SqueakSource – Smart Monticello Repository
ESUG Innovation Technology Awards 2005

Adrian Lienhard, lienhard@iam.unibe.ch
Lukas Renggli, renggli@iam.unibe.ch

Software Composition Group
University of Bern, Switzerland

May 16, 2005

Abstract

SqueakSource is a web-based Monticello code-repository for Squeak. By providing a web-browser based frontend it facilitates simple means to set up and to use Monticello repository. Not only it makes Monticello conveniently usable for collaborative development, SqueakSource also provides features such as configurable access rights, a wiki, statistics, and rss-feeds.

Keywords: Squeak, Monticello, Source Code, Versioning, Repository, Management, Web Interface, Seaside

1 Description

Monticello is a distributed concurrent versioning system based on a declarative representation of Squeak source code. Since it does not depend on a central repository, several kind of strategies for remote sharing of packages evolved, including HTTP, FTP, and GOODS.

To set up and manage those repository types, though, is not very convenient or even possible for everybody, because a server has to be set up,

configured, and maintained. Another drawback is that those *passive* repositories do not allow one to easily manage meta data, such as user accounts with different access rights, or automatically provide change notifications.

SqueakSource is a repository that provides the same functionality as a HTTP repository but adds meta information such as projects, developers, releases and provides additional functionality such as browsing the code online, generating rss-feeds or statistics about commits. SqueakSource, as seen in Figure 1, is similar to Sourceforge which provides repositories, so that you do not have to manage the hosting of your own.

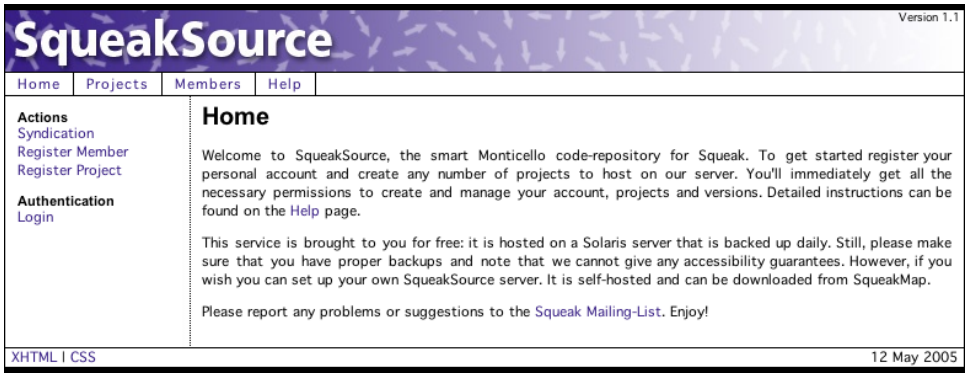

Figure 1: The frontpage of `www.squeaksource.com`.

In contrast to setting up a new HTTP or FTP repository for each new project, SqueakSource allows one to create a new project and get started within seconds. The simplicity of using SqueakSource for Monticello based team development is one of the main reasons why there are that many projects registered on the official server, as the project list in Figure 2 illustrates.

2 Features

Monticello Repository. Acts like a standard HTTP interface for Monticello, therefor no additional code has to be loaded into the developer image.

Access Policies. The security aspects are freely definable by the administrator of a project. The administrator is able define access rights

