

Scala Plugin

Description

The Scala plugin enables compiling and running Scala code on your Griffon application. Scala is a statically typed language for the JVM that has a great level of interoperability with Java, and so with Groovy too.

Installation

The current version of **griffon-scala-plugin** is **0.8**
To install just issue the following command

```
griffon install-plugin scala
```

Usage

You must place all Scala code under `$appdir/src/scala`, it will be compiled first before any sources available on `griffon-app` or `src/main` which means you can't reference any of those sources from your Scala code, while the Groovy sources can. However in order to send a message back to the calling code you can create a Scala Trait (roughly equivalent to a Java interface) and have a Groovy class implement it. Another alternative would be the usage of Structural Types.

Starting with version 0.3 you will be able to use the [LangBridge Plugin](#) facilities to communicate with other JVM languages.

Scripts

- **scala-test** - runs Scala tests found in `$basedir/test/scalatest` using Scalatest 1.0. Test classes must have `Tests` as suffix.
- **scala-check** - runs specifications found in `$basedir/test/scalacheck` using Scalacheck 1.6. Test classes must have `Specification` as suffix.
- **scala-repl** - runs a Scala REPL with the application's classpath fully configured.

Configuration

The `BuildConfig.groovy` file in `griffon-app/conf` folder of your application can be used to configure behavior of the scala plugin.

```
scala.src.encoding='UTF-8' //Scala src file encoding, UTF-8 by default
```

The following properties can be configured for `scala-test`

- **scala.test.parallel** - run tests in parallel. Default: false.
- **scala.test.numthreads** - set the number of threads for parallel execution.
- **scala.test.haltOnFailure** - whether to stop tests on failure or not. Default: false.
- **scala.test.fork** - fork the jvm. Default: false.
- **scala.test.maxmemory** - maximum memory to be used by test jvm.
- **scala.test.membersonly** - specify suites using members-only or wildcard package names.
- **scala.test.wildcard** - specify suites using members-only or wildcard package names.
- **scala.test.reporter.stdout** - sets the flags for the stdout reporter. Default: FAB.
- **scala.test.reporter.file** - sets the name for a file reporter.
- **scala.test.reporter.reporterClass** - sets the className for a reporter class.
- **scala.test.scalaConfig** - Map with additional properties.

The `scala-test` script supports the following command line flags:

- **stdout** - overwrite the flags for the stdout reporter.
- **file** - sets the name for a file reporter.
- **reporterClass** - sets the className for a reporter class.
- **suite** - only executes the specified suite.

The `scala-check` script supports the following command line flags:

- **spec** - only executes the specified specification.

History

Version	Date	Notes
0.8	04-01-11	Release sync with Griffon 0.9.2
0.7.1	08-24-10	Avoid overwriting default target accidentally
0.7	07-22-10	Release sync with Griffon 0.9, Upgraded to Scala 2.8.0, Scalatest 1.2, Scalacheck 1.7
0.6	03-01-10	Upgraded to Griffon 0.3
0.5.1	11-17-09	Updated bundled libs to Scala 2.7.7.final
0.5	11-15-09	Added Scalatest 1.0, Scalacheck 1.6 and REPL (2.7.6) support
0.4	09-10-09	Updated bundled libs to Scala 2.7.5.final. Check timestamp of compiled sources to decide if recompile is needed
0.3	07-28-09	Added dependency to LangBridge Plugin
0.2	04-02-09	Synchronized features with grails-scala-plugin 0.3
0.1.1	03-09-09	Bumped version to be compatible with Griffon 0.1.0
0.1		