A SCHEME OF STRATEGIES FOR REAL-TIME WEB COLLABORATION BASED ON AJAX/COMET TECHNIQUES FOR LIVE RIA

Walter Balzano, Maria Rosaria Del Sorbo and Luca Di Liberto
Istituto Nazionale Fisica Nucleare, Università degli Studi di Napoli Federico II, via Cinthia, 4 – 80126, Napoli, Italy

Keywords: AJAX, Comet, Web collaboration, Real-time, Data fidelity, DWR, Reverse AJAX.

Abstract: The last web applications advances, even if significant, doesn't yet allow to substitute desktop applications: data are often redundant, user/web interfaces interactions notably differ from the user/desktop ones, data propagation isn't fully instantaneous. This work shows some strategies for real-time client/server and client/client communications by server to manage multiuser collaboration issue in innovative ways: the user can use two techniques AJAX (Garrett J., 2005) and Comet (Russell A., 2006), recent and standard based, without installing plug-ins. Web 2.0 (O'Reilly T., 2007) was this project's leading philosophy, by which a mash up of two features was mainly realized: a chat module and a WYSIWYG (What You See Is What You Get) text editor. The results show a potential cooperation among users interacting simultaneously using simple graphical interfaces, with a large work speeding up. Many applications exploit client/server interactions, directly sending updated information to connected users, as in online auctions (Wellman M. et al., 1998) multiuser cooperation (Tapiador A. et al., 2006) and e-learning (T. Chan, 2007) systems. Further details about the code and simple demos are available on http://people.na.infn.it/~wbalzano/AJAX.

1 INTRODUCTION

Broadband communicating systems and participation architecture give users deeper and more active interaction with the web as in Wikipedia (Lih A., 2004), Amazon (Linden G., et al 2003), Google Maps (Markoff J., 2005), (Tulloch D. L., 2007).

AJAX and Comet (Bozdag E. et al., 2007) techniques improved client/server communication so as user/web interface interaction (Smullen III C. W. et al., 2008), (Smullen C.et al., 2007), allowing real-time client/server interaction where users can trade information instantly as in Google Docs (Google, 2007) beta application. Multiuser cooperation and support tools for formerly cited techniques are shown, specially DWR (http://getahead.org/dwr, 2008) and Jetty (Jetty Web Server, 2003).

This work will show recent applications of the preceeding technologies, that offer a solution of frequent problems in web interactions and open new perspectives: users haven't to download plug-ins, somehow complex procedure when they aren't operating on their own PC, but they can directly achieve these services all over only using a browser.

Multiuser cooperation issue is argued to improve cooperation by chat and to speed up the work by a WYSIWYG text editor, with no instant messaging software or specific formal language. In this context data reliability and single user's data maintenance are key factors, and then critical information is properly spread to all members by Comet technique.

This paper is organized in two sections: in the first section, basic techniques of the model are illustrated, with their most relevant features, and client/server interaction is described, highlighting the aspects tied with real-time feature and some AJAX patterns; in the section two some projects are illustrated with the respective implementation aspects. These projects are compared with our proposal. Moreover our multiuser model and chosen strategies, support tools and future perspectives are shown in details.

The conclusions highlight the effectiveness of a synergic use of these techniques to improve some parameters of client/server communication, in the aim to allow a better interaction between clients via server in the real-time collaborative schema, exploiting just standard solutions.

Balzano W., del Sorbo M. and Di Liberto L. (2009).
A SCHEME OF STRATEGIES FOR REAL-TIME WEB COLLABORATION BASED ON AJAX/COMET TECHNIQUES FOR LIVE RIA.
In *Proceedings of the 11th International Conference on Enterprise Information Systems - Software Agents and Internet Computing*, pages 35-40
DOI: 10.5220/0001858100350040

2 CLIENT/SERVER ASYNCHRONOUS COMMUNICATION

2.1 AJAX/Comet

AJAX is a set of interconnected techniques applied to improve web applications. Data are retrieved using the `XMLHttpRequest` (http://www.w3.org/TR/XMLHttpRequest, 2008). Despite of the name, `XMLHttpRequest` object is often combined with JSON (JavaScript Object Notation), (http://json.org, 2008), and SVG (Scalable Vector Graphics).

