DSpace Under the Hood

How DSpace works
This presentation

• Purpose
  • You don’t need to be a mechanic to drive a car…
    …but it helps if you know how to top-up your oil, check your tyres and explain problems to your mechanic

• Two presentations
  • How DSpace Works
  • The DSpace Community Development Process (and YOUR role in it)

• The presenters
  • 3 repository managers / DGOC members
  • 3 developers / committers
  • 29 years of combined DSpace experience
Contact details

- Leonie Hayes
  - Research Repository Manager, The University of Auckland Library
  - l.hayes@auckland.ac.nz (1.5)

- Richard Jones
  - Head of Repository Systems, Symplectic Ltd.
  - richard@symplectic.co.uk (1.0)

- Stuart Lewis
  - IT Innovations Analyst and Developer, The University of Auckland Library
  - s.lewis@auckland.ac.nz (1.2)

- Monica Roos
  - Special Librarian, The University of Bergen Library
  - monica.roos@ub.uib.no (1.4)

- Kim Shepherd
  - IT Software Analyst and Developer, The University of Auckland Library,
  - k.shepherd@auckland.ac.nz (1.5)

- Elin Stangeland
  - Repository Manager, Cambridge University Library
  - es444@cam.ac.uk (1.2)
A bird’s eye view
The information architecture

• AKA: “What goes where?”

• Parts of an item are stored in different places
  • …for different reasons

• Some data needs to be searched
• Some data needs to be displayed
• Some data is large
The Database

- Item metadata
- Users
- Community and collection structure
- Browse indices
- Authorizations
- Registries (metadata and bitstream)
Files on disk

- Files stored in the ‘asset store’
- Each bitstream has an internal id; a large integer derived from the date of submission and checksum
  - e.g. 908735856294756292618068267495783
- Files are stored in the directory ab/dc/ef, where abcdef are the first 6 characters of the internal id
  - e.g 90/87/35
- Files are stored with the internal id as the filename
- All actual information about the file is in the "bitstream" table in the database, including internal id.

908735856294756292618068267495783 ->
/dspace/[assetstore]/90/87/35/908735856294756292618068267495783
• To make DSpace work in a local context
• Enable local settings – urls, links to physical storage etc.
  • dspace.cfg
  • Lots of settings – read it one day!

• Enable local configurations
  • Input forms
  • Metadata crosswalks
  • OAI-PMH configuration
  • Language packs
Lucene search indexes

- Maintained outside of the database
- Contains the search index
- Very fast for full text searching
  - Stop words and stemming
- A cached copy
  - Can be re-built or updated from the database
  - [dspace]/bin/dspace index-update
Log files

- The dspace.log file performs several functions as:
  - Providing log information useful in debugging
  - Providing information about who has accessed the system
  - It generates the classic DSpace statistics
  - Contain error messages useful for fixing problems
• Classic statistics were derived from log files
  • Only generated once a day
  • Dependant on keeping all your old log files

• Solr stats
  • Statistical information entered real-time into solr index
    • Event-based, not log-based
  • Able to query statistics
  • Different views of statistics
    • Able to write new statistics reports
Scheduled tasks

• Things need to happen in the background:
  – filter-media (generate full text search indexes, create image thumbnails)
  – sub-daily (send daily subscription emails)
  – stat-general / stat-monthly (generate classic statistics)
  – generate-sitemaps (creates search engine sitemaps)
  – checker (checksum checker)
Authentication

- Authentication is about verifying who a user is
  - DSpace provides multiple options
    - In-built database of users and passwords
    - LDAP login (link to local LDAP or ActiveDirectory)
    - Shibboleth
    - IP authentication

- Makes use of a plugin stack
  - You can chain authentication methods
  - You can create new authentication methods
Authorization basics

- Resource policies

- EPeople - DSpace users
  - Groups
    - Anonymous
    - Admin
  - Special groups

