on mobile devices. This is achieved by leveraging the processing and communication capabilities of current mobile browsers while providing an acceptable user experience.

First, we will present a brief overview of Mobile Ajax followed by a description of our MobileWeaver Ajax framework and its components. We will provide some future research directions in the last section.

Copyright is held by the author/owner(s).  
WWW 2008, April 21–25, 2008, Beijing, China.  
ACM 978-1-60558-085-2/08/04.

## 2. MOBILE AJAX

The basic technology in Mobile Ajax is not much different from desktop Ajax. Basically, XHR or an equivalent ActiveX component is used to handle asynchronous communications. However, the way Ajax is used in mobile environments could be different from the way it is used for the desktop. On the desktop, Ajax is mainly associated with user experience using partial updates, drag and drop, type ahead, etc. In addition to the user experience enhancements, Ajax provides an answer to a basic need related to communication management on cellular networks and data management.

Because of the latency and perishable nature of the mobile and wireless networks, Ajax can be of great help in managing and optimizing the data exchange between the mobile application and the enterprise system. Mobile Ajax applications can handle asynchronous messaging and possibly client side business logic for offline processing during periods of disconnection.

## 3. MOBILEWEAVER AJAX FRAMEWORK

MobileWeaver Ajax is an end-to-end infrastructure for composing and consuming mobile Ajax applications that are connected to SAP or third party Web services using either SOAP or other protocols such as XML or JSON. In addition to the portability issue among different Ajax enabled browsers, MobileWeaver Ajax provides transparent client runtime and middleware components to facilitate SOA consumption on mobile device browsers. MobileWeaver Ajax provides a simple method for reducing SOAP messages and mitigating the network latency through pre-fetching, batching and client-side processing.

MobileWeaver uses XML to describe the applications and uses JSON for message exchange.

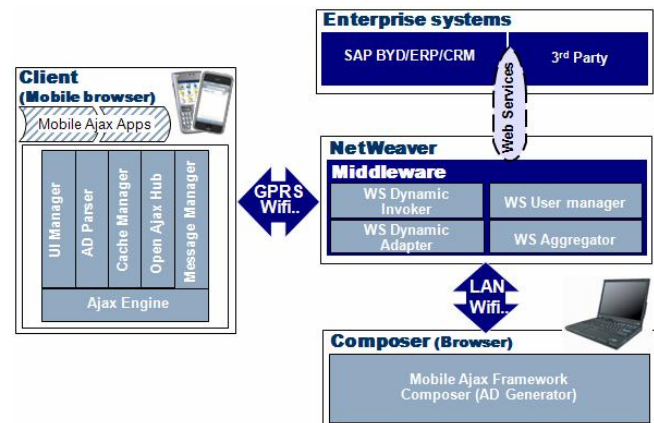


Figure 1. MobileWeaver Ajax Framework architecture.

The main components of MobileWeaver Ajax framework, illustrated in Figure 1, will be described in detail in the following sections.

### 3.1 MobileWeaver Ajax Composer

MobileWeaver composer is a completely browser-based environment for mashing-up multiple Web services and binding them visually to mobile forms. The forms are also designed in a drag-and-drop fashion. When the composition step is completed, the composer generates an Application Model (AD) and deploys it into the user's work area. The application model contains a description of the UI and configuration data to be used during runtime for aggregation and reduction of web services.

Figure 2 shows a screenshot of the MobileWeaver Ajax composer in its current state.

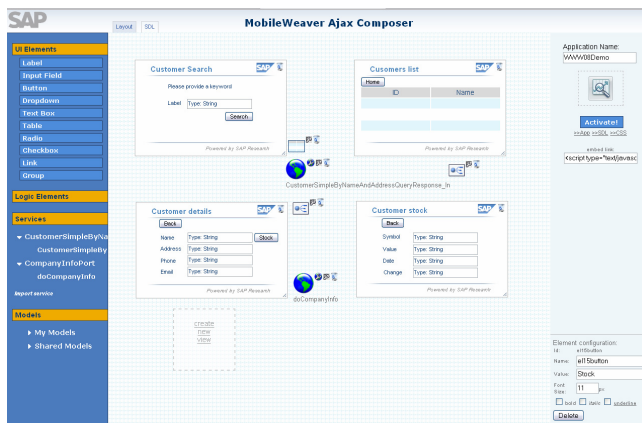


Figure 2. MobileWeaver Ajax Composer.

### 3.2 MobileWeaver Ajax Middleware

The middleware consists of four core components:

#### 3.2.1 Web Service Dynamic Invoker

This component is responsible for loading WDSL files and creating dynamic proxy clients automatically, which are used to invoke the remote Web service and receive the results as generic Java objects.

#### 3.2.2 Web Service Dynamic Adapter

Based on the configuration which was designed during the composition phase, the Web service dynamic adapter takes the Web service response object and reduces it to optimized messages following a JSON format. The resulting JSON message contains only filtered information.

#### 3.2.3 Web Service User Manager

The Web Service User Manager stores the login information that the business users provide to login to the participating web services. The login component is important because the participant web services can come from different systems including the freely available ones on the internet.

#### 3.2.4 Web Service Aggregator

This component is used to aggregate different Web services and provide a single output. This component is used in conjunction with Web Service Dynamic Adapter. Aggregating web services may seem straightforward; however, many challenges need to be

addressed in the upcoming months. The major challenges are related to visual composition of business logic concurrency and transaction management.

### 3.3 MobileWeaver Ajax Client

#### 3.3.1 Ajax Engine

This component is a set of JavaScript utilities used to abstract the specificities of mobile browsers and provide them as a unified and browser-independent set of interfaces.

#### 3.3.2 Application Model parser

The Application Model Parser is a JavaScript parser that is capable of transforming MobileWeaver Ajax Framework application models into an xHTML and JavaScript code. The parser generates xHTML pages, generates the client side validation code and the necessary bindings between the UI components and the remote services.

#### 3.3.3 Open Ajax Hub

OpenAjax Hub is an implementation of the Pub/Sub specification of OpenAjax organization. The OpenAjax Hub allows client-side JavaScript components to collaborate in a loosely coupled fashion through an event management mechanism.

#### 3.3.4 Cache Manager

The Cache Manager is a JavaScript component responsible for holding and indexing JSON objects. The technique used to organize the cache follows REST architecture URIs and uses a timestamp. The cache manager attempts to clean the cache every time a new append is requested.

#### 3.3.5 Message Manager

The messages manager is used for managing the XHR object and for queuing and batching asynchronous requests. Queuing is used in case of disconnection. If the browser gets disconnected for any reason, the Message Manager can queue the user requests for future trials.

## 4. FUTURE WORK

In the near future, we plan to assess and measure the usability of our framework against real customer needs. We also plan to test the framework on a number of different smart-phones and cellular networks and adapt the architecture consequently.

## 5. ACKNOWLEDGMENTS

Special thanks to Ris Stephane, Andreas Goeb and Daniel Koelsch, Christian Eck and Jean-Francois Desjeans Gauthier. Thanks also to our fourth author Abdel Benharref.

## 6. REFERENCES

- [1] Business case for Mobile Ajax.  
<http://www.w3.org/2007/06/mobile-ajax/papers/sap.hamdi.pdf>
- [2] Data Management for Mobile Ajax Web 2.0 Applications.  
<http://www.springerlink.com/content/b21k0th14m243lt5/>
- [3] What does AJAX mean for you?  
<http://portal.acm.org/citation.cfm?doid=1288515.1288523>
- [4] Open Ajax. <http://www.openajax.org>