Comet, word introduced by Alex Russell, leading developer at Dojo Toolkit, in communication style defined Push Technology, consists of a long-lived HTTP connection between client and server, always initialized by client by a handshaking but then left open so that the server might send data, when needed, even if user hasn't actually required them.

The persistent connection, oppositely, could lead to issues in the resources management as it creates a pending request between client and server to which a thread is always allocated. An Event Driven architecture is apt to face these problems, as SEDA (Welsh M. et al., 2001). The standard feature Continuations of Jetty web server is also exploitable, managing up to 10000 concurrent connections from 857 threads and reducing the memory stack from 694 MB to 57 MB.

2.2 Real-Time Client/Server Interactions

Synchronous client/server paradigm defined "click, wait and refresh" is frequent in web applications. But this scheme has some drawbacks as redundant data loading and blocked web interface. AJAX faces these problems producing two key benefits:

- interactivity between users and web interface, since even if the browser is making a request to the server, possibly hidden to the user, the user can interact with the web interface, so always available;
- reactivity in the communication, as only a section of the page is reloaded, that is only the data needed in response to the client.

The asynchronous interaction created by `XMLHttpRequest` object allows the user to perceive a greater efficiency, satisfaction and usually a richer experience of web utilization (White A., 2006).

The client/server communication can further improve through two patterns: Predictive Fetch (http://AJAXpatterns.org/Predictive_Fetch, 2008), to try to anticipate the actions of user himself, and Submission Throttling (http://AJAXpatterns.org/Submission_Throttling, 2008), to dispatch data to server in the background, and to manage these interactions opportune widgets or techniques as Yellow Fade are used. The main graphical schema is shown in fig. 1.

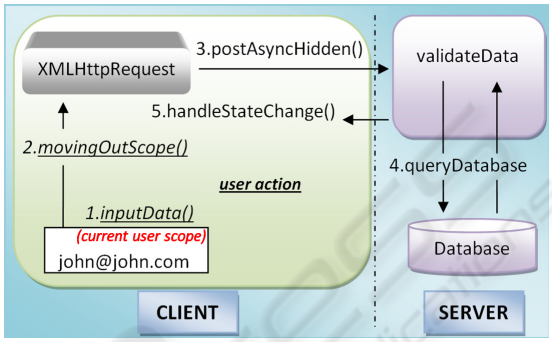


Figure 1: Real-time data validation made in background by the browser.

In the case of critical data update, the Comet technique allows a client/server communication perceivable as in real-time because data are exchanged among the various participants as soon as possible, without explicitly requiring refreshes of the page by client-side or cycles request by the browser to the server, leading to a useless network traffic. In figure 2 is illustrated an example of critical data update, regarded in our project. A client updating shared context by `XMLHttpRequest` object generates data sending from server to other clients and update of only his personal context, with a persistent client/server connection to obtain a better performance in communicating with one slow-start and fewer of RTT (Round Trip Time)

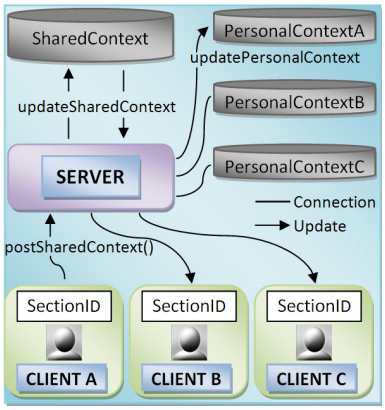


Figure 2: Critical data update in a Client/Server interaction schema.

3 MULTIUSER COOPERATION

3.1 A Comparative Analysis

In this section are recalled and compared only recent works, pertinent to real-time collaborative problems according to Web 2.0 style: particular attention has been paid to the updates diffusing procedures to users and the base techniques of applications.

