

The Long Term Future of LionShare
Implementation and Development
DRAFT 8/22/05

Alex Valentine
alexvalentine@psu.edu

The LionShare project is in the process of completing a two year grant awarded by the Andrew W. Mellon foundation. The LionShare team will release version 1.0 on September 30th 2005. The initial grant is scheduled to end on September 30th 2005. LionShare development will continue on past the September 30th release date until the end of 2005, without additional funding thanks to a no-cost extension. The additional development period will be spent on incremental improvements to the existing deliverables, focusing on usability and scalability.

The purpose of this document is to outline the post-grant future of the LionShare project, specifically: development paths, financial viability, and implementation opportunities. This document proposes a two tiered development track. The first track is a federated LionShare network adopted by several large scale institutions. LionShare will need to be thoroughly tested and piloted before multiple universities would even consider running LionShare as an everyday service. In order for this track to succeed, there will need to be an institution that will serve as the shining example of a successful LionShare implementation.

The second development track will refactor the LionShare model to meet the needs of smaller institutions. The infrastructure dependencies required for the current iteration of LionShare are simply too overwhelming for a small institution to handle. While the advanced dependencies are a fact of life for a federated inter-institutional network, they are also a hindrance to adoption.

The second development track for LionShare is a bold proposal which eliminates all the infrastructure requirements for the implementation of LionShare, creating a self contained network within smaller organizations. Such networks would not be able to communicate with each other or with the federated LionShare network, but they would serve as a way for smaller organizations to build campus wide networks without any organizational requirements.

The final portion of this document outlines the financial opportunities and implementation plans of the the LionShare project. In order for the project to succeed on a long term basis, it must self-sustaining from a financial perspective. There are several options under consideration, including revenue streams from implementation, support, and additional development paths funded by participating institutions.

Strengthening Federated LionShare (10/1/2005 – 12/31/05)

The LionShare design strategy for 1.0 development centered around developing a secure federated P2P network for academic collaboration. This theme will continue after the 1.0 release, with a greater focus on making the necessary steps for LionShare to become an everyday academic service. The proposed development plan for federated LionShare will make evolutionary changes to the existing model in order to encourage adoption of LionShare by major institutions.

The focus for federated LionShare will be on scalability, usability, and expandability. During the development process leading up to the 1.0 release, much of the development time was spent on getting the basic functionality of each piece working and then moving on to the next piece. This development strategy allowed the LionShare team to build a large amount of components in a short period of time. While the rapid development plan led to the on-time development of all the major LionShare components, there was not enough focus on making small usability optimizations. Usability refactoring will be a major focus for post 1.0 LionShare development.

The second portion of the development plan for federated LionShare is scalability testing and optimization. The nation's largest learning institutions are the initial targets for potential LionShare adoption. Institutions that have most of the dependencies and expertise in place to adopt LionShare without adding an array of new services. Our target institutions will need to have a documented and proven scalability analysis before even considering the adoption of LionShare as a stable academic service.

The distributed nature of LionShare's topology makes it a scalable design by default. The modern Gnutella messaging system has already scaled to hundreds of thousands of concurrent users online. The design of LionShare is quite different than a typical decentralized P2P network. Centralized pieces such as the SASL-CA and PeerServers will need to be tested and optimized to service as many users as possible.

The final portion of federated LionShare development is expandability. A diverse array of communities have shown interest in using LionShare to share a wide range of academic materials. Every few weeks, a new repository is added to the Peer's search capabilities. A portion of development work for the next stage in the life of LionShare will need to be spent adding user requested additions and modifications.

LionShare 2.0: Dependency Optional

LionShare's greatest strength and greatest weakness is the integration with campus wide services to create a truly federated P2P file sharing network. The current installation requirements for LionShare include:

- Shibboleth

- Kerberos (The SASL-CA supports others authn methods, but GSSAPI/Kerberos is the only tested SASL mechanism)
- Directory Service (LDAP or others)
- SASL-CA
- Dedicated Ultrapeer (optional but recommended)
- Webcache (optional but recommended)
- PeerServer (optional but recommended)

Assuming an institution has all the dependencies required, configuration to integrate campus services is time consuming. After configuring the SASL-CA to pull all the relevant information from campus services, an adopting institution will need to change Peer settings to integrate with the newly installed SASL-CA. Additional packaging and customization might be required to customize the look and feel of the Peer to meet the needs of the adopting institution.

In order to increase adoption of the LionShare Peer, it has been suggested that we look at ways to reduce the amount of dependencies required for implementation. The primary issue with dependency minimization, is the subsequent reduction in the overall security of the LionShare network. For instance, the requirement of an identity management system is in place in order to prevent the sharing of copyrighted materials and to allow users to create access control rules. The requirement of a directory service is used to obtain valid user attributes. The Shibboleth requirement is necessary for the federated model of the LionShare network.

While dependency reduction is not an option for an institution looking to connect to the federated LionShare network, there is an option to modify the LionShare codebase to create a campus wide P2P network that does not connect to the federated LionShare network. LionShare 2.0 is a proposal to create a self contained LionShare package that has zero dependencies, but retains the ability to connect to campus wide services if available.

The primary goal for LionShare 2.0 would be to create a single server model that takes care of all the dependencies for operating a self contained LionShare network. The default package would provide the fundamental capabilities such as user management, certificate generation, PeerServer, UltraPeer and bootstrap capabilities in an easy to use package.

