

# RaveProposal

## Apache Rave Proposal

### Status

The Rave Proposal has been [accepted](#)

Apache Rave: <http://incubator.apache.org/rave>

### Abstract

Apache Rave is A new WEb And SOcial Mashup Engine. It will provide an out-of-the-box as well as an extendible lightweight Java platform to host, serve and aggregate (Open)Social Gadgets and services through a highly customizable and Web 2.0 friendly front-end. Rave is targeted as engine for internet and intranet portals and as building block to provide context-aware personalization and collaboration features for multi-site/multi-channel (mobile) oriented and content driven websites and (social) network oriented services and platforms. For the [OpenSocial](#) container and services the (Java) [Apache Shindig](#) will be integrated. At a later stage further generalization is envisioned to also transparently support [W3C Widgets](#) using [Apache Wookie](#).

### Proposal

The reason for starting Rave is to bring together and combine several existing projects and teams currently working towards more or less the same or overlapping goals but each in their own small(er) target audience and community.

The goal for Rave is to become a lightweight and open-standards based extendible platform for using, integrating and hosting OpenSocial and W3C Widget related features, technologies and services. It will also provide strong context-aware personalization, collaboration and content integration capabilities and a high quality out-of-the-box installation as well as be easy to integrate in other platforms and solutions.

The initial features for Rave will at least be based on the current capabilities from the contributing external projects, for which they will provide the necessary code contributions. However, the code base for Rave will be built anew with strong focus on generalization, customization and extendibility to support the intended multi-purpose adoption and integration. The contributing external projects will start using and switch to the new Rave based solution as soon as the initial features become available to ensure the continued participation and interest from their side as well as their own communities.

#### The intended initial features include:

##### Core Features

1. Advanced OpenSocial compliance and optional features support
2. OpenSocial persistence and SPI (Service Provider Interface) implementation
3. Self-service application administration including security, gadget management and page templates
4. User and group management with full privacy model
5. Gadget repository with life-cycle management (install/update/remove) and extended meta data (categories, comments, ratings, etc.)
6. Dynamic and highly customizable front-end engine (skins, pages, tabs, layouts, navigation)
7. Full OAuth support
8. Support for security restrictions on both Gadgets and page/tag/layout customizations
9. Set of common and general purpose Gadgets to be usable out-of-the-box
10. Support for inter-gadget messaging with examples

##### Extensible Features

1. Pluggable persistence
2. Pluggable security model with example modules for authentication and authorization
3. Support for OpenSocial extensions not (yet) defined in the specification
4. Support for other (non-standard, yet) pluggable container services and extensions

Beyond these initial features the vision and scope for Rave goes much further and includes integrating and providing other highly desired/needed features like:

- native W3C Widgets support through [Apache Wookie](#)
- pluggable and extendible content integration and management services
- space extensions and management features, like [http://wiki.opensocial.org/index.php?title=Space\\_extension](http://wiki.opensocial.org/index.php?title=Space_extension)
- context aware features and extensions integration for personalized and social network and (mobile) device oriented sites and channels
- enhanced client-side widget messaging, coordination and co-location support like using [OpenAjax](#) Hub and Registry
- space, page and Gadget based linking, navigation, coordination and collaboration
- inline widget rendering, like <http://issues.apache.org/jira/browse/SHINDIG-1402>
- [Activity Streams](#) support
- integration with other social networks like with [W3C Federated Social Web Incubator Group](#), [OneSocialWeb](#) and [OStatus](#)
- integration with [Apache Wave](#)
- runtime services monitoring, reporting and configuration/life-cycle management support

### Background

The plan for starting a project like Apache Rave was initially proposed at an [ApacheCON meetup](#) November 2010 in Atlanta and [in more detail](#) at the first [European OpenSocial Event](#) December 2010 in Utrecht (The Netherlands). This resulted in participants from [Pervasive Technology Institute Indiana University](#), [The MITRE Corporation](#), [SURFnet SURFConext](#), [OSS Watch](#), [Hippo](#) and other individuals to come together and discuss the possibilities to collaborate and work together on this joined effort, resulting in this proposal for the Apache Rave project.