UsaProxy 2 (Atterer R. et al., 2007) allows the vision and use of the same page or application by two different browsers at the same time, without installing client-side code. In particular it offers two modes: Monitoring, in which all user actions are communicated to the second user's browser that in this case can only assist, and Shared Browsing, in which the two browsers are locked together and each user can see other's actions which he may possibly influence and then even generate conflicts. The definite area we are interested in is the Shared Browsing, in which rather to see a remote pointer of the other user, the two users can also chat in a window. Unlike our project, collaboration is mutually exclusive, one user at a time can operate, and it is also mainly directed to a couple of users. The real-time data update between the two browsers is implemented through the polling technique and a chat window is placed above the webpage, hindering users a complete vision.

In (Gstöttner G., 2007) the author, member of the Austrian ISAC-team (ISAC-team, 2002) interested in systems for applied mathematics whose main area is e-learning, suggests to use AJAX techniques, with the aim to achieve interactive and responsive web applications, and Data Push method to send data from the server to the connected clients, by Polling and Comet. In this case different connected users can observe the computation progress but only one user can control and act. Comet technique is particularly indicated to send immediately the results of the computation to the connected users. In this article it is suggested the DWR framework to implement Comet technique.

In (Toffetti Carughi et al, 2007) the author focuses their attention on the modelling methods shortage for the push-enabled RIA characteristics, particularly suited for collaborative web applications, proposing an extension of a language to implement the different aspects of distributed communication such as synchronicity, persistence, and message filtering using a combination of primitive and patterns. The use case presented is the multiuser collaboration through a server and the methods to send and receive events triggered in response to user actions or directly by the server. To implement this model was used OpenLaszlo framework which led to client-side installation of a flash plug-in.

Google Docs represents a very famous application for real-time web collaboration according to Web 2.0 in Beta version. Also in this case the users share the project and receive simultaneous and regular updates.

As in (Atterer et al., 2007) we propose a chat module to support the communication between users, with the innovation of a proper space in web interface and as Schrottner we suggest Comet technique to inform users about critical data update mainly using DWR framework unlike the indications in (Toffetti Carughi et al, 2007).

Moreover in our project there are many other dissimilarities properly emphasized in the following description.

3.2 Our Approach: The Ajax/Comet Techniques for Live RIA

The aim of our project is to enable all and only users involved in the task to update every parts of work simultaneously whereas other users are enabled just to see the project in the last realised state.

Figure 3 shows an example of interaction by graphical interface. The Red user logins and accesses real-time update mode. His login is notified in the dashboard in both users and notifications sections. Green user has idRT as his scope, Smith user has idAC as his scope and Red, once logged in, has idChat as his scope. Subsequently Green user updates idRT, action reported in notifications, in the meantime Smith has changed scope, selecting idApp and Red is devoted to idAC.

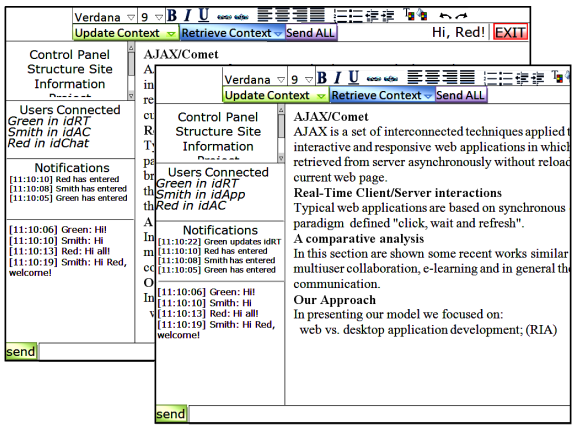


Figure 3: An effective chat example of RIA collaborative schema by Ajax/Comet technique.

```
public void updateMessage(String msg)  {
    ...
      messages.addFirst(new Message(msg));
      hours.addFirst(this.getHour());
      WebContext wctx = WebContextFactory.get();
      String currentPage = wctx.getCurrentPage();
      ScriptBuffer script = new ScriptBuffer();
      script.appendScript("displayMessages(")
     .appendData(nicks).appendScript(",")
           .appendData(hours).appendScript(",")
           .appendData(messages).appendScript(");");
      Collection pages = wctx.getScriptSessionsByPage(currentPage);
      for (Iterator it = pages.iterator(); it.hasNext();)
     { ScriptSession otherSession = (ScriptSession) it.next();
            otherSession.addScript(script); }
    ...
     }
```

Code Fragment 1: Java method invoked by a Javascript function for Return or Send button.

