Planning 21.05 Release Testing

Developed plan based on results of 20.09 and 21.01 release testing:

- **20.09 key results:**
  - 21 tutorials (21 out of 17 planned)
  - Tool test results comparison (20.09 vs. previous runs)
  - Approx. 3 out of 350 release notes items
  - Time spent: approx. 2 weeks

- **21.01 key results:**
  - 24 tutorials
  - 30 out of 283 release note items
  - Time spent: approx 3 days
Recommendations from 21.01

○ Structure
  ○ More structured testing plan (protocol for what and how to test)
  ○ Open issues more aggressively (problems should not be postponed/forgotten/lost)

○ Too many items (283 in 21.01, 350 in 20.09): not possible to cover all manually
  ○ Don't have sense of getting things done (list doesn't get smaller)

○ Have to prioritize; usually not obvious how
  ○ Highlighted items were covered in the first place; the rest is hard to rank

○ Not clear how to test:
  ○ Waste time on figuring out how to test
  ○ Possible duplication of what's done by automated testing (lots!)
  ○ Not everything is or should be testable on main (e.g. required admin access)

○ Many items should not be manually tested
  ○ Identify the tests that require a human
21.05 Release Testing Plan

Scope

○ Release notes: use for (loosely-defined) requirements-based testing
  ○ Dev team curated list of issues
○ Pull requests that include instructions for manual testing (43 / 369)
○ GTN tutorials: use for scenario-based testing (core, UI, admin, science)

Approach

○ Specific testing protocol
  ○ When to open an issue (e.g. incorrect/outdated steps that conflict with the current UI)
  ○ What to ignore (e.g. outdated screenshot colors, UI elements, typos, grammar, style...)
  ○ How to verify a problem is relevant, verify against previous release, etc.
○ Open issues; do not fix bugs
○ Time commitment: ~2 full days
Results: Summary

- Tutorials: 22
- Release Notes: 9
- Opened issues: 19
- Blocking issues: 1
Results: GTN Tutorials

Covered by the Release Team

21.05 - Tutorials covered

- Galaxy-interface: 8 (36.4%)
- Admin: 12 (54.5%)
- Introduction (Core): 2 (9.1%)
Results: Release Notes Items

Covered by the Release Team out of 317

21.05 - Release notes

- Bug fixes: 44.4%
- Enhancements: 33.3%
- Highlights/Features: 22.2%
Results: Opened Issues

Opened issues by the Release Team: 19

Blocking issues: 1

- All issues opened in the Galaxy repo.
- No major issues found on tutorials but 2 PRs with typo fixes were opened.
Some Examples of Identified Problems

- **UI issues:**
  - Mostly some paper-cuts issues around Data Libraries and the Workflow Editor

- **Server errors**
  - Job handlers acting as workflow schedulers despite presence of explicit workflow schedulers
    - #11986

- **Tutorials**
  - Outdated screenshots, outdated and/or inconsistent steps
  - Changes to configs expressed with patch files
  - Minor inconsistencies in style between tutorials
  - Minor typos were fixed on the spot

- **Often discovered with irregular/exploratory input by tester**
  - Workflow Editor - Connectors and/or noodle maintain incorrect state
    - #12032
<table>
<thead>
<tr>
<th>#</th>
<th>Title</th>
<th>Description</th>
<th>Assignee</th>
<th>Milestone</th>
<th>Project</th>
<th>Label</th>
<th>Type</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Workflow Editor - Connectors and/or noodle maintain incorrect state</td>
<td>Workflows Editors Connectors and/or noodle maintain incorrect state.</td>
<td>2010</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Libraries exporting all datasets in current folder (as a collection) only exports the visible datasets at the current page</td>
<td>Libraries exporting all datasets in current folder (as a collection) only exports the visible datasets at the current page.</td>
<td>2010</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Invalid file upload causes missing file name in History panel</td>
<td>Invalid file upload causes missing file name in History panel.</td>
<td>2010</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Libraries: the select all checkbox when adding datasets to a library from history doesn't behave as expected</td>
<td>Libraries: the select all checkbox when adding datasets to a library from history doesn't behave as expected.</td>
<td>2010</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>Libraries: editing a library from page 1 displays the description of the library at the same row but at page 1</td>
<td>Libraries: editing a library from page 1 displays the description of the library at the same row but at page 1.</td>
<td>2010</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>Libraries: non-admin users with edit permission can not change the library name or delete</td>
<td>Libraries: non-admin users with edit permission can not change the library name or delete.</td>
<td>2010</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>Libraries: non-admin users with manage permission cannot really manage</td>
<td>Libraries: non-admin users with manage permission cannot really manage.</td>
<td>2010</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>Job handlers acting as workflow schedulers despite presence of explicit workflow schedulers</td>
<td>Job handlers acting as workflow schedulers despite presence of explicit workflow schedulers.</td>
<td>2010</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>Show all running jobs on Admin -&gt; jobs page</td>
<td>Show all running jobs on Admin -&gt; jobs page.</td>
<td>2010</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>Component property mutation</td>
<td>Component property mutation.</td>
<td>2010</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>RuleBuilder Nested List Tutorial Returns an Empty List</td>
<td>RuleBuilder Nested List Tutorial Returns an Empty List.</td>
<td>2010</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Testing Process Analysis

