Griffon

"Griffon is a Grails like application framework for developing desktop applications in Groovy."

What is Griffon?

- framework for developing Swing apps

- Groovy based, Grails inspired

- extensible by plugins

- MVC

- convention over configuration

- aims to make build Swing apps easy!

State of Swing

- very verbose to get it right

```java
public void actionPerformed(ActionEvent e) {
    (new Thread(new Runnable() {
        public void run() {
            final String result = longRunningOperation();
            SwingUtilities.invokeLater(new Runnable() {
                public void run() {
                    label.setText(result);
                }
            });
        }
    })).start();
}
```

- SwingWorker helps with the pattern

```java
final JLabel label = ... ;
class MeaningOfLifeFinder extends SwingWorker<String, Object> {
    @Override
    public String doInBackground() {
        return findTheMeaningOfLife();
    }
    @Override
    protected void done() {
        try {
            label.setText(get());
        } catch (Exception ignore) {
        }
    }
}
(new MeaningOfLifeFinder()).execute();
```

Groovy helps a lot

- SwingBuilder DSL for fluent syntax

- shortened forms and simplified configuration

  - Java: JPanel panel = new JPanel();
    new BoxLayout(panel, BoxLayout.X_AXIS)

  - Groovy: panel { boxLayout() }

- map coercion VS constructors

- huge reduction in boilerplate

- simple Binding

Griffon helps even more

- CompositeBuilder: combines numerous builders into one common dsl

- IGriffonApplication: provides a common infrastructure

- meta-enhancement: additional methods added to Swing components at run-time

- makes publishing and launching easy

Builders

- SwingXBuilder - Swing component extensions

- FxBuilder - JavaFX script support

- JideBuilder - Swing component extensions

- SwingXtrasBuilder - aggregate of 3 smaller Swing component extensions(I2fprod, BalloonTip, xswingx)

- CSSBuilder - Swing Clarity: adds cssClass property to all Swing nodes

- TridentBuilder - animation library

- GfxBuilder - Java 2D(without pulling your hair out)

App Configuration

- Your A, B, C's

    - Application.groovy- title, root MVC group to initialize on startup, all MVC groups registered in the app

    - Builder.groovy - CompositeBuilder configuration

    - Config.groovy - logging, dev/test/prod environments, additional build script or lifecycle event configuration

App Lifecycle

- Initialize - before any Swing code is realized; used for L&F and setup

- Startup - after startup MVC groups are initialized

- Ready - all events sent to EDT queue have been processed

- Shutdown - opportunity to free resources before exit

- Stop - Applet specific

Distribution and Installation

- stand-alone, webstart and applet support

- create launchers for Windows(.exe), Linux(shell script), Mac(.app) and as executable jar

- create installers using IzPack, RPM and DMG

- 'big jar' support for those lucky people in the US who can use the Java Store

- can create all installers and launchers in one step

Testing and Analysis

- FEST – framework for Swing integration testing; equivalent of Selenium for Web testing

- EasyB – scenario based testing

- Cobertura – code coverage metrics

- JDepend – static analysis

Language support

- Scala – plugin available

- Clojure – plugin available

- Java - if you REALLY want :)

The EDT in Groovy

```
doOutside{
  //long running operations in a new
  thread
  edt {
    //jump back to the EDT if needed
  }
  doLater{
    //update the UI when we're done
  }
}
```

EDT With SwingWorker in Griffon

```
jxwithWorker {
  onInit {
    //setup
  }
  work {
    //do long running work and publish intermediate results
    publish(...)
  }
  onUpdate {
    //respond to intermediate results
  }
  onDone {
    //update UI with final results
  }
}
```

How to get Griffon?

- download the izpack installer

- macports

- install from source - set $GRIFFON_HOME and put $GRIFFON_HOME/bin in your path, there's an ant target to install

- tonite's demo was 0.2-SNAPSHOT built from source

How to use Griffon?