## Rationale

The reason for the Apache Rave proposal is that there is a clear need for a more lightweight collaborative and social network oriented integration and services platform, portal if you like, which is easier to use, extend and manage than the more heavy weight J2EE / Portal / Portlet standards based solutions. The OpenSocial, WC3 Widgets and similar related standards are much more "network" and collaboration oriented and lighter weight in their definition, implementation and integration requirements, and are moving much more rapidly and agile forward. Several open-source projects already have stepped into this arena, converting over or starting from scratch, to create new and lightweight solutions based on these emerging standards.

In most cases these new implementations are still at the early stages, and much of the basic work is duplicated as is the effort to build up communities and attract developers. The Apache Rave contributing projects are all building on the same concepts (OpenSocial), have the same or very similar functional needs with regards to Gadget handling, service handling, back-end integration, front-end rendering, code maintenance, etc. The rationale for Apache Rave is to combine these efforts and work and collaborate together on a common base solution and to get out of each own "silo". This joined effort than can benefit everyone, both on technical implementation and feature realization and provide a much wider and stronger community perspective.

The initiating projects and individuals on this joined effort have their existing code based licensed under the ASL 2.0, or will do so as part of this project. Participants include long time users, committers and members of the Apache Software Foundation. That, and the strong alignment with existing Apache projects like Shindig and Wookie, and the strong belief in the Apache "way" with focus on collaboration and community, makes starting the Rave project at Apache the logical thing to do.

The external projects providing the initial code contributions all fully support the goal to come up with a **new** base solution to replace their own project base line solution. Apache Rave will not be started from one of these initial code contributions directly, but be build from scratch where selected features and implementations might be cherry picked from the code contributions where applicable. Each of the code contribution projects are willing to give up their current implementation and replace that with the new solution from Apache Rave, even if no part of their existing implementation details would "survive".

## Current Status

### Community

This project will form a new community, made up initially of the contributors listed below. However, from our experience at the ApacheCon, we believe that this project will quickly draw interest and its community will broaden easily. The proposers have worked well together so far, in producing this proposal and clarifying design approaches, and are keen to continue this in a more open venue.

### Code

Three initial codebases that implement parts of the scope of this project have been offered to this project. All are, or will be, Apache licensed, and grants can be provided. All three of these codebases will be imported into Apache SVN, each into separate sandboxes.

A followup codebase donation is planned, providing a context aware mapping engine and content integration features, under the same conditions as for the initial codebases.

## Initial Goals

The initial goal will be to produce a single usable codebase from the three imported codebases, and to get this to a suitable state where contributing parties can use it within their own products/projects.

This will be done by starting with a fresh, clean repository and then, on a feature-by-feature basis, cherry-picking the best code for that feature from the three codebases.

## Known Risks

Any project joining the Incubator is going to have risks - aspects to it that could lead the project to fail. In this section, the risks that have been noted for the Rave project are discussed.

### Three Code Bases

Working with three initial code bases will be a challenge. However, we are hoping that it will be less of a challenge than starting with none!

All parties contributing code have agreed that they are prepared to participate in this project even if none of their contributed code is incorporated in the final product, recognising that for each feature, it will be necessary to pick the best code for the job, rather than the code that one participant is most attached to.

## Speed of Development

Initially at least, this project is dependent upon contributions made on company time. For this approach to succeed, the project must deliver a workable system in a timeframe acceptable to those companies. The initial parties have the intention of releasing a first version within 6 months after starting the Incubator. Failure to do so could prevent the project reaching critical mass, and could prevent the project from being in a position to attract new developers.

## Inexperience with Community Development

None of the participants have worked together before this project was conceived. Some participants have released open source code, but not done development 'in the open'.

There will clearly be some learning in how to participate together in this project, in a meritocratic way, as a newly formed community. However, we do have in our mix folks that are experienced in open source development, including existing Apache committers and Apache members, as a consequence of which we believe we have enough experience to work with the challenges of forming a new community.