- Permissions
  - READ / WRITE
  - ADD / REMOVE
JSPUI styling options

- All JSPs: dspace-jspui/dspace-jspui/webapp/src/main/webapp
- Main layout: [jsp]/layout and [jsp]/styles.css.jsp
- Page localisations should go in dspace/modules/jspui
  - Caveat: The more changes you make, the harder it is to upgrade

```
header-default.jsp
```

```
location-bar.jsp
```

```
navbar-default.jsp
navbar-admin.jsp
```

```
Page Content
```

```
footer-default.jsp
```
Composed of two components that control different part of the Manakin interface:

- **Aspects:**
  - "plugins" that provide features for the repository

- **Themes:**
  - "Rules" that defines where themes are installed in the repository.
  - Style the look-and-feel of the repository.
Error messages

- Help – an error message!
  - To help you, we need to know exactly what went wrong
    - The easiest way to do this, is with a ‘stack trace’
  - XMLUI: Shown on the error page
  - JSPUI: Shown in the HTML source
  - All: In the dspace.log file
Help – an error message!

To help you, we need to know exactly what went wrong

The easiest way to do this, is with a ‘stack trace’

2008-09-18 15:08:13,263 INFO org.dspace.app.xmlui.utils.AuthenticationUtil @ [EMAIL PROTECTED]:
session_id=F1FB96AF6FA3464C393A3366621534A4:ip_addr=139.147.66.108:
login:type=explicit java.lang.NullPointerException

at
org.dspace.authenticate.LDAPHierarchicalAuthentication.getSpecialGroups(LDAPHierarchicalAuthentication.java:144)

at
org.dspace.authenticate.AuthenticationManager.getSpecialGroups(AuthenticationManager.java:308)

at
org.dspace.app.xmlui.utils.AuthenticationUtil.logIn(AuthenticationUtil.java:222)
To: dspace-tech@lists.sourceforge.net
Subject: Error with XYZ when ABC happens

Dear dspace-tech,

[Description of your problem]
[Stack trace and/or log file extracts]
[DSpace version]
[Environment: OS flavour]
[Any other relevant details]

Please help!

Richard
DSpace Under the Hood

Any questions?
DSpace Under the Hood

The DSpace Community Development Process (and your role in it)
Core Development:

• Analysts
• Software Developers
• Designers
• Testers

DSpace Analogue:

• Repository Managers
• In-house, external consultants, committers, contributors
• Software Developers
• Committers and community; testathon
DSpace community vs Commercial Company

Product Environment:

- Writers
- Marketing
- Sales
- Customer Support

DSpace Analogue:

- Repository Managers, Committers, DuraSpace
- DuraSpace, the Community
- DuraSpace
- Mailing Lists
DSpace community vs Commercial Company

Organisational Support:

• Project Management
• Finance and HR

DSpace Analogue:

• Repository Managers, DuraSpace
• Participating Institutions, DuraSpace
How the community fits together
DSpace committers group

- Committer = able to ‘commit’ code to the code repository
- General voting rights (usually opened to anyone)

- Becoming a committer
  - Meritocracy
  - Vote by current committers
  - We keep an eye out for new potential committers
  - You can suggest yourself as a new committer
  - We need more committers!
  - Some committers are not developers
  - We like to see:
    - Dedication to DSpace (usually via employment)
    - Friendly and helpful (participate in email lists)
    - Contributes code
    - Joins in development discussions
  - Some take years, some take only a few months
DSpace Global Outreach Committee

• Repository managers group
• Current projects
  o DSUG meetings at Open Repository Conferences
  o Community requirements gathering
  o DSpace instance database development

• Engagement in development process
DSpace Ambassadors

• A recent initiative with a focus on regional networking

• Focusing on connecting DSpace users together

• Providing support, networking and mentoring for new users

• Further details: Contact Valorie Hollister
Developing code for DSpace

- Submit patches

- Follow guidelines
  https://wiki.duraspace.org/display/DSPACE/Guidelines+for+Committing

- Don't throw over the fence and leave - engage with the committers

- Inform the community what you’re doing

- Ask for feedback, early and often
• Issue tracking and project tracking for software development
  • A worldwide tool used by many groups for many purposes
  • Organised around projects (e.g. DSpace 1.x), components (e.g. XMLUI), issues (e.g. a sw bug) and workflows
  • Issue prioritisation
  • Voting
What goes in JIRA?

