

Google Summer of Code Ideas

DuraSpace GSoC Ideas List

This page provides a summary of GSoC ideas from across all [DuraSpace technology platforms](#). Please note: not all of the DuraSpace platforms may provide projects for a given Google Summer of Code. As each platform has its own separate development community, it all depends on whether volunteer mentors are available during a given year.

More information about DuraSpace and these technology platforms can be found on our [Google Summer of Code \(GSOC\)](#) page.

GSoC Ideas Lists per platform:

- [DSpace 2014 Ideas List](#)
- [Fedora 2014 Ideas List](#)
- [DuraCloud 2014 Ideas List](#)

DSpace 2014 Ideas List

The [DSpace](#) platform offers the following Google Summer of Code project ideas. If you would like more information about DSpace or about any of these GSoC ideas, please feel free to contact the DSpace Developers by any of the following:

- DSpace Developers Mailing List (requires signup): <http://lists.sourceforge.net/lists/listinfo/dspace-devel>
- DSpace IRC: [#dspace](#) on the 'irc.freenode.net' server
- More information about the DSpace Developers' strategies for successful GSoC projects: [DSpace Summer of Code](#)
- More information about DSpace in general can also be found in the latest Documentation: [DSpace 4.x Documentation](#)

NOTE: Mentors who wish to add to this list should visit [DSpace Summer of Code Ideas](#) (which is the page that generates the below table)

Project Title	Description	Mentor Volunteers
File format validation with DROID and JHOVE	<p>When a file is currently uploaded in DSpace, its file format is guessed by looking at the file extension. This means if I change the extension ".exe" of a file containing a virus to ".pdf", DSpace will not have a clue what's going on and identify the file as a PDF. You can make this more robust by working on the class FormatIdentifier. The student is expected to work on this class to handover the task for format identification to a library like JHove, DROID, or both.</p> <p>Minimal viable work:</p> <ul style="list-style-type: none">• Implement JHove OR Droid, and demonstrate that it can perform identification on a number of text and image formats when uploaded to DSpace. <p>Extra kudos:</p> <ul style="list-style-type: none">• The student implemented BOTH JHove and Droid, and produced a performance analysis on which of the two (or both) should be preferred, leading up to a more optimized solution. <p>GODLIKE:</p> <ul style="list-style-type: none">• Aside from the inclusion in the submission form, the student also creates a DSpace Curation Task that allows a repository manager to run file format identification against existing items, or entire collections. The administrator receives a report on those items that are differently identified by JHove/DROID, compared to what's already stored as the bitstreams format. <p><i>Difficulty level: Easy</i></p>	<p>Philip Vissenaekens (Atmire)</p> <p>Andrea Schweer</p>

<p>Translate Wiki Integration</p>	<p>The DSpace community has approached TranslateWiki.net (TWN), the mediawikibased platform for interface translation of open source projects. The initial discussions are promising and the TWN community is currently building support for the Apache Cocoon message format that is used by XMLUI. We need an ambitious GSOC student to connect the dots, working with both communities in extending the integration and ensuring that in the end, the threshold is lowered for external translators to provide interface translations for DSpace.</p> <p><i>Related Links:</i></p> <ul style="list-style-type: none"> • JIRA Ticket describing the current status of the dialogue between DSpace and TWN <p><i>Related DSpace Components:</i></p> <ul style="list-style-type: none"> • dspace-api-lang : JSPUI and API level language packs (in Java Properties file format) • dspace-xmlui-lang : XMLUI language packs (in Apache Cocoon XML format) • dspace-xmlui : XML-based interface based on Apache Cocoon • dspace-jspui : Custom JSP-based interface <p><i>Recommended Skills:</i></p> <ul style="list-style-type: none"> • Multilingual and/or translation experience • Familiarity with Java web application Internationalization <p><i>Difficulty Level:</i> Medium</p>	<p>Bram Luyten (Atmire)</p>
<p>Virtual Sets: Separate the internal repository structure from the navigation structure</p>	<p>Currently, the hierarchical structure used in DSpace allows sharing items between collections by explicitly declaring these relations on each item. However, DSpace does not allow to relate a collection or a sub-community between two or more communities.</p> <p>Virtual Sets are arbitrary aggregations of DSpace Objects and criteria, composed of:</p> <ul style="list-style-type: none"> • arbitrarily selected communities, collections and/or items, • dynamic results from criteria/queries (logical expressions; e.g. Solr queries) • other declared Virtual Sets (initially, cycles are not allowed) <p>Virtual Sets in DSpace would allow the creation of complex navigation structures regardless of the hierarchical (perhaps administrative) structure of communities and collections.</p> <p>Structures, as described above, are supported in at least another repository software such as Fedora-based Hydra due to its generic object model.</p> <p>The initial implementation would affect the dspace-api component and either of the web UIs, since both now use Discovery (Solr) by default. Virtual Sets should be implemented at the dspace-api level for DSpace Objects to offer more orthogonal features like Virtual Sets backup, export and exposure through OAI-PMH. Stretch goals would include making use of Virtual Sets in other interfaces like REST and/or OAI.</p> <p><i>Related links:</i></p> <ul style="list-style-type: none"> • Sets, sets or sets? <p><i>Related DSpace Components:</i></p> <ul style="list-style-type: none"> • dspace-api : DSpace Java API • dspace-xmlui : XML-based interface based on Apache Cocoon • dspace-jspui : Custom JSP-based interface <p><i>Recommended Skills:</i></p> <ul style="list-style-type: none"> • Java programming experience <p><i>Difficulty Level:</i> High</p>	<p>Ivan Masár</p>

