

Render Module

Module:	Render
Module Maintainer:	Jesse Eichar
Status:	★★★★★
Email Help:	Geotools-gt2-users@lists.sourceforge.net
Volunteer:	geotools-devel@lists.sourceforge.net
IP Review:	review.txt

Gold Star Quality Assurance Check

- ★ IP Check: review.txt added, all headers are in place
- ★ Releasable: no blocking issues, but feedback from potential users is appreciated
- ★ Quality Assurance: we think this is quite good; but we will get back to you later with a number
- ★ Stability: No planned MapContext API changes
- ★ Supported: Documentation available, module maintainer does watches user list, answers email.

Recent Development

For the next release Renderer has been effected by two changes:

- Code Moved: Some of the data structures used by rendering have been moved from main into this module
- Expression Improvements - XPath: we can finally support XPath - however performance will be different so give us feedback
- Expression Improvements - Objects: Expressions can now run against more then just Features, this means you can finally just draw normal Java Beans. We created a Data Access proposal (so renderer would know how to ask for your objects) but it was poorly received and will be removed from the 2.4 release - if you have any suggestions on how we can make this kind of functionality available to you please speak up. Right now you can make a layer with just a Collection of Objects (no fancy query supper without DataAccess)

There are also tips and tricks that should be noted:

- you will **need** at least one epsg-xxxxx jar before anything will draw

If you are a volunteer we could use help improving the following:

- Patches - found a bug got a fix? Send it our way
- TestCases - found a bug? Prove it! Send us the test case
- Issue Reports - got a request? Please add it to the issue tracker or show up at a weekly meeting

Outstanding Issues

The module maintainer will take care of assigning these issues to developers:

Error rendering macro 'jiraissues' : Unable to determine if

sort should be enabled.

Remember that the module maintainer is a volunteer, and if you can help test this module please jump on the developers list and help out. Patches may be attached to the above issues for review by the module maintainer.

Although not the responsibility of the geotools module maintainer additional or related issues may be found in the the geoserver or udig bug trackers.

Module Status

The following changes are planned - details as we figure it out (volunteer!)

Labelling Improvements

There are a couple of bug reports concerning Labeling; it looks like they will only be addressed by changing the LabelCache strategy object. This is a very low-level change so most casual users will not be effected.

Summer of Code

Several Google [Summer of Code 2007](#) projects will be working with the data model (ie MapContext) defined by the renderer package. These projects may extend what MapContext can do over the next couple of releases.

When considering this functionality we will look at the following for inspiration/naming:

- Web Map Context - this specification is limited to WMS Layers but does contain useful suggestions
- OWS Context - this specification is a better fit (supporting WMS, WFS, WCS) and has some measure of extensibility for additional resources
- uDig per Layer Rendering - the uDig project separates out the MapContext from the strategy used to render individual "layers". Each layer brings together a "resource" and a "style" and extension point is processed to choose the best renderer for the job. The approach is exactly the same as how the current StreamingRenderer delegates to the GridCoverageRenderer; the difference is the decision is not hard coded.
- uDig graphics & composition: the uDig project has a separate interface for the backend (allowing the switch of Graphics2D, SWT). This allows layer composition methods to be tried (JAI vs a for loop vs drawing onto textures rendered by JOGL).