Release Manager Responsibilities

Release Managers are Assigned on a rotating basis to the Hydra Framework components.

<table>
<thead>
<tr>
<th>Component</th>
<th>Fall '12</th>
<th>Win '13</th>
<th>Spring '13</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hydra Head</td>
<td>Jessie Keck</td>
<td>Jessie &quot;Jeff&quot; Keck</td>
<td></td>
</tr>
<tr>
<td>Hydra Tutorial</td>
<td>Chris Beer</td>
<td>Steven Anderson</td>
<td></td>
</tr>
<tr>
<td>OM</td>
<td>Mike Stroming</td>
<td>Justin Coyne</td>
<td></td>
</tr>
<tr>
<td>Active-Fedora</td>
<td>Adam Wead</td>
<td>Chris Beer</td>
<td></td>
</tr>
<tr>
<td>Solrizer / solrizer-fedora</td>
<td>Matt Zumwalt</td>
<td>Naomi Dushay</td>
<td></td>
</tr>
<tr>
<td>Rubydora</td>
<td>Jeremy Friesen</td>
<td>Jeremy Friesen</td>
<td></td>
</tr>
<tr>
<td>Jetty Wrapper</td>
<td>Chris Beer</td>
<td>Chris Colvard</td>
<td></td>
</tr>
<tr>
<td>Hydra-Jetty</td>
<td>Eddie Shin</td>
<td>Carolyn Cole</td>
<td></td>
</tr>
<tr>
<td>Hydra-LDAP</td>
<td>Mike &quot;Still fine with that&quot; Giarlo</td>
<td>Dan &quot;of the Dans&quot; Coughlin</td>
<td></td>
</tr>
<tr>
<td>Jenkins</td>
<td>Ben Armintor</td>
<td>Chris Beer</td>
<td>N/A</td>
</tr>
<tr>
<td>Web Site (content)</td>
<td>Richard &quot;I don't add snark&quot; Green</td>
<td>Richard Green</td>
<td></td>
</tr>
<tr>
<td>Web Site (technical support)</td>
<td>Mark Bussey</td>
<td>Mark Bussey</td>
<td></td>
</tr>
<tr>
<td>Wiki gardener</td>
<td>Richard Green</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Committers Call Housekeeper</td>
<td>Michael Klein</td>
<td>Adam Wead</td>
<td></td>
</tr>
<tr>
<td>Technical Documentation</td>
<td>Steven Anderson</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Hydra project reporter</td>
<td>Jeremy Friesen</td>
<td>Jeremy Friesen</td>
<td></td>
</tr>
<tr>
<td>RightsMetadata-policyfinder</td>
<td>Eddie/Ben ?</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CLA manager</td>
<td>Richard Green</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

former quarters’ release managers

Release Manager Process Notes

Component Release Manager Duties

Version Numbers

We use semantic versioning:

- Major number bumps for incompatible changes (2.3.1 ==> 3.0.0)
- Minor number bumps for new features (2.3.1 ==> 2.4.1)
- Point number bumps for bugfixes (2.3.1 ==> 2.3.2)

NOTE: the version numbers of the different components in the Hydra stack (HydraHead, ActiveFedora, OM, Solrizer ...) are NOT synchronized.

Release Process

- major release process
- minor & patch release process

When to do a Release

- to fix a bug
- otherwise, no more often than once a week
- when a new feature is needed by a non-committer (someone who wouldn't want to pull from trunk)
- bigger releases (e.g. not backwards compatible) should not occur more often than once a quarter.

Responsibilities for Each Component's Release Manager

Barest minimum:

- participate in Hydra Committers calls
- ensure pull requests are dealt with in a timely fashion (a few days at most)
  - they must have tests
• code authors MUST have an iCLA in place (or a cCLA from their institution) before any code is committed on their behalf. See: CLA submission list. If no CLA exists, liaise with CLA manager to get one.
• do releases as indicated.
• Continuous Integration builds
  • if they break, ensure that they are fixed in a timely fashion
  • get rid of warning messages if possible (e.g. invalid YARD tag)
• Tests
  • must run cleanly for component
  • full Hydra stack tests must run cleanly.
  • make sure bugs aren't just fixed, they're tested