In order to present our model we focused on some important issues about the main idea of a new approach for collaborative AJAX/Comet techniques oriented to applications:

- desktop vs. web application development (RIA);
- real-time client/client communication (Live RIA);
- data fidelity.

RIA. According to the philosophy of Web 2.0 we have mainly conceived a mashup composed by two features, a chat module and a WYSIWYG (What You See Is What You Get) text editor. The chat module, realized using Comet technique, offers to users a straight communication in a web interface. A dashboard is added to the chat, allowing each user the sight of the scope of every other user: near the logged nick appears the id of the current working section to inform the users about the working state with elaboration notifications. A simple idea of the program code's kernel is shown in the following chunk of a representative Java method invoked by a Javascript function when an user sends a chat message by clicking Return or Send menu button. In this method the actual context is obtained by a feature of DWR toolkit and then Javascript function

`displayMessages(nicks,hours,messages)`

updates chat content.

The WYSIWYG text editor realized in the AJAX style facilitates the upgrade procedure usually divided into three steps, editing, preview and review and publication, in only one step which consists in editing and immediately seeing the final update's outcome.

Several operations are performed in background without user's explicit actions, other operations are executed by the server in response to some events caused by user's actions, always by an immediate Desktop like interaction as it happens with postAsynchHiddenE(id, event, url).

Live RIA. Following suggestions in (Bhide et al, 2002) for data characterized by a critical update in web applications, the data push technique implemented in the Comet model has been used, so as to immediately inform the different people involved: when a user causes an event enabling a data modification, first the current context with all the users involved in updating is available and then through Java methods and Javascript functions users of this context are updated.

In figure 4 is shown the main schema for a critical data update using Comet technique and allowing the server to send data to the client in a minimum update time.

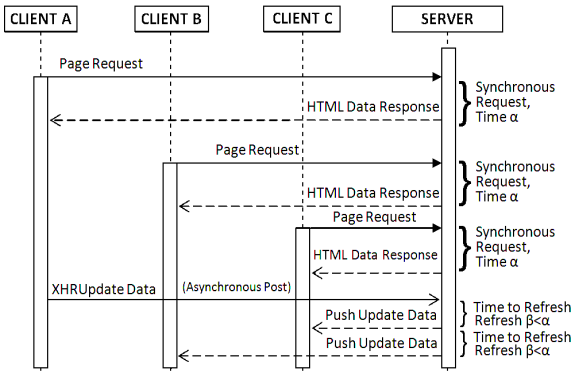
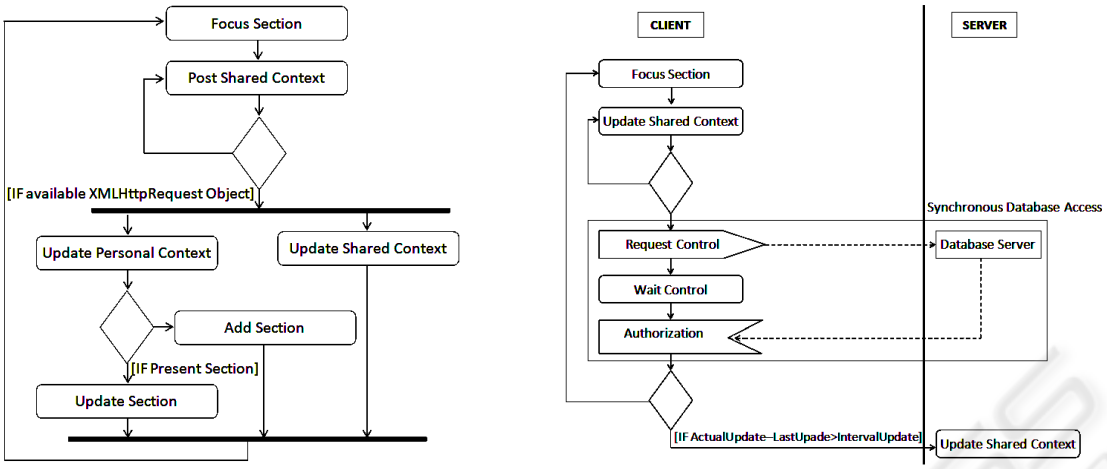