On the command line

- create/run/test Applications

- create MVC groups or their individual components

- create tests

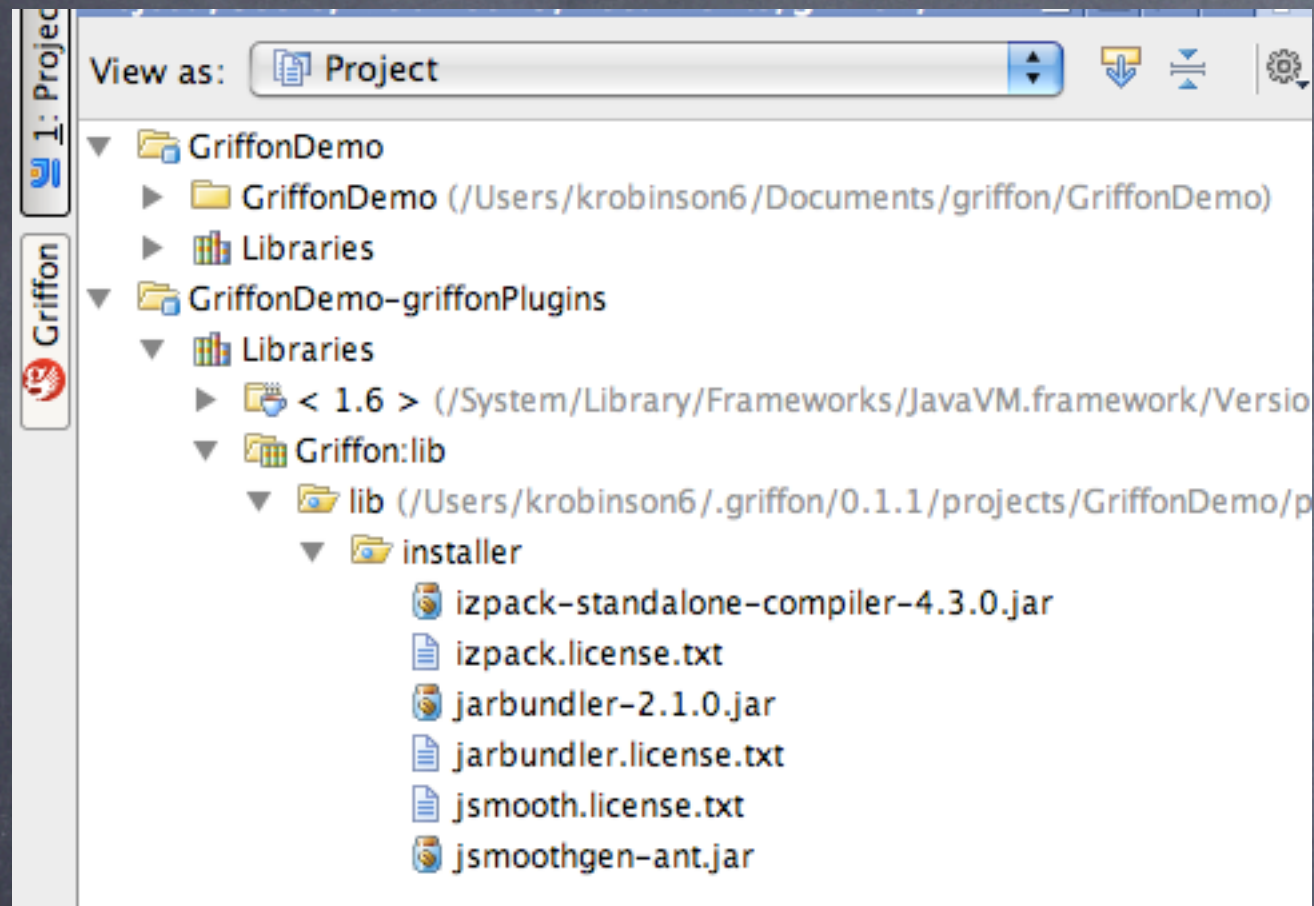- install/uninstall plugins
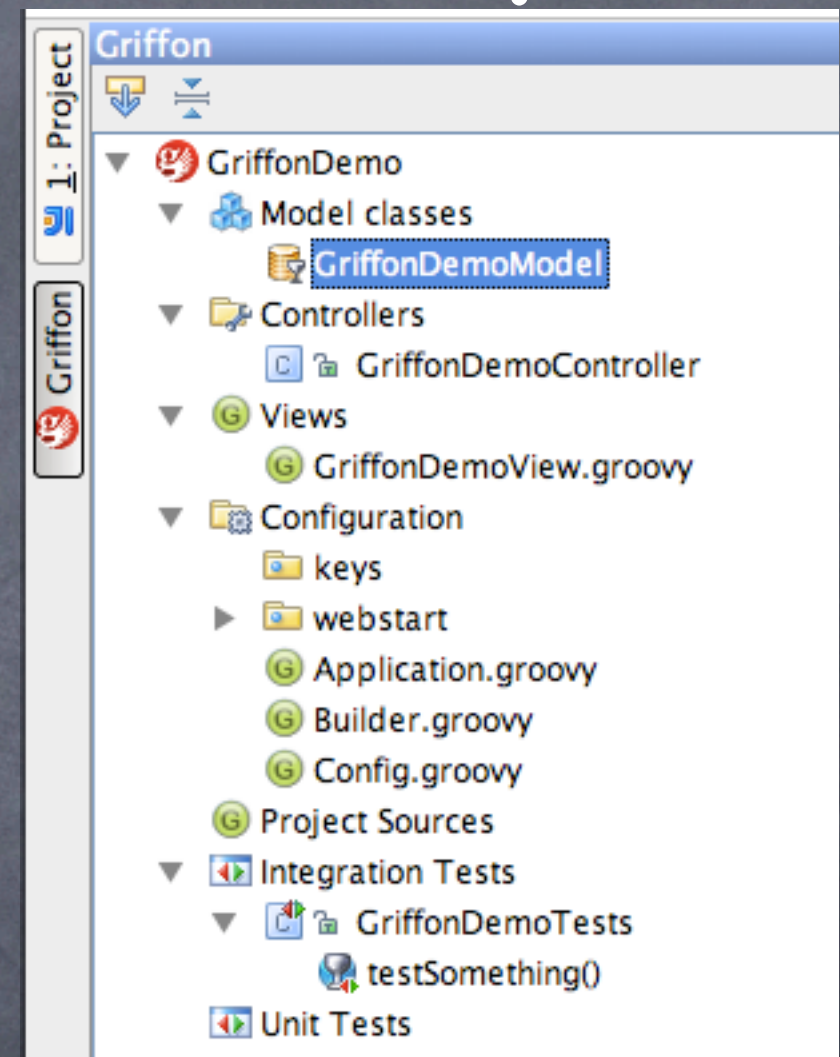
- create your own plugins

Tool support

- Netbeans - Geertjan Wielenga crafted an alpha version plugin for Java One this year

- Intellij - Maia (version 9, in EAP now) parallels existing Grails support

- Eclipse - Groovy plugin, no specific support for Griffon; yet

- Griffon in Action has instructions for each of these IDE's without plugins, based on ant

IntelliJ Maia Closeup

Project view
showing libraries/
plugins

Griffon MVC
plugin view

What's to come?

- Spring plugin

- Maven

- Gradle

- Grails plugin ports

- Griffon In Action from Manning Books

How much code does it take?

Name	Files	LOC
Controllers	4	112
Models	4	26
Views	4	122
Lifecycle	5	76
Integration Tests	1	4
Totals	18	340

Questions

- ???