## Reliance on Salaried Developers

At present, the vast majority of contributors will be doing so as a part of their day jobs. Therefore, as already alluded to, there is a risk that the project won't gain enough traction to be of use to their employers. However, given the centrality of these codebases to the participating companies, it is clearly in their best interests to transition to an openly developed alternative.

## Other Risks

Of the other risks commonly raised regarding incubator proposals, none seem particularly pertinent to this project.

The project is no less likely to be orphaned than any other new podling. Given the multi-company support it has easily gained, it would seem to be quite the opposite.

Given the range of organisations putting themselves behind this proposal, there is no obvious risk of over-dependence on one contributing organisation.

The participants are attracted to Apache due to its governance model, and proximity to similar projects, e.g. Apache Shindig. This affinity is clearly shown by the fact that the original group who formed this proposal met at ApacheCon.

## Documentation

Details for the initial contributed code can be seen here:

- MITRE: [OpenSocial Enterprise Container](#)
- SURFnet: [SURFConext Portal](#); [SURFconext technical design](#)
- OGCE [OGCE Gadget Container](#)

Because this project will create new code, we will not have any documentation on first starting. It is likely however that partial documentation can be reused from either codebases.

## Initial Source

The initial source will be made up of three contributions, namely:

- MITRE [OpenSocial Enterprise Container - Source](#)
- SURFnet SURFconext Portal - This source will be brought into the Incubator as SURFnets contribution
- OGCE: Gadget Container code is part of the [OGCE SourceForge Site](#). See [the portal download page](#) for SVN instructions.

## Source and Intellectual Property Submission Plan

### Hippo

The code to be offered by Hippo is already open source and Apache Licensed.

### The MITRE Corporation

The code offered by The MITRE Corporation is already open source and Apache Licensed.

### SURFnet SURFConext Portal

This source will be brought into the Incubator as SURFnets contribution; initial project members have already been given access for evaluation purposes.

### Open Gateway Computing Environments

The OGCE will contribute its Gadget Container code, which is Apache licensed and available from SourceForge, with current dependencies and their licenses described below. Other OGCE software efforts are not covered by this incubator proposal and are not included in the code contribution.

## Dependencies on other Apache Projects

- Apache Shindig
- Apache Tomcat
- Apache Tiles
- Apache HttpComponents
- Apache Commons
- Apache Log4J
- Apache Maven
- Apache Ant

## Cryptography

There is no cryptography in the code at this time.

OGCE has requirements to support security for Computational Grids and Clouds. Although the code will not have any cryptographic implementations, in the future there will be dependencies on external cryptographic libraries. Before committing any cryptographic dependencies, the process listed at <http://www.apache.org/dev/crypto.html> will be followed.

## Required Resources

### Mailing lists

- rave-dev
- rave-commits
- rave-private (moderated subscriptions)

### Subversion Directory

- <http://svn.apache.org/repos/asf/incubator/rave>

### Issue Tracking

We intend to make use of Jira for issue tracking.

Proposed JIRA key: JIRA-RAVE

### Other Resources

None right now.

## Initial Committers

Names of initial committers with affiliation and current ASF status:

- Ate Douma (Hippo), Apache Member, id: ate
- Unico Hommes (Hippo), Apache Committer, id: unico
- Ard Schrijvers (Hippo), Apache Committer, id: ard
- Woonsan Ko (Hippo), Apache Committer, id: woonsan
- Matt Franklin (MITRE, osec)
- Jesse Ciancetta (MITRE, osec)
- Tony Carlucci (MITRE, osec)
- Ross Gardler (OSS Watch), Apache Member, id: rgardler
- Sander van der Waal (OSS Watch)
- Scott Wilson (University of Bolton), Apache Committer, id: scottbw
- Hadrian Zbarcea (independent), Apache Member, id: hadrian
- Marlon Pierce (Pervasive Technology Institute Indiana University, OGCE)
- Raminderjeet Singh (Pervasive Technology Institute Indiana University, OGCE)
- Zhenhua Guo (Pervasive Technology Institute Indiana University, OGCE)
- Suresh Marru (Pervasive Technology Institute Indiana University, OGCE)
- Niels van Dijk (SURFNET, SURFConext)
- Maarten Kremers (SURFNET, SURFConext)
- Bas Zoetekouw (SURFNET, SURFConext)
- Joost van Dijk (SURFNET, SURFConext)
- Okke Harsta (Zilverline, SURFConext)