Figure 4: The main interaction client/server updating schema using Comet technique.



(a) Updating of shared and personal contexts.

(b) Conditional update of shared context.

Figure 5: Updating of context area of a web collaborative application.

An extra critical problem related a multiuser cooperation scenario with numerous concurrent users, is the timing management. The proposed solution was to offer a wide choice of action to users who can work on the same section changing it without the risk of losing their jobs if someone else terminates before and updates the context. In the following figure 5a is shown the reference schema for a data sending to the shared context: the main operations carried out involve the updating of shared and personal contexts. In figure 5b is represented an exemplifying schema of a conditional update: the application tries to update the shared context area using methods similar to conditional-get technique, i.e. comparing the last change time with the present change time. The updating occurs only if the observed period was greater than a specific IntervalUpdate.

Data Fidelity. The data shared between users are promptly updated and the work done by individual users is saved. If for any reason, a user does not receive the update from the shared context, he can still ask for it when he tries to update a section recently amended or when he reads processing notifications on the dashboard.

4 CONCLUSIONS

Thanks to AJAX/Comet techniques we can extend the range of many scenarios, from desktop to the web, as the case of multi-collaboration (Shen H. et al., 2007).

The interest for AJAX is increasing in different contexts, such as e-learning not only based on multimedia features, but even on these innovative techniques (Lin Y. et al., 2007). For example, in a scientific context, using AJAX and SVG excellent applications could be achieved, that graphically represent mathematic functions.

Comet technique is still in an initial phase and even if it presents some problems on the server side, it is fundamental for an immediate propagation of data to clients. There is an increasing interest about it by the web developers community because very probably the Push Technology represents a mode to spread updated data better then Polling. In particular, the Dojo Toolkit has realized the Bayeux Protocol (Davis D. et al., 2007) to make client/client communication via server more responsive.

Further features can be performed to improve real-time web collaboration as audio chat and operation log: in this way each user can better understand the current work state thanks to the cognition of the earlier states, because the user gets information of operations and their performers.

A real-time client/client communication via server realized by AJAX/Comet techniques involves important benefits, in particular no plug-in installation: users can exploit these web technologies in every computer connected to Net. Furthermore an implementation based on standard features allows developers to rely on normal support groups in the Web and more friendly with different browsers.

Strategies and techniques described lead to possibility to take advantage of different services everywhere according to Cloud Computing style.

REFERENCES

Atterer R., Schmidt A., Wnuk M., 2007. A Proxy-Based Infrastructure for Web Application Sharing and Remote Collaboration on Web Pages. In *Human-Computer Interaction - INTERACT 2007: 11th IFIP TC 13 International Conference, Rio de Janeiro, Brazil, Proceedings*, Springer.

Bhide M., Deolasee P., Katkar A, Panchbudhe A., Ramamritham K., Shenoy P., 2002. *Adaptive Push-Pull: Disseminating Dynamic Web Data*. IEEE Transactions on Computers, vol. 51, no. 6, pp. 652-668. IEEE Computer Society.

Bozdag E. et al., 2007. A comparison of push and pull techniques for AJAX. In *Proceedings of the 9th IEEE International Symposium on Web Site Evolution (WSE'07), pages 15–22. IEEE Computer Society*. IEEE Computer Society.

Chan T., 2007. Implementing Web 2.0 Technology for ELearning Systems. In *Proceedings of World Conference on E-Learning in Corporate, Government, Healthcare, and Higher Education*. G. Richards Editor.

