Proposed RoadMap to 2.0

This page represented a proposed RoadMap to 2.0 from Tim Donohue. It still needs broader discussion/approval before adoption.
This proposal was initially discussed in the DSpace Developers Meeting on March 16, 2011. See the meeting notes for more info about initial discussion.

Overarching Goals / Priorities

The overarching, defining goals of this general 2.0 RoadMap are as follows:

1. Simplify the process of installing & upgrading (make it easier for institutions to stay "up to date" with their installation)
   - As a subpoint, also make steps to simplify configuration/setup, so that more of it can be completed via an Admin User Interface
2. Improve upon the ability to import / export data and generally make data more accessible/usable by external applications/systems (i.e. improve "integration points")
3. Make steps to allow for decentralized support of add-ons, third-party user interfaces, etc. (i.e. make steps to no longer require that all DSpace modules/UIs need to be supported centrally by the Committers)
4. Make steps to enable DSpace with Fedora Inside as an installation option

Proposed RoadMap to 2.0

Please note the the features listed under each release are only suggestions of priority! It may be that specific tasks/features will end up happening in either an earlier or later release, based on available developer/committer time.

DSpace 1.8.0 (Oct 2011)

The following modifications should be high priority for this release:

1. Initial simplification of Installation/Upgrade processes (see DS-802)
2. Refactoring & modularization, which may include taking initial steps detailed in the following proposals (*Note: the implementation details of these proposals are obviously up for discussion*):
   a. Begin to Restructure Trunk Projects and Asynchronous Release Process - with a goal of ensuring our various applications do not **all** need to be installed out of the box
   b. Begin Refactoring the DSpace Domain Model - with a goal of ensuring we begin to separate our data model from underlying data access services (will be necessary to help support move towards Fedora-Inside, while also potentially enabling use to expand support for additional databases, etc)
   c. Move towards more widespread usage of DSpace 2.0 Services (some examples in this proposal: Replace DSpace ConfigurationManager and PluginManager)
3. Begin to Improve "Interoperability" / Integration points
   a. Release a beta version of REST API
   b. Potentially allow for AIP Transmission via SWORD?
4. Also release features which just missed 1.7, but seem of high interest. Especially the following (this is not an exhaustive list!):
   a. Additional Curation Tasks?
   b. Configurable Reviewer Workflow - project that enhances Reviewer Workflow configurability that is orthogonal to Curation Tooling.
   c. Context-Guided Ingest?
   d. SWORD Client for DSpace? - of high interest if we wanted to enable AIP Transmission via SWORD
   e. Other features listed on DSpace Release 1.8.0 Notes

DSpace 1.9.0 (TBD - Oct 2012?)

The following modifications should be high priority for this release:

1. Additional Refactoring & Modularization (especially in preparation for "DSpace with Fedora Inside"). This may include the following proposals (*Note: the implementation details of these proposals are obviously up for discussion*):
   a. Any refactoring that didn't happen in time for 1.8.0
   b. Refactor Packagers to support Chain of Command (if it doesn't make it in 1.8.0)
   c. Metadata For All - begin to enable metadata on all DSpace Objects
2. Improvements / simplification of DSpace Configuration (perhaps even moving some configs to a DB). Minimally, this needs to simplify the 'dspace.cfg'.
3. Change which Interfaces come "out-of-the-box", and allow others to be optionally installed at a later time (*The following is only meant as an example.*)
   a. Default install may only include: XMLUI, SOLR, possibly SWORD?
   b. Change LNI, OAI-PMH, REST and JSPUI to "optional".
      i. Potentially 'deprecate' support for LNI (or locate external, third-party support for it). Discuss whether any other Interfaces should be similarly deprecated (based on community-wide usage, etc).
   c. Encourage the community to develop / share / support their own third-party UIs (which potentially could communicate with DSpace via REST API, or via a refactored DSpace Domain Model API).
4. Asynchronous Job Scheduler? (would replace the need to schedule jobs via 'cron' or similar)
5. Any Features/Changes that just miss making it into 1.8.0
   a. Context Sensitive Help for XMLUI (@mire)

DSpace 2.0 (TBD - Oct 2013?)

1. Support for DSpace with Fedora Inside (will likely require more Refactoring/Modularization to achieve)
2. Any additional refactoring/changes that didn't happen in time for 1.9.0.