

XapProposal

XAP - eXtensible Ajax Platform

An Ajax-based Rich Internet Application framework

April 2006, Coach Wei (cwei at nexaweb dot com)

PROJECT PROPOSAL

XAP is to provide an XML-based declarative framework for building, deploying and maintaining rich, interactive, Ajax-powered web applications. A basic principal of XAP is to leverage existing Ajax projects such as Apache Kabuki and Dojo, as well as other community efforts such as Eclipse openAjax. It aims to be pluggable with various Ajax toolkits, reduce the need of scripting and solve the development challenge as well as application maintenance challenges associated with Ajax programming.

0. Rationale

Ajax is a widely recognized approach for building richer, dynamic, more interactive web applications, known as "Rich Internet Application" (RIA). RIA combines the richness of desktop software with the universal deployment advantages of the web, representing the next evolution of application development.

With the recent excitement around Ajax, tremendous effort is underway, from many directions, to make the creation of RIAs easier. Toolkits that provide DHTML and [JavaScript](#) (Ajax) support are proliferating and provide a rich set of functionality allowing developers to quickly add dynamic features to web applications.

Although these toolkits ease development of Ajax-powered web applications, there are still significant development and maintenance challenges, mainly associated with writing, debugging and maintaining [JavaScript](#) code. In particular, some developers would prefer not to use [JavaScript](#).

There are many possible solutions to address these challenges and each approach can solve the challenges to a different degree. On the one side, open source community has responded with [Apache Kabuki](#) and Eclipse [openAjax](#) projects. On the other side, standard organizations have been working on standardizing Ajax(XmlHttpRequest for example) and related browser behaviors. A higher level abstraction via a declarative format helps solve the Ajax development challenges by reducing the need for scripting while providing a straightforward way of building rich user interfaces, but also provides a mechanism to combine the above mentioned community efforts together for greater benefits.

Building on these vibrant community efforts already underway, the proposed project is to create an extensible software framework for building and deploying Ajax-powered rich internet applications declaratively. The project is "pluggable" with client side Ajax toolkits such as Kabuki, works with openAjax as well as other Ajax initiatives. The mission of this project is to encourage innovation around declarative Ajax development, build a community, and provide an open source framework for building and deploying rich internet applications independent of browsers and servers.

This project seeks to provide

- XML markup that provides a declarative format for Ajax-based rich user interface, separating presentation and behavior;
- Declarative data binding that links user interface components with data sources, separating presentation and data;
- Declarative modification syntax for updating the user interface;
- A "bridge" mechanism that allows easy plug-in of existing UI Ajax toolkits.
- A framework that ties everything together; enabling asynchronous update and event handling;

The architecture of this proposal embraces existing Ajax toolkits, such as Apache Kabuki and Dojo. At the core of the architecture is the concept of "bridges" that allow declarative tags to be connected to any toolkit and allow the toolkits to be easily interchanged. It also supports "plugins", allowing tags to be easily added or extended to support custom widgets and behavior.

Key benefits:

- XML syntax describes UI instead of [JavaScript](#) code
- XML syntax describes asynchronous update instead of [JavaScript](#) code
- Built in support for data binding
- Enables tools that can generate XML to create the UI rather than [JavaScript](#)
- Simplifies application development and maintenance
- Extendable with various [JavaScript](#) UI toolkits via the "bridge" mechanism

0.1 Criteria

Meritocracy:

We plan to do everything possible to encourage an environment that supports meritocracy. We know that meritocracies don't just evolve from good intentions; they require actively asking the community for help, listing/specifying the work that needs to be done, and keeping track of and encouraging members of the community who make any contributions.

Community:

We are committed to building a strong community around the proposed project. The committers will supply example code and documentation to help bring new members up to speed as to the current functionality of the code and how it is organized and maintained. In addition, the committers plan to spend the time necessary to answer user and developer questions. Along with our plans to encourage meritocracy (mentioned above), we hope these efforts will eventually create a user and development community that can live beyond the contribution of any one person and beyond the goals of any contributor's employer.