The default configuration would use built-in user management and directory services, but every function could be configured to operate with external services such as Kerberos and LDAP. The basic insulation would allow administrators to set up a self contained LionShare network quickly, while allowing a configuration to move incrementally to a structure that ties in to campus services.

The LionShare 2.0 server package would contain Apache tomcat and set of webapps to perform all the basic services on a single server installation by default. Configuration of every service would take place via web-browser, eliminating the need for text file configuration. Customization and deployment of the LionShare Peer would be wizard oriented task requiring no prior knowledge.

The advantage of refactoring LionShare to operate as a single server model, is that it would open up the population of potential adopters, removing all the barriers to entry created by having a set of strict dependencies. The current requirements of LionShare make adoption feasible to only a limited number of institutions, without requiring the installation of additional services. This proposal for refactoring LionShare would make virtually every organization and organizational unit, a potential adopter. Institutions that eventually meet the requirements for the federation version of LionShare, could join without having to re-deploy every single component in their existing self-contained LionShare network.

The Road to Self Sufficiency

The development of LionShare was made possible by a grant from the Andrew W. Mellon foundation. The initial development grant officially ends on September 30th 2005. LionShare Development will continue past September 30th up to the end of 2005, thanks to a no-cost extension. The post 1.0 development period, from October 1st till the end of December, will be spent strengthening federated LionShare to meet the requirements of large institutions.

The future of LionShare after the no-cost extension is uncertain at this point. In order for institutions to consider adopting LionShare, steps will need to be taken to ensure that long-term support and integration services will be available. Institutions are also unlikely to adopt a product, commercial or non-commercial, when active development has ceased.

To ensure that the LionShare project will reach a successful state of adoption, the next stage of development will need to be funded by an outside organization. Widespread adoption and self sufficiency, will be the next mission of the LionShare project. In order to spark adoption of LionShare, the dependency reduction path of LionShare 2.0 will need to be funded. Beyond development changes, offerings for official support and integration services must be available for adopting organizations.

While outside funding will help pave the way for widespread adoption of LionShare, there will need to be a restructuring of the project to create a revenue model for self sufficiency. Relying on outside funding is an unsustainable method for the long term viability of a software project. In the past, projects have used a variety of methods for developing revenue streams that eventually lead to self-sustainability.

One option is the development of a non-profit organization that offers support and integration services. Revenue collected from services is used to fund further development. A common method for revenue generation is to form a federation of adopters, who help fund development and support services, mutually benefiting the adopters and the project. Some federations, such as institutions participating in the development of Sakai, offer not only monetary resources, but human resources as well.

The future of LionShare is dependent on developing revenue sources for long term sustainability. The development of LionShare 2.0, to rid the curse of hard coded

dependencies, will go a long way towards increasing adoption. Only then, will the widespread interest of a growing install base, create an environment where LionShare could develop a self sustaining revenue model for long term viability.

Implementation and Testing

The LionShare project has featured publicly accessible development releases since the 0.4 LionShare Peer release of January 2005. During this time, any PSU access account holder has had the ability to download and test the LionShare Peer. In the Spring of 2005, the LionShare team set up an additional testing realm for users who do not have a Penn State access account. Higher education faculty and staff are able to register for a LionShare testing account.

Because of the rapid release cycle and the early stage at which the project has allowed public access, the amount of actual use from outside the development team has been relatively small by design. The availability of test releases is a feedback tool for the development team.

At this time, there are plans in the works for formalized testing, including a pilot program for LionShare as an official Penn State service. During the upcoming fall semester, LionShare will be used by several groups of students and faculty. The assessment team will be working to use these initial trials as a means for improving the overall usability and convenience of the application.

The additional Fall development time will be spent on the testing and optimization of LionShare components for use in a large scale pilot. The goal of the pilot will be to test the viability of the LionShare network as a service for Penn State. Mike Halm is in discussions with the appropriate departments to finalize a timeline for Penn State's LionShare pilot.

The proposed timeline for the Penn State pilot features a stable network for use by Penn State access account holders. Unlike current development releases, the LionShare pilot will be publicized and promoted through normal university channels. The goal of pilot will be test the viability of LionShare as a full fledged academic service. The proposed pilot timeline would end with a decision to adopt LionShare as a fully supported Penn State service during the 2006-2007 academic year.

During the proposed pilot timeline, the LionShare team will be seeking partners to develop a stable inter-institutional network. The level of Penn State's enthusiasm towards adoption of LionShare will be vital for the success of federated LionShare. Not only will Penn State's adoption of LionShare improve interest in joining federated LionShare, but it will also serve as a great example. Small universities interested in installing the self-contained version of LionShare, can look to Penn State as an example of LionShare's scalability.

Looking Towards the Future

The original LionShare proposal, was a bold effort to take the strengths of P2P and centralized services, to create a new model for secure collaboration. The development of LionShare is closing in on the lofty goals declared by the original proposal. The strength of depending on several independent services, for the federated version of LionShare, the biggest roadblock to widespread adoption by mid-size and small institutions.

The LionShare project should not drop the federated and highly secure roots of the initial proposal, instead we should look towards refactoring LionShare so it can meet the needs of smaller organizations, while retaining the federated capabilities and secure roots of the initial proposal.

The long term status of the LionShare project is uncertain at this point in time. Steps will need to be taken to ensure that LionShare will become a long term success. Development of LionShare 2.0, will lead the way towards widespread adoption, creating an environment where the project can exist without depending on outside funding.

With continued support, LionShare has the potential to change the way people share information, on campuses around the globe.