Create a puppet-dspace module	<p>Develop a Puppet-DSpace module for deploying DSpace, and provisioning DSpace-related services. The goal would be to deploy on at least one popular Linux OS, and likely the target will be Debian/Ubuntu since testing will be done with Vagrant-DSpace, which currently uses only Ubuntu. The final product should be useful for deploying DSpace to a cloud infrastructure, or really any server configured to run Puppet (in other words, the module should make no assumptions that rely upon paths used by Vagrant). A stretch goal for this project will be to make the puppet module OS-agnostic, and have it run on both Debian/Ubuntu as well as CentOS/RHEL. But just meeting the Ubuntu target would be sufficient for this project. The puppet module already built for use in Vagrant-DSpace would be a great starting point. The module, when complete, should enable an operator to go from a standard OS base image to a running instance of DSpace, complete with a container to host the application, and (optionally) a PostgreSQL database for metadata.</p> <p><i>Related DSpace Components/Links:</i></p> <ul style="list-style-type: none"> • vagrant-dspace : A Vagrant setup for DSpace development. It includes the "starting point" for a puppet-dspace module under "/modules/dspace/" • Installing DSpace : DSpace 4 Installation instructions <p><i>Recommended Skills:</i></p> <ul style="list-style-type: none"> • Experience with Puppet or similar tools (e.g. Chef). At a minimum some base familiarity and even some Ruby experience (which Puppet is based on). • Familiarity with Vagrant, or willingness to learn. <p><i>Difficulty Level:</i> Medium</p>	
Next-gen UI	<p>MDS is an experimental offshoot of DSpace in which new ideas may be prototyped and examined. Recently a REST API (with CRUD operations, etc) has been added to MDS. A valuable 'proof of adequacy' is building an entire functional web UI backed only by the API. The goal of this project is to construct such an admin UI for MDS, using a modern, agile web application framework. A proof of concept using AngularJS already begun could serve as a basis for further work.</p> <p><i>Related DSpace Components:</i></p> <ul style="list-style-type: none"> • mds : "Modernized DSpace". An attempt to refactor/redesign the DSpace API to make it more simplistic/modernized. This is a side-project of long-time Committer, Richard Rodgers • mds/webapi : the REST API for MDS project. This API supports CRUD operations. • dspace-rest (loosely related): the official REST API which now ships with DSpace 4 may also provide a possible integration point. However, it is currently read-only. <p><i>Recommended Skills:</i></p> <ul style="list-style-type: none"> • Experience or familiarity with one or more agile web frameworks • Experience or familiarity with buiding agile interfaces against a REST API <p><i>Difficulty Level:</i> Medium to High</p>	<p>Richard Rodgers</p> <p>Philip Vissenaekens (Atmire)</p>

Fedora 2014 Ideas List

- Fedora will not be participating in GSoC 2014.

DuraCloud 2014 Ideas List

- DuraCloud will not be participating in GSoC 2014.