What we really want:
• track dependencies and schedules of other components as needed for synchronization
  • This is especially crucial for the HydraHead component manager
• Documentation
  • Review & Manage Documentation: make sure it is all current, sufficient, clear ...
    • API documentation
      • ensure API docs are built (currently by Hudson build, on Hudson box) (http://hudson.projecthydra.org/job/hydra-head-rails3-plugin/Documentation/)
      • doc files packaged with component code (e.g. README.textile)
    • Hydra-Head / Hydra framework documentation re: component
      • github wiki (https://github.com/projecthydra/hydra-head/wiki)
      • ditto project hydra wiki (this one)
      • ProjectHydra website (refer to web site component lead)
• Tests
  • must meet target percentage for test coverage
• Responsible for Inclusion/Exclusion of Code Changes
• Contributions from non-committers must be evaluated in a timely fashion
• Ensure contributors have CLA on file (contributor license agreement)
• Code Changes must
  • have test coverage
  • run cleanly for component, and change does not break full Hydra stack tests.
  • have documentation coverage (also yard/doc must cleanly compile - see above re API docs)
  • behave correctly
  • have readable code
• Creating a Release
  • Release Notes
    • must indicate the full release number, the date, and what has changed since the last release.
      • Github tickets and possibly git history are best ways to find this information
      • put into HISTORY.textile which is symlinked to RELEASE_NOTES; both in root directory of component.
    • tag in git?
      • send email to hydra-tech indicating the new release, its number, and what has changed. Send an email once a week (if there is a release that week)
        • Call out Bug Fix, New Feature, Major Release labels in the release announcement email.
        • See this Sample Release Notice Email Format
      • for MAJOR and MINOR releases, send an email to the hydra-releases@googlegroups.com list with the release notes.
      • for MAJOR RELEASES, provide details on release to Web Site Manager / Steering Group sufficient to generate a Press Release.
      • for MAJOR RELEASES, document upgrade path
• Find and train your successor Component Lead
• Support your successor component lead w/ tying up any loose ends

Web Site Manager Housekeeping Responsibilities
• overall maintenance of website
  • ensure a steady stream of news
  • provide notification of major software releases
  • maintain up to date information about partners and projects
  • periodically review site for outdated and/or inaccurate content
• regular patching of WordPress site to latest version (directly, or by liaising with Web site designated technical support)
• News
  • keep an ear to the ground for information that should become news (steering and partner call, lists, etc)
  • write up and post news (chasing content from individuals where necessary)
  • follow up news if it should become or affect page content
  • try to ensure a steady flow (at least something new each month)
• Software releases
  • Liaise with colleagues to provide web content around software releases (see above)
• Partners and projects
  • In collaboration with partners, ensure that all partners and formally announced projects are properly represented on the site
  • Ensure that pages containing information about partners and projects are kept up to date
  • Create new pages for partners and projects as appropriate (chasing individuals for information where necessary)

• Content
  • add new content when available (posts and pages), maintain menus as required
  • periodically review site for outdated or inaccurate content: liaise, as appropriate, to update, change or delete such
  • notify significant changes to the appropriate Hydra lists
  • regular full exports from WordPress as backup (after any significant change)

Committers Call Housekeeping Responsibilities

• prepare weekly agenda
  • in wiki: [https://wiki.duraspace.org/display/hydra/Notes+from+Meetings+and+Calls](https://wiki.duraspace.org/display/hydra/Notes+from+Meetings+and+Calls)
  • send email prior to meeting (preferably on Friday, or at least before Monday morning Pacific time) with weekly agenda inline
  • topics culled from
    • hydra-tech email since last call
    • IRC
    • JIRA
    • any leftover agenda items from previous call
    • any relevant items from other related meetings (hydra partners, combined blacklight/hydra committers calls ...)
  • always included:
    • call for items
    • schedule next call
      • choose next moderator
      • choose next notetaker
      • choose next JIRA wrangler (optional)
    • JIRA roundup for sprint ending on day of call

• JIRA
  • this is currently done by one Jira groomer for all components.
  • [https://jira.duraspace.org/browse/HYDRA](https://jira.duraspace.org/browse/HYDRA)
  • BEFORE call:
    • create JIRA version in HYDRA project for next weekly sprint
    • try to maintain an appropriate order for the versions
    • look at previous un-released sprints (before the one ending on day of call)
      • ensure all tickets are resolved and/or closed (you may need to move some or ask people or what-have-you)
    • do a JIRA "release": from manage versions, when you hover over version, on far right is setup wheel - one of the menu items is "release".
  • note that there are JIRA versions for the components and their releases: these are used in addition to the versions for the weekly sprint, and are not part of the Committers Call Housekeeping Responsibilities

tech documentation

• ensure the documentation for creating a hydra application from scratch is excellent
• ensure it is easy for folks to find the documentation they seek
• watch for questions on the hydra-tech list, and ensure the documentation reflects the information desired
• organized for sane newbie access
• invite newbies to smoke test installs, provide feedback to enhance docs and process
• keep track of what upgrade problems people are having and update the upgrade notes with that information