Core Developers:

More than half of the initial committers are key members of Nexaweb's development, test, and project management team. The rest are developers who have had significant experience with the framework or the technologies it is built upon. Some of these committers are contributing to this project on behalf of their employer, some of them are self-employed consultants, and some of them are just contributing as individuals.

Alignment:

The project is a pure client-side implementation. It should support any server side infrastructure. The initial code base is targeted to support the following environments:

- Client side: Internet Explorer and Firefox initially. We expect support for other browsers in the future.
- Server side: any web server environment (Tomcat, Geronimo, Apache, .NET, PHP, etc.).

For further information, please see <http://www.nexaweb.com/open/xap> and <http://www.openxal.org>

0.2 Warning signs

Orphaned products:

The initial code contribution is being developed specifically for the developer community and is not an orphaned product. The committers have long term interest to develop and maintain the code.

Inexperience with open source:

Only a few of the initial committers have contributed to open source projects; however, all of the initial committers have been reading Apache process documents, the incubator general mailing list, and the dev lists of current Apache projects. We've also spent time with ASF members and at [ApacheCon](#) to prepare ourselves as much as possible.

Homogenous developers:

The initial list of developers consists primarily of paid employees of the donating company. The donating company has reached out and will continue to reach out to build a diverse and vibrant community. The remaining initial committers are independent. They have had experience with the technology before and are personally passionate about the technology.

The committers are geographically distributed. They are experienced with working in a distributed environment.

Reliance on salaried developers:

Some of the initial committers are salaried developers employed by [Nexaweb](#). Nexaweb is committed to open source and committed to building a community for this project. The remaining developers are individual volunteers who are passionate about the technology. The donating company has reached out and will continue to reach out in its effort to build a diverse community.

No ties to other Apache products:

This proposal is related to many ongoing projects at Apache, such as Kabuki and [MyFaces](#), and it fits into the overall vision of that set of projects. There is an optional dependency on Kabuki.

A fascination with the Apache brand:

The committers are intent on developing a strong open source community around the XAP framework whether Apache is the right place or not; however, we believe that the project's current use of Apache projects, the potential for future synergies, and the Apache way of developing software make the ASF the ideal host community.

1. Scope

Provide declarative syntax and framework for writing Ajax applications.

The initial commit will contain:

- XML processing abstraction
- XPath support - limited support initially
- Incremental Update - remove, set-attribute and append
- UI Component Library bridges

- Declarative framework
- Open XML specification and schema
- Samples and documentation

For further technical or background information, please see:

<http://www.nexaweb.com/open/xap> and <http://www.openxal.org/>.

2. Identify the initial source from which the subprojects are to be populated

The initial source will be denoted by Nexaweb Technologies Inc. The donating company will contribute the initial code base immediately after the proposal is accepted and necessary infrastructure has been set up. For further background or technical information, please see <http://www.nexaweb.com/open/xap> and <http://www.openxal.org> for more details.

2.1 External Dependencies of the project

None.

3. Identify the ASF resources to be created

3.1 mailing list(s)

- xap-ppmc (with moderated subscriptions)
- xap-dev
- xap-commits
- xap-user

3.2 Subversion repository

- <https://svn.apache.org/repos/asf/incubator/xap>

3.3 Jira

- xap

4. Identify the initial set of committers:

- Atsuko Pien
- Scott Boyd
- Robert Buffone
- Cliff Schmidt
- Coach Wei
- James Margaris
- Michael Turyn
- Jonathan Levin
- Peter Eacmen
- Animesh Kumar
- Doug Schepers
- Igor Kaplansky

5. Identify Apache sponsoring individual

Champion: Cliff Schmidt

Mentors: Cliff Schmidt, Susan Wu, Robert Burrell Donkin.

We request that the Apache Incubator PMC sponsor XAP as an incubating project. There is not currently another TLP that would be an obvious fit as sponsor for this project. As the project approaches graduation, we would reevaluate possible TLP destinations and work with others at Apache to consider whether a new TLP is warranted to include XAP and possibly other related Apache projects.