ChemistryProposal

Apache Chemistry

Abstract

Apache Chemistry is a generic Java language implementation of the upcoming OASIS CMIS specification.

Proposal

Apache Chemistry is an effort to provide a Java (and possibly others, like JavaScript) implementation of an upcoming CMIS specification, consisting of a high-level API for developers wanting to manipulate documents, a low-level SPI close to the CMIS protocol for developers wanting to implement a client or a server, and default implementations for all of the above. Chemistry aims to cover both the AtomPub and SOAP bindings defined by the CMIS specifications.

Background

The proposed CMIS (Content Management Interoperability Services) standard defines a domain model and set of bindings, such as AtomPub and SOAP that can be used by applications to work with one or more Content Management repositories and systems. The standard aims to help providing vendor-neutral formats to share information across Internet.

The submission of the specification was originally proposed by seven CMS vendors and since that a number of other vendors have joined the OASIS TC, both commercial and Open Source. It is now in an early definition phase (started in September 2008); improvement of specifications and compatibility among existing implementations are the short term goals for CMIS.

There has been considerable interest in a JCR-based implementation of this proposed standard, and some months ago a group of interested people started working on such a codebase in the Apache Jackrabbit sandbox. Write access to the Jackrabbit sandbox was opened to all Apache committers to lower the barriers for cooperation.

This work gave birth also to an effort at creating a more generic CMIS implementation framework. This effort, named "Chemistry", has been led by Florent Guillaume and other Nuxeo employees. The Chemistry work has taken place in a Mercurial source repository outside Apache, since the developers are not yet Apache committers and thus do not have write access to the Jackrabbit sandbox.

This situation with separate codebases in separate source repositories was causing confusion and coordination problems. The effort was clearly no longer just an extra component in Jackrabbit, and so it was proposed that the project should brought to the Incubator where it can grow in a more controlled and yet less constrained manner.

The project name went through a few iterations before reaching "Apache Chemistry" (based on CMIS, as in !CheMIStry). There were some concerns about that name being too "general" and conflicting with potential future chemistry projects in Apache, but we feel that these are mostly hypothetical issues. There is also a minor naming overlap with another Java project called "Chemistry Development Kit", but the developers of that project weren't too concerned about that so we think it's fine for us to use this name.

Rationale

More vendors moving to CMIS means more choice for customers who will be able to use different implementations. For the standard to succeed, however, ensuring interoperability is paramount: in order to manage an ever growing context and leverage the enormous portability and interoperability issues that a globally adopted Standard brings, it is necessary to think about how to make the related ecosystem healthy and sustainable. Successful modern standards are driven by:

- Clear documentation, built iteratively with continuous feedback from stakeholders
- A clearly defined compatibility process, enforced by tools that allow to gauge how implementations can be compatible and interoperable
- Accurate compliance criteria, documented in writing as well as in actual testing code that measure how tools and libraries are able to interoperate
- A reference implementation to clear up potential doubts and ensure that the standard can actually be implemented in real life scenarios

The above mentioned pieces are able to make the development activity, towards a CMIS implementation, easier and more successful. Having an healthy ecosystem will ensure a smoother implementation process, more compliant products and, ultimately, a wider adoption of the standard.

The CMIS ecosystem has been successful in creating and documenting early drafts the basic standard, yet it might potentially lack two very important components, that is a Reference Implementation (RI) and a Technology Compatibility Kit (TCK). As much as such artifacts can be developed independently by any CMIS proponent, the value of having a neutral party as a steward of actual code is to be considered. The Apache Software Foundation has been playing this kind of role for many years, and can provide the perfect environment to foster contributions on the CMIS theme with a great amount of expertise.

Initial Goals

- Implement the latest draft of the CMIS specification
- Provide input to the TC on the implementation details of the spec

Current Status

Meritocracy

All of the initial committers are open source developers and many of them have already worked in other Apache projects. We will follow standard Apache procedures in accepting new contributors to the project team.

Community

Managed by an OASIS Technical Committee, the CMIS standard definition should be based on the idea of a community driven effort. In the last months the lack of a development community around CMIS has showed up.

Apache Chemistry aims to be a community driven initiative in developing a Java CMIS implementation. Such an approach is allowing more transparency and direct feedback even within the definition and improvement of CMIS specifications.

We welcome everyone interested in CMIS to join the Chemistry effort.

