Mobile Ajax Position Paper

Kimmo Raatikainen

University of Helsinki and Helsinki Institute for Information Technology

kimmo.raatikainen@{hiit.fi,cs.helsinki.fi}

Although "One Web" advocates require that there should not be any mobility specifics in the Web, the past experiences have clearly demonstrated that direct port of technologies designed and developed for desktop clients and fast reliable (i.e. fixed network) connectivity to a mobile computing and communications environment leads to unacceptable end-user experience. This does not mean that mobiles devices and nomadic users would need a Web of their own. Instead, what is needed, is a Web infrastructure that can adequately support mobility as well as constraint computing and communication capabilities of carry-on or hand-held devices.

The infrastructure must be able to hide all mobility specifics and to adequately adapt to the available capabilities if the application does not want take them into account. However, the isolation should not be the only option. If an application wants to take mobility and capability constrains into account, the infrastructure must provide APIs to retrieve current context information and to express the user preferences.

Helsinki University Computer Science Department and Helsinki Institute for Information Technology [1] have a long (15 years) experience in mobile computing and communication research. We have contributed to IETF (TCP enhancements and QoS mechanisms for wireless links as well as HIP or Host Identity Protocol), to OMG (Wireless Access and Terminal Mobility in CORBA), FIPA (Efficient Agent Communication), and W3C (XBC and EXI WGs, MWI). The prototype implementations are available as Open Source Software.

In our vision efficient XML messaging and processing are the key elements of Mobile Ajax to deliver satisfactory end-user experience. W3C's EXI format [2] will be a crucial component of the "Mobile Ajax stack". Efficient XML synchronization, far beyond the capabilities of the current SyncML [3], is to be another important component of great importance. We also anticipate that SOAP may need specialized enhancements in addition to gains obtained by employing the EXI format.

If, as we hope, W3C starts work on Mobile Ajax, we will be an active participant. Our primary interest is in efficient XML synchronization. In the Fuego Core –project [4] at HIIT [1] have developed Fuego Mobility Middleware [5] that allows efficient XML messaging, processing and synchronization. The current prototype is available from Hoslab [6]. A forthcoming paper "***How to Edit Gigabyte XML Files on a Mobile Phone with XAS, RefTrees, and RAXS***" by Jaakko Kangasharju and Tancred Lindholm (both from HIIT) describes the "Fuego XML Stack" in details. Below we briefly summarize the key points.

The number of mobile phones and other limited devices on the Internet capable of hosting generic applications is rapidly rising, and may soon outgrow the number of regular computers. In terms of software development, these devices are challenging due to their significantly lower processing capabilities compared to desktop systems. Thinking that faster processors will make this go away is a fallacy: as these devices are battery-powered, saving CPU cycles translates into longer operation on a single charge, which is often a key selling point. Thus, in this domain, saving processing cycles is of paramount importance.

Storage on limited devices, on the other hand, is another business, as it typically does not consume any power when idle. During the recent years, we have seen rapid growth in storage capacity on limited devices. This has led to an increasing mismatch between storage and processing power. For instance, the mobile phone we use can accommodate a 1 GB XML file. However, it will take the phone some 9 hours just to parse that file. This is around 270 times slower than the same task on our current desktop PC, ***that is 2 min vs. 9 h!***

The Fuego XML stack, which is the collective name for the XML processing components in the Fuego middleware, provides efficient read and write random access to ordinary XML files, small and large alike. The distinguishing features of the stack are:

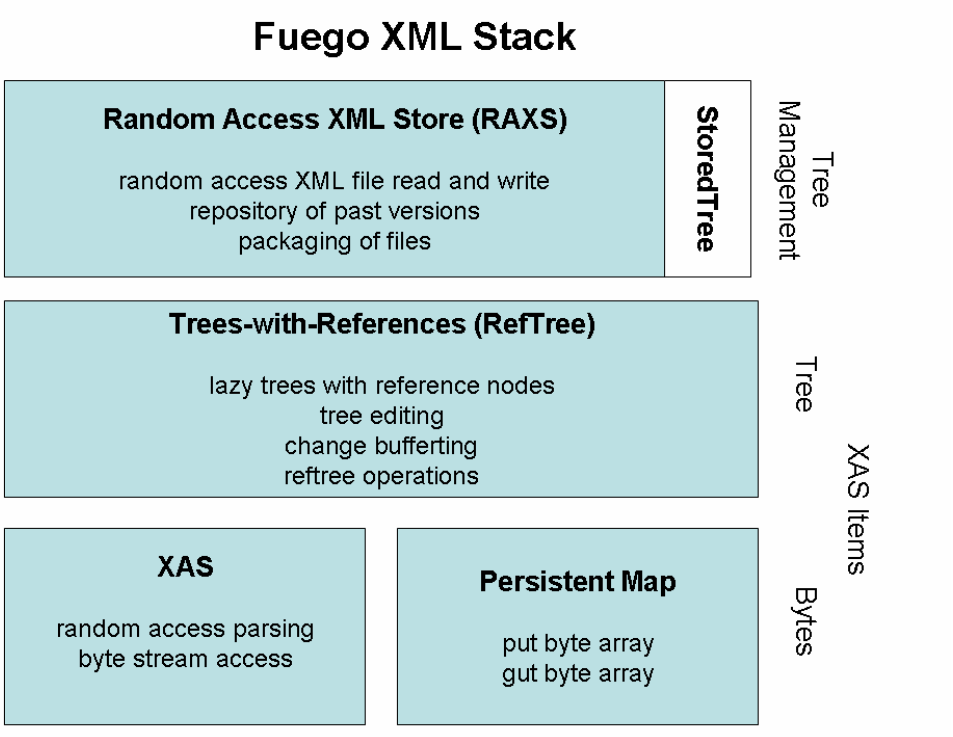
i) the ability to process significantly larger XML files on limited devices than what is commonly considered feasible, and

ii) extensive use of verbatim XML for on-disk storage.

Our stack enables application developers to use large XML data files on mobile devices, without format conversions, in a very interoperable and open manner.

The components of the stack (see the Figure below) are the XAS API for efficient XML parsing and serialization, the RefTree API for lazy XML tree manipulation, and the Random Access XML Store (RAXS) API for XML document management such as packaging, versioning, and synchronization. The efficiency of our stack stems from the use of lazy data structures and the advanced parsing and serialization capabilities of XAS.

When storing data on disk, we use the original XML in its verbatim format as far as possible. That is, we avoid having to import and export XML documents to and from a store format, and instead use the XML file directly. This way, our store is open to XML processing applications, and we can avoid costly and sometimes error-prone format conversion. We demonstrate the use of the stack by building a viewer and editor for Wikipedia XML dumps. We are currently able to successfully run the editor on a real mobile phone with a 1 GB XML file.



The Fuego XML Stack is open and can be used as a basis or a candidate for W3C standard APIs for Mobile Ajax libraries. To the best of our knowledge there are no IPR or patent issues.

References

[1] http://www.hiit.fi/

[2] http://www.w3.org/TR/2007/WD-exi-20070716/

[3] http://www.openmobilealliance.org/release_program/SyncML_v121A.html

[4] http://www.hiit.fi/fi/fc

[5] Sasu Tarkoma, Jaakko Kangasharju, Tancred Lindholm, and Kimmo Raatikainen. Fuego: Experiences with Mobile Data Communication and Synchronization. In 17th Annual IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC), September 2006. http://ieeexplore.ieee.org/search/wrapper.jsp?arnumber=4022787

[6] https://hoslab.cs.helsinki.fi/savane/projects/fuego-core/