Davis D., Russell A., Wilkins G. and Nesbitt M., 2007. *Bayeux Protocol -- Bayeux 1.0draft1*. The Dojo Foundation.

Garrett J., 2005. *AJAX: A new approach to web applications. Adaptive path*. http://www.adaptive-path.com/ideas/essays/archives/000385.php.

Google, 2007. *Google docs and spreadsheets*. http://docs.google.com.

Gstöttner G., 2007. *Feasibility study: Communication of a web 2.0 browser with an interactive mathematics server*. Master's thesis, IICM, Institute for Informations Systems and Computer Media, A-8010 Graz. http://AJAXpatterns.org/Predictive_Fetch, 2008.

http://AJAXpatterns.org/Submission_Throttling.

http://getahead.org/dwr, 2008.

http://json.org, 2008.

http://www.w3.org/TR/XMLHttpRequest, 2008.

ISAC-team, 2002. *User requirements document*. Technical report, Institute for Software technology, University of Technology, Graz. http://www.ist.tugraz.at/projects/

isac/publ/urd.pdf

Jetty Web Server and Servlet Container, 2003. http://jetty.mortbay.org.

Lih A., 2004. Wikipedia as participatory journalism: reliable sources? *In Proceeding of 5th International Symposium on Online Journalism, Austin, Texas*. http://jmsc.hku.hk/faculty/alih/publications/utaustin-2004-wikipedia-rc2.pdf

Lin Y. et al., 2007. A Web 2.0 Synchronous Learning Environment Using AJAX. *Ninth IEEE International Symposium on Multimedia Workshops*. IEEE

Linden G., Smith B., York J., 2003. *Amazon.com Recommendations: Item-to-Item Collaborative Filtering*. IEEE Internet Computing, vol. 07. IEEE.

Markoff J., 2005. *Marrying maps to data for a new web service*, New York Times (18 July), Technology section, p. 1.

O'Reilly T., 2007. *What is Web 2.0: Design Patterns and Business Models for the Next Generation of Software*. Communications & Strategies, No. 1, p. 17, First Quarter 2007. http://ssrn.com/abstract=1008839.

Russell A., 2006. *Comet: Low latency data for the browser*. http://alex.dojotoolkit.org/?p=545.

Shen H., Yang Z., and Sun C., 2007. *Collaborative Web Computing: From Desktops to Webtops*, IEEE Distributed Systems Online, vol. 8, no. 4.

Smullen C., Smullen S., 2007. AJAX Application Server Performance. Proceedings of the IEEE SoutheastCon (CH37882), Richmond, Virginia. IEEE.

Smullen III C. W., Smullen S.A., 2008. *An Experimental Study of AJAX Application Performance*. Journal of software. Academy Publisher.

Tapiador A., Fumero A., Salvachua J., Aguirre S., 2006. A Web Collaboration Architecture. *Proceedings of the 2nd IEEE International Conference on Collaborative Computing: Networking, Applications and Worksharing*. Atlanta, Georgia, USA. IEEE.

Toffetti Carughi G., Comai S., Bozzon A., Fraternali P., 2007. Modeling Distributed Events in Data-Intensive Rich Internet Applications. In *Proceedings of Web Information Systems Engineering 2007, LNCS 4831*. Springer.

Tulloch D. L., 2007. *Many, many maps: Empowerment and online participatory mapping*. First Monday, vol. 12, n. 2 . http://firstmonday.org/issues/issue12_2/

tulloch/.

Wellman M. and Wurman P., 1998. Real time issues for Internet auctions. *First IEEE Workshop on Dependable and Real-Time E-commerce Systems*. IEEE.

Welsh M., Culler D.E., and Brewer E.A., 2001. SEDA: An Architecture for Well-Conditioned, Scalable Internet Services, *Proceedings of 18th ACM Symposium on Operating Systems Principles*. ACM Press.

White A., 2006. *Measuring the Benefits of AJAX* http://www.developer.com/xml/article.php/ 3554271.