## Sponsors

## Champion

- Ate Douma

## Nominated Mentors

- Ate Douma
- Sylvain Wallez
- Upayavira
- Ross Gardler
- Hadrian Zbarcea

## Sponsoring Entity

- Apache Incubator Project

## Appendix A: External Dependencies

Here's the list of identified external dependencies for each of the initial code contributions:

### MITRE **OpenSocial** Enterprise Container

- *Build Time/Test Dependencies*
  - Apache Licensed**
    - Jakarta ORO
    - Apache log4j
    - cglib
    - EasyMock
    - Google Guice
    - h2datasource (MITRE Developed)
    - Objenesis
    - Spring Framework Test
    - servlet-api
  - BSD Licensed**
    - YUI Compressor
    - YUI Compressor Ant Task
    - JUnit
  - MPL && EPL Licensed**
    - H2 Database Engine
  - Other Licenses**
    - ASM: <http://asm.ow2.org/license.html>
    - Cobertura: <http://cobertura.sourceforge.net/license.html>
- *Runtime Dependencies*
  - Apache Licensed**
    - Apache Commons
    - Apache Taglibs
    - Google Collections
    - Jackson
    - Apache log4j
    - Shindig
    - Spring Framework
    - Spring Security
    - Apache Tiles
    - Google Guice
    - h2datasource (MITRE Developed)
  - MPL && EPL Licensed**
    - H2 Database Engine
  - EPL Licensed**
    - EclipseLink
    - javax.persistence API
    - AspectJ
  - BSD-style Licensed**
    - FreeMarker: [http://freemarker.sourceforge.net/docs/app\\_license.html](http://freemarker.sourceforge.net/docs/app_license.html)
  - CDDL Licensed**
    - JSTL
    - JavaMail
  - MIT Licensed**
    - SLF4J
  - Other Licenses**
    - AOP Alliance: <http://aopalliance.sourceforge.net/>

## SURFnet SURFConext portal

*Dependencies and licenses of all dependencies of the COIN portal (excluded the shindig dependencies)*

### Apache Software License, Version 1.1

- cglib:cglib-nodep:jar:2.2:compile
- Apache Software License, Version 2.0**
- org.codehaus.jackson:jackson-mapper-lgpl:jar:1.5.5:compile
- org.springframework:spring-core:jar:3.0.3.RELEASE:compile
- org.springframework:spring-orm:jar:3.0.3.RELEASE:compile
- org.springframework:spring-jdbc:jar:3.0.3.RELEASE:compile
- org.springframework:spring-tx:jar:3.0.3.RELEASE:compile
- org.springframework:spring-context:jar:3.0.3.RELEASE:compile
- org.springframework:spring-aop:jar:3.0.3.RELEASE:compile
- commons-dbcp:commons-dbcp:jar:1.2.2:compile
- commons-pool:commons-pool:jar:1.3:compile
- commons-codec:commons-codec:jar:1.4:compile
- org.springframework:spring-webmvc:jar:3.0.3.RELEASE:compile
- org.springframework:spring-asm:jar:3.0.3.RELEASE:compile
- org.springframework:spring-beans:jar:3.0.3.RELEASE:compile
- org.springframework:spring-context-support:jar:3.0.3.RELEASE:compile
- org.springframework:spring-expression:jar:3.0.3.RELEASE:compile
- org.springframework:spring-web:jar:3.0.3.RELEASE:compile
- commons-collections:commons-collections:jar:3.1:compile
- org.codehaus.jackson:jackson-mapper-asl:jar:1.5.3:compile
- org.codehaus.jackson:jackson-core-asl:jar:1.5.3:compile
- org.apache.httpcomponents:httpclient:jar:4.1-alpha2:compile
- org.apache.httpcomponents:httpcore:jar:4.1-beta1:compile
- commons-logging:commons-logging:jar:1.1.1:compile
- org.easymock:easymock:jar:2.5.2:compile
- org.objenesis:objenesis:jar:1.0:compile
- org.apache.commons:commons-io:jar:1.3.2:compile
- javax.validation:validation-api:jar:1.0.0.GA:compile
- org.hibernate:hibernate-validator:jar:4.0.2.GA:compile
- javax.xml.stream:stax-api:jar:1.0-2:compile

