

Swing Planning

Planning for the gt-swing module; Micheal has been working on consolidating what we have with a view to promoting swing to supported land...

- [Quality Assurance](#)
- [Document](#)
 - [RnD Target](#)
- [JLayeredMapPane](#)
 - [JList Emulation](#)
 - [Printing](#)
 - [Shared Interfaces with SWT](#)
- [Feedback from 2010](#)

Reference:

- [JIRA Swing Component](#)
- [GTRenderer](#)

Quality Assurance

1. Going through all of the current classes and doing general tidying refactoring, adding unit tests, culling unused or duplicate code, and updating javadocs. This is what I'm working on at the moment.

2. Tackle the current open jiras, particularly:

- Easier custom toolbar items in JMapFrame
- Avoiding redundant drawing in the map pane (when simple)
- Support for rotation
- Support for linked map panes.

2 and 3 might not be done in that order - more likely my usual random mix unless external constraints demand otherwise.

Document

Docs minimal:

- Single code example (the tutorials have this covered)

Docs recommended:

- class diagrams and a few code examples of the good bits

The existing docs have this covered:

- <http://docs.geotools.org/latest/userguide/unsupported/swing/index.html>

RnD Target

Recent Developments:

- Add DirectLayer to StreamingRenderer
- Refactor JMapPane to use MapContent and MapViewport

Interesting RnD topics:

JLayeredMapPane

JLayeredMapPane - class about 1/2 written. This is the idea of having multiple MapContent in a single JPanel and allow "background" rendering into backing images. Get back to JLayeredMapPane and bring it up to being able to do an animation example.

This will also involve some discussion about handling DirectLayers outside the renderer. With Andrea's assistance I added support for DirectLayers to StreamingRenderer a while ago but you won't want to go through the renderer for non-static graphics, especially if a high frame-rate is required.

JList Emulation

The ability to treat a map display as a normal JList

- JListModel - with an adapter to produce features for each of the objects
- JListSelectionModel

Printing

It would be good to package this up (this docs do cover it)

Shared Interfaces with SWT

- [GEOT-3906 GUI interfaces for gt-api](#)

Feedback from 2010

Feedback collected on gt-swing:

-  MapContext is deprecated; the adption to use the MapContent is progressing as we speak on trunk showing active maintenance of the library by the module maintainer.
-  JMapPanel can use a tutoiral on how to embeded it into your own application. The JMapFrame class is set up for tutorials; and does not serve as a good example itself (too many helper methods that are "off topic").
- The documentation is out of date:

<http://docs.geotools.org/latest/userguide/unsupported/swing/jmappane.html>

- Test case coverage is low; may be able to make the argument that the coverage by the tutorial code is more than sufficient to keep the API stable
- JMapPane handles a single background MapContent; it would be good to use a second BufferedImage for draw quick feedback on (such as feedback from tools).
Motivation: This would prevent the entire map needing to redraw to show selection.
- Ability to change the icons for the Actions would be nice; this is a low priority

Trvial feedback (raised as issues in the issue tracker) the following issues were identified that made coding more verbose than needed:

- ⚠️ Unable to smoothly ask the map to redraw itself in response to selection changes; since we do not have a layer list this was difficult. The JMapComponent needs a method to accept MapChangedEvents and trigger the appropriate redraw activity.

This was actually amusing as we would call JComponent.repaint() and get the same map generated as the component draws into a back buffer).

- ✅ MapMouseEvent ReferencedEnvelope based on number of pixels: <https://jira.codehaus.org/browse/GEOT-3715>
Update this issue has been closed (2 day response time).
- ⚠️ Relative paths for ExternalGraphics (ie Icon references) is not well supported

Options (these are alternatives):

- Preprocess Style using a visitor to rewrite relative paths
- Update ImageGraphicFactory or SLDStyleFactory to handle relative file paths
- Use env variable subst: so `\${base}\face.png` would refer to the SLD document base
- ⚠️ Small trouble sharing data between base map and selection layer
<https://jira.codehaus.org/browse/GEOT-3716>