Tutorials
- No major issues found.

Release notes
- The Dev-team curated release notes greatly helped in focusing
- Automated classification of items based on PR template
  - Identify items that require manual testing and avoid those that provide automated tests
- Not always clear how to test:
  - The PR template helped a lot in many cases
  - Still some PR don’t fill out the required steps to test when there are no automated tests
  - Written by domain experts for domain experts
Testing Process Analysis

Improvements compared to 21.01:

- Much easier with curated list of issues
  - But we didn’t cover as many issues
- The PR template was really helpful to identify items that required manual testing
  - Some PRs don’t specify the manual testing steps
- Testing on AnVIL and GVL instances
  - Slows down the process
- Availability of VMs very helpful for Admin testing
- 100% focus on testing the release
<table>
<thead>
<tr>
<th>Release</th>
<th># People</th>
<th>Days</th>
<th>Notes</th>
<th>Tutorials</th>
<th>NPP/day</th>
<th>TPP/day</th>
</tr>
</thead>
<tbody>
<tr>
<td>20.09</td>
<td>3</td>
<td>14</td>
<td>3</td>
<td>21</td>
<td>0.071</td>
<td>0.5</td>
</tr>
<tr>
<td>21.01</td>
<td>5</td>
<td>2</td>
<td>30</td>
<td>24</td>
<td>3.000</td>
<td>2.4</td>
</tr>
<tr>
<td>21.05</td>
<td>4</td>
<td>2</td>
<td>9</td>
<td>22</td>
<td>1.125</td>
<td>2.75</td>
</tr>
</tbody>
</table>
Discussion of Ideas for Improvement

- Remove the Tool Tests from the plan as they are already automated
- Narrow down the Tutorial list to reduce overlapping when they touch similar parts of Galaxy
- Use of labels to simplify classification instead of parsing the PR body
- PR template merged: https://github.com/galaxyproject/galaxy/pull/11530
  - Asks for details on how to test + encourages writing accompanying tests
  - Screenshots for UI features!
- Idea: tag PRs that need special attention
  - [test: on main], [test: eu], [test: admin required] etc.
- Overall, a greater focus on automated testing
  - Automate tutorials with Selenium for testing?
    - Determine pre- and post-tutorial test VM state
  - Can we require tests for new features and fixed bugs?
    - Related long term goal: lower the bar for writing tests to make ^ possible
- Let's use Sentry to hunt for problems!
- Idea: side-by-side testing new/previous release of select items, as needed
  - Use VMs? Use test vs. main?
Ideas for Improvement: Automation

1. Release notes can be generated with a one liner if issues tagged `gh pr list --label test/manual` ...
2. Can filter by:
   --state merged
   --search milestone:21.05
   --search merged:
   --search created:
3. Can output JSON and format with a Go template
4. Add to Makefile so becomes part of release process
Ideas for Improvement: Process

2. Pick 21.09 team now
   - at least coordinator
   - rest of team at least two weeks before code freeze
3. Stress that team has been curated by PI’s
4. Automate team selection
   - scrape participant pool from WG spreadsheet
   - random.select N names
5. N based amount of testing to be done
   - can be guestimated based on `gh pr list` outputs
Discussion