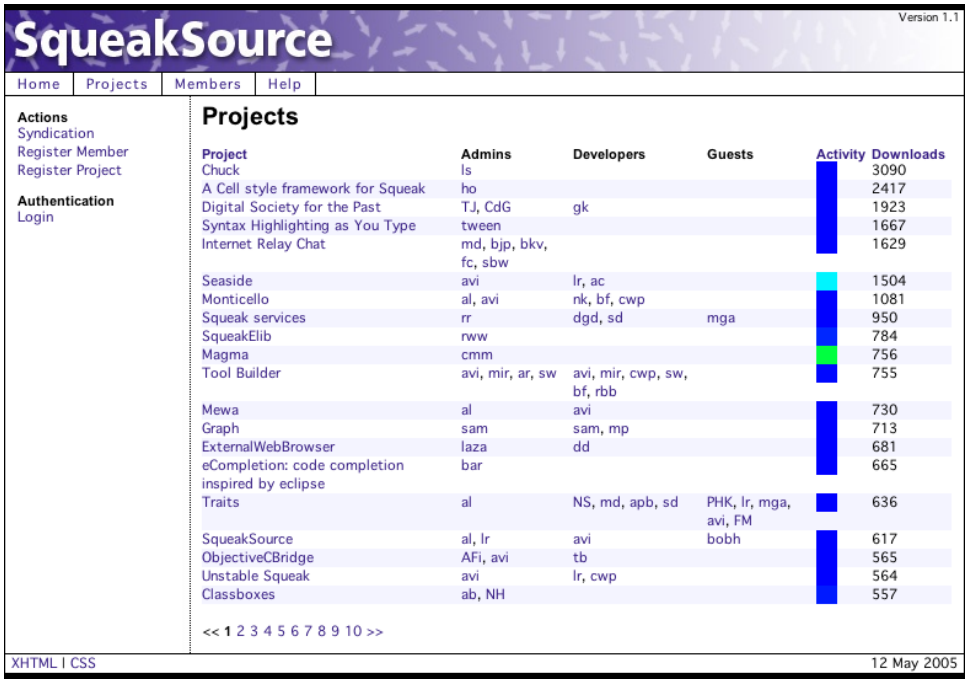

Figure 2: Projects hosted on `www.squeaksource.com`.

globally, such as *no access*, *read* and *read and write*, and assign roles to specific members, such as *guest*, *developer* or *admin*. The guest role only allows the user to read from the repository, the developer role also allows one to commit and as admin one is able to change all aspects of that particular project.

Releases. Specific versions can be tagged as releases and therefor be read by everyone, even if the project itself doesn't allow read access for public.

SqueakMap. Releases can be automatically published to SqueakMap, a catalogue covering most libraries and frameworks to be used in Squeak.

Wiki. Every project includes its own wiki, where the members are able to document their project.

Syndication. Rss-feeds for releases and public accessible versions are generated automatically for all projects to keep observers and developers

up-to-date. Administrators are also able to publish custom messages into their project feed. A master feed, combining all the project feeds of SqueakSource, allows one observing the ongoing activities.

Statistics. Every project includes simple statistics about the number of downloads, versions and releases.

Importing. It is possible to import existing monticello directories into SqueakSource by using a mass-import facility.

Browsing. If the permission is granted, one is able to browse the full source code of any version of a project within the web-browser, using a similar interface like the Smalltalk code-browser within Squeak, see Figure 3.

User Management. Registered members have their own account with a login. If one forgets his password a mail can be requested which has a secure link to a page where she is able to edit her account.

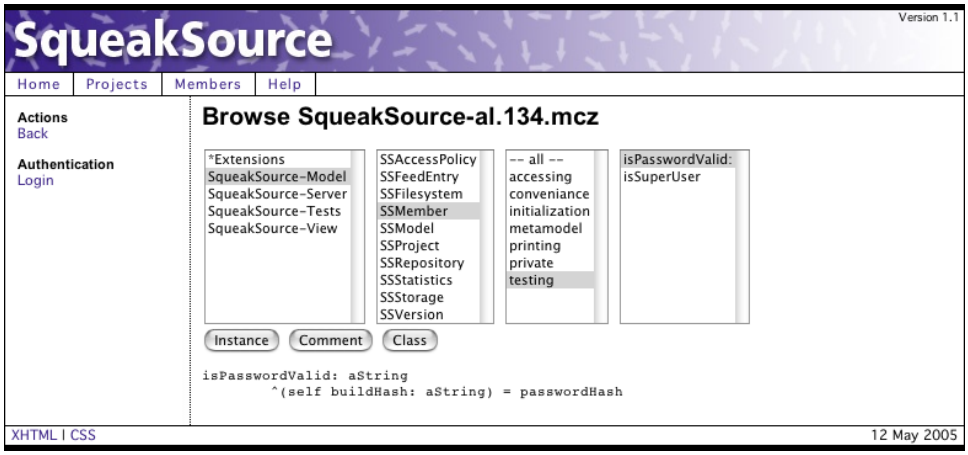

Figure 3: Browse the full source-code of a public Monticello package.

3 Platfrom

SqueakSource has been implemented in *Squeak 3.6* using *Seaside* as a web-framework.

4 License

SqueakSource is released under the *MIT license* which grants unrestricted rights to copy, modify, and redistribute as long as the original copyright and license terms are retained.

5 Developers and Hosting

SqueakSource was implemented at the University of Bern by *Adrian Lienhard* and *Lukas Renggli*. The main SqueakSource server, as shown in Figure 1 is hosted on a machine at the same place and is maintained by the authors. Figure 4 shows statistics that were collected since the start of the official SqueakSource server in February 2004.

6 References

To download the latest code base for Squeak using the *SqueakMap Package Loader* and load the package called *SqueakSource*. SqueakSource is self-hosted as illustrated in Figure 5: latest releases can be found on the main SqueakSource server itself.