Core Developers

The core developers of the Chemistry project have a strong background within the ECM industry. While a core structure of a CMIS sandbox has been set up by some Jackrabbit PMC members, there has been a very important contribution by a development team from Nuxeo, led by Florent Guilaume. The Nuxeo team has implemented a draft version of the project on a Nuxeo repository; such a version has been merged inside the Jackrabbit sandbox. Developers from Day and Sourcesense have contributed to the project since its early stages: Dominique Pfister (Day) and Paolo Mottadelli (Sourcesense), with some patches contributed by Gabriele Columbro (former Sourcesense and now Alfresco); also a few commits by Jukka Zitting (Day) are tracked.

Alignment

An ancestor for the project was built in the Apache Jackrabbit sandbox. One of the first backend to be written for Apache Chemistry will be one for JCR, using Apache Jackrabbit.

Chemistry also exploit capabilities from some other Apache projects. The AtomPub binding is built upon an Abdera extension covering specific CMIS elements. The SOAP binding takes advantage of CXF and its JAX-WS APIs as a services framework.

Chemistry will potentially establish an interesting relationship with every Apache project handling any sort of content. Web frameworks such as Cocoon, Struts, Wicket and Tapestry can find a good relation, as well as SOA frameworks or ESB including Tuscany and Synapse.

Known Risks

Orphaned products

Apache Chemistry is a fresh new codebase that targets the still moving CMIS standardization effort. Thus the future of this project depends heavily on the success of the standard. We hope and expect that our implementation efforts will strengthen and support the CMIS standard.

Inexperience with Open Source

All the initial committers are experienced open source developers.

Homogenous Developers

The initial committers work for a number of different companies, each with their own backgrounds and priorities.

Reliance on Salaried Developers

Many of the initial committers are paid to work on this or related projects.

Relationships with Other Apache Products

Chemistry will directly use at least the following projects:

- Abdera - for AtomPub bindings
- CXF - for SOAP bindings

Many other Apache projects may be interested in using Chemistry to add CMIS support once the standard is final.

A Excessive Fascination with the Apache Brand

We value Apache as a neutral place where diverse communities can work together on implementing shared standards. We hope that this part of the Apache brand helps attract contributions from many potential CMIS vendors. However, the brand value is not the main reason why we prefer to have this project at Apache.

Documentation

This project is still at an early stage, so there is not much documentation available. See the OASIS CMIS page for information about the CMIS standardization effort.

Initial Source

The initial source for the Chemistry project comes from the following two sources:

- The Chemistry codebase developed at Nuxeo (http://hg.nuxeo.org/sandbox/chemistry/)
- The JCR-CMIS sandbox components developed at Apache Jackrabbit (http://svn.apache.org/repos/asf/jackrabbit/sandbox/jcr-cmis/)

Source and Intellectual Property Submission Plan

- Nuxeo has already filed a software grant covering their Chemistry codebase
- All contributions to the JCR-CMIS components in Jackrabbit have been made by people with CLAs on file

External Dependencies

All the external dependencies of the initial codebases comply with Apache licensing policies.

A binding to the Metro web service stack is being planned for increased interoperability. The licensing implications of that dependency (CDDL+GPL) will be considered in more detail during incubation once the technical details are clearer.

Cryptography

Apache Chemistry is not expected to implement or use cryptographic code.

Required Resources

Mailing lists

- chemistry-dev@incubator.apache.org
- chemistry-commits@incubator.apache.org
- chemistry-private@incubator.apache.org

Subversion Directory

- https://svn.apache.org/repos/asf/incubator/chemistry

Issue Tracking

- JIRA CMIS (CMIS)

Other Resources

- none

Initial Committers

Name	Email	CLA
Florent Guillaume	fg@nuxeo.com	yes
Bogdan Stefanescu	bstefanescu@nuxeo.com	yes
Stéphane Lacoin	slacoin@nuxeo.com	yes
Sun Seng David Tan	stan@nuxeo.com	no
Gabriele Columbro	columbro@gmail.com	no
David Nuescheler	uncled@apache.org	yes
Paolo Mottadelli	paolo@apache.org	yes
Dominique Pfister	dpfister@apache.org	yes
Stéfane Fermigier	sf@nuxeo.com	no

Affiliations

Name	**Affiliation**
Florent Guillaume	Nuxeo
Bogdan Stefanescu	Nuxeo
Stéphane Lacoin	Nuxeo
Sun Seng David Tan	Nuxeo
Gabriele Columbro	Alfresco
David Nuescheler	Day
Paolo Mottadelli	Sourcesense
Dominique Pfister	Day
Stéfane Fermigier	Nuxeo

Sponsors

Champion

- Jukka Zitting

Nominated Mentors

- Jukka Zitting
- Felix Meschberger
- Gianugo Rabellino

Sponsoring Entity

- Apache Jackrabbit PMC