#### GNU Lesser General Public License

- c3p0:c3p0:jar:0.9.1.2:compile

#### Dual license of the MPL and the LGPL

- javassist:javassist:jar:3.8.0.GA:compile

#### BSD License

- org.hsqldb:hsqldb:jar:1.8.0.10:compile

#### GPL-2.0

- mysql:mysql-connector-java:jar:5.1.10:compile

#### Public Domain

- aopalliance:aopalliance:jar:1.0:compile

#### LPGL v2.1

- org.hibernate:hibernate-core:jar:3.3.2.GA:compile
- org.hibernate:hibernate-annotations:jar:3.4.0.GA:compile
- org.hibernate:ejb3-persistence:jar:1.0.2.GA:compile
- org.hibernate:hibernate-commons-annotations:jar:3.1.0.GA:compile

#### BSD license

- antlr:antlr:jar:2.7.6:compile
- dom4j:dom4j:jar:1.6.1:compile
- org.hamcrest:hamcrest-core:jar:1.1:compile

#### COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL) Version 1.0

- javax.transaction:jta:jar:1.1:compile
- javax.mail:mail:jar:1.4.3:compile
- javax.activation:activation:jar:1.1:compile
- javax.servlet:jstl:jar:1.2:compile
- javax.xml.bind:jaxb-api:jar:2.1:compile
- com.sun.xml.bind:jaxb-impl:jar:2.1.3:compile

#### MIT license

- org.slf4j:slf4j-api:jar:1.5.8:compile
- org.slf4j:slf4j-log4j12:jar:1.5.8:compile
- log4j:log4j:jar:1.2.14:compile
- org.mockito:mockito-core:jar:1.8.5:compile

## OGCE Gadget Container

### Apache Software License, Version 2.0

- OpenAJAX
- Commons-codec
- Commons-logging
- Commons-fileupload
- Commons-collections
- Commons-digester
- Commons-el

- Commons-lang
- Tomcat Catalina
- Google-collections
- Xalan
- Xerces dom3-xml-apis and dom3-xercesImpl
- Tomcat annotations-api
- commons-beanutils
- commons-io
- openid4java
- axiom-api
- openxri-client, openxri-syntax
- jug
- stax-api
- Guice
- cilogon-portal-servlet (also have BSD & NCSA Licenses)
- **MIT**
- Facebook Java API
- slf4j (3 jars: simple, api, and jdk14)
- xmlsec
- **BSD**
- Dom4j
- Antlr
- **CDDL**
- JSTL
- **GPL V3**
- EXT-JS v2.2
- **GPL v2**
- Trimpath v1.0.38
- Rhino JS 1.7R1 (Also has MPL1.1 license)
- Sun Jersey (Also has CDDL license)
- mysql-connector-jar
- **LGPL**
- Hibernate
- IC3P0
- JBoss Javassist (Also has MPL License)
- htmlparser (also has CPL License)
- **Common Public License, v1**
- Junit, junit-addons
- **Sun Binary**
- Servlet API
- activation
- **EPL**
- higgins-configuration-api
- higgins-sts-api
- higgins-sts-common
- higgins-sts-server-token-handler
- higgins-sts-spi
- **Other**
- Google FriendConnect: <http://code.google.com/apis/friendconnect/docs/terms.html>
- TMT JS: <http://www.massimocorner.com/license.htm>
- JSON: <http://www.json.org/license.html>
- H2: <http://www.h2database.com/html/license.html>
- icu4j: <http://source.icu-project.org/repos/icu/icu/trunk/license.html>