Running instances of SqueakSource servers we are aware of are:

`http://www.squeaksource.com` (main server)
`http://squeak.saltypickle.com`
`http://source.impara.de`

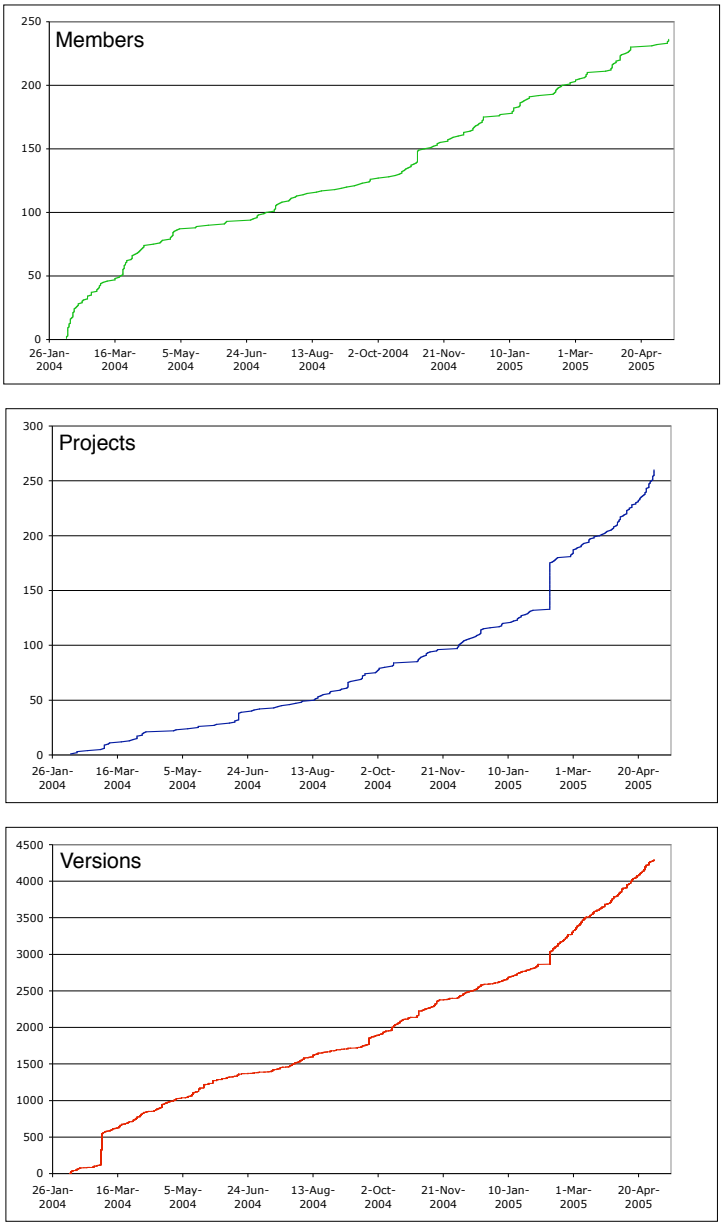

Figure 4: Member-, Project- and Version-Statistcs of `http://www.squeak-source.com`.

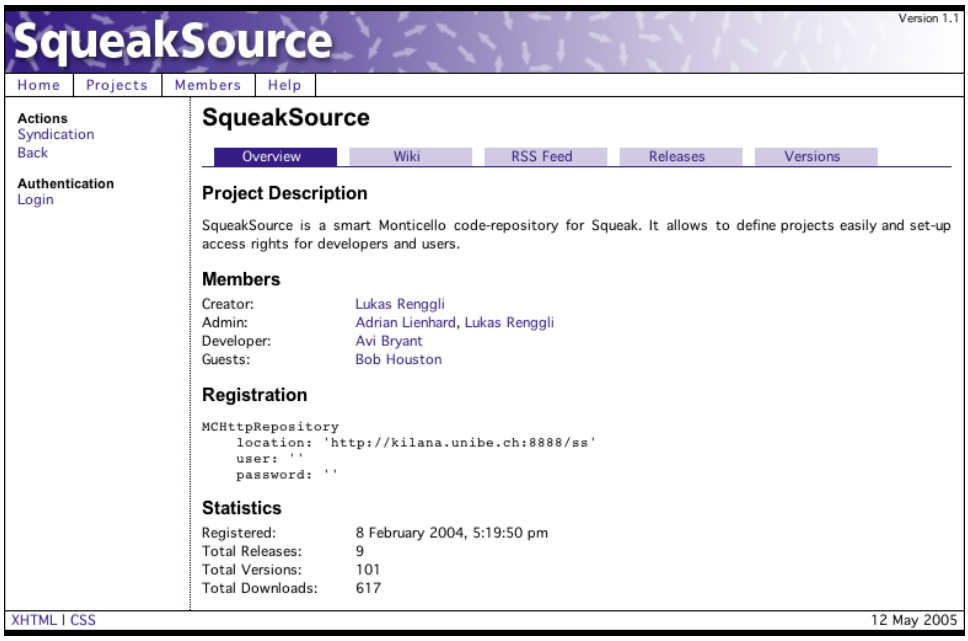

Figure 5: The front-page of the SqueakSource project, hosted itself on SqueakSource.