- **Bug Reports**
  - Supply as much information as possible
  - If you're not sure, email dspace-tech first

- **Bug Fixes**
  - If you think you've fixed a bug, attach a patch!
  - Encourage peer review and constructive suggestions

- **New Features**
  - JIRA helps manage collaborative work on new features
  - Make use of "sub-tasks" and relationships
• Technical questions/problems. Better places to get help are:
  • dspace-tech mailing list
  • #dspace on IRC

• Suggestions/ideas not previously discussed
  • New ideas are always wanted, but...
  • Discuss them with the community first
  • Mailing lists, community surveys, IRC, conferences or user groups are good places to float new ideas
• Major 1.0 (big changes, database changes)
  – Minor 1.6 (smaller changes)
    • Sub-minor 1.6.2 (bug fixes)

• Upgrades routes:
  – If I’m on 1.3.2 and I want to get to 1.6.2
    • 1.3.2 -> 1.4
    • 1.4 -> 1.5
    • 1.5 -> 1.6.2
Release co-ordinator

- Co-ordinates the release
  - Sets deadlines
  - Manages processes
  - Public relations
  - Deciding vote (very rarely used)
  - Tries to encourage participation
How development works

• Weekly IRC meetings
  • Anyone welcome

• DSpace-devel email list

• Processes continue to change and improve:
  – Recent changes:
    • No partial features in trunk
    • No new features without supporting documentation
Post-development / Pre-release

- Developer Testing
  - Eliminate the obvious code bugs

- Community Testathon
  - Find user experience bugs, less obvious problems

- Bug Fix
  - Not just for committers - contribute here too

- Create release candidate
  - Start with x.y (e.g. 1.5)

- Repeat as necessary
  - Increment sub-minor part of version number
  - (e.g. 1.5.2 is the second bugfix release for 1.5)
DSpace email lists

- Send to an appropriate list
- Include enough useful information
- Leave it a while before prompting if no reply
- Please don’t email people directly
DSpace email lists

• Who lives in which house?

• DSpace General (dspace-general)
  • General community discussion around digital repositories and related applications

• DSpace Technical (dspace-tech)
  • Technical discussion/support around DSpace installation, configuration and operation

• DSpace Development (dspace-devel)
  • Discussion/support around developing DSpace
  • Automated notices and alerts from JIRA
  • Suggestions for new features/improvements
  • Release announcements and updates
There are two "rooms" dedicated to DSpace
1. #dspace: For all general DSpace Questions and Answers
2. #duraspace: For committer, developer meetings and other DuraSpace activities.

The easiest way to access the service is from:
http://webchat.freenode.net/
See also…

• The DSpace wiki resource page:
  • https://wiki.duraspace.org/display/DSPACE/DSpaceResources

• DSpace manual:
  •
    https://wiki.duraspace.org/display/DSPACE/DSpaceResources#DSpaceResources-DSpaceSystemDocumentation

• The DSpace course:
  • http://hdl.handle.net/2160/615
How can you play a part?

What are your interests?

What are your passions?

What could you do?
• Under the hood image:
Contact details

- Leonie Hayes
  - Research Repository Manager, The University of Auckland Library
  - l.hayes@auckland.ac.nz (1.5)

- Richard Jones
  - Head of Repository Systems, Symplectic Ltd.
  - richard@symplectic.co.uk (1.0)

- Stuart Lewis
  - IT Innovations Analyst and Developer, The University of Auckland Library
  - s.lewis@auckland.ac.nz (1.2)

- Monica Roos
  - Special Librarian, The University of Bergen Library
  - monica.roos@ub.uib.no (1.4)

- Kim Shepherd
  - IT Software Analyst and Developer, The University of Auckland Library,
  - k.shepherd@auckland.ac.nz (1.5)

- Elin Stangeland
  - Repository Manager, Cambridge University Library
  - es444@cam.ac.uk (1.2)