Generate UML class diagrams from your Grails app source code.

UML Class Diagram Plugin - Reference Documentation

Authors: Igor Rosenberg

Version: 0.3.3

Table of Contents

1 Introduction

This plugins allows to generate UML diagrams for your Grails application.

Rendering is done via the integrated plantUML:http://plantuml.sourceforge.net/ library (without sending da

The diagrams can be generated when the webapp is running

```
http://server:port/webapp/uml
```

Or before runtime

```
grails uml-class-diagram
```

The plugin is released under the Apache License 2.0 license, source code is available .

The data shown in the UML class diagrams is configurable.

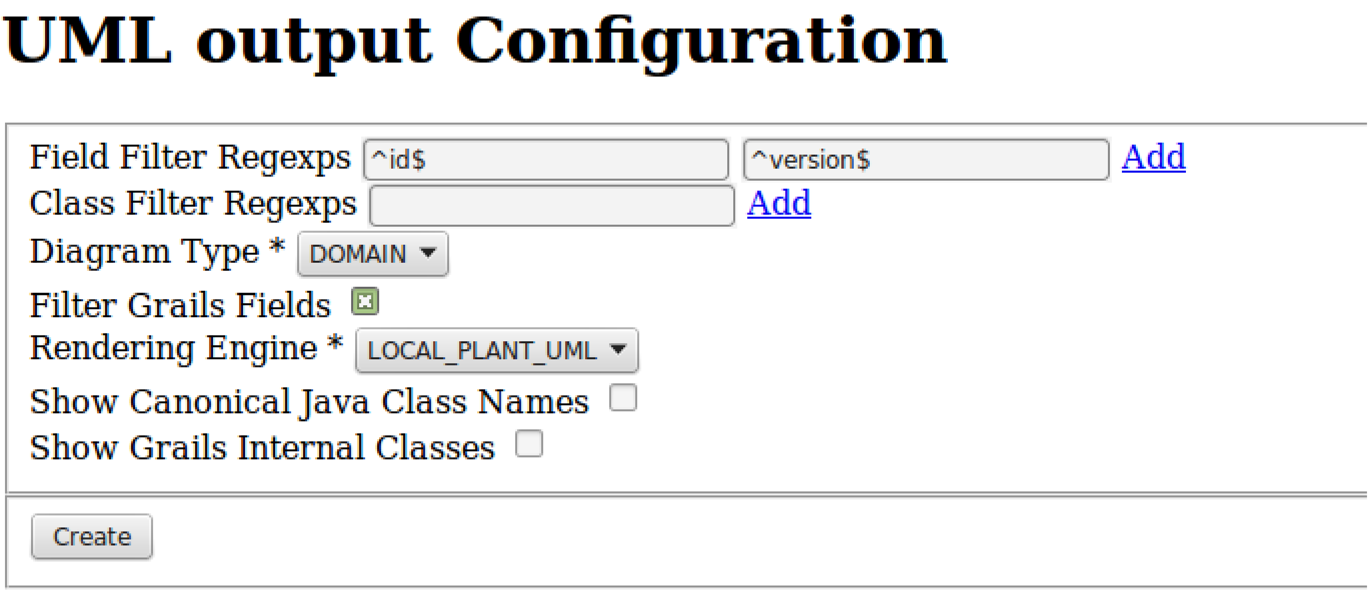
2 Online

When the webapp is running, the UML diagrams are exposed through the following URL:

```
http://server:port/webapp/uml
```

HTML Form

An HTML form allows to configure the diagram data:



UML output Configuration

Field Filter Regexps ^id$ ^version$ Add

Class Filter Regexps Add

Diagram Type * DOMAIN

Filter Grails Fields ☑

Rendering Engine * LOCAL_PLANT_UML

Show Canonical Java Class Names □

Show Grails Internal Classes □

HTTP access

The diagrams are available without using the online form, for example:

```
curl -v -H "Content-Type: application/json" -d '{
   "fieldFilterRegexps"=["^id$","^version$"],
   "classFilterRegexps"=[".*City"],
   "diagramType"="DOMAIN",
   "showCanonicalJavaClassNames"="false"
}' http://server:port/webapp/uml/draw
```

3 Offline

The UML diagrams can be generated before runtime

```
grails uml-class-diagram
```

4 Configuration

The data shown in the UML class diagrams is configurable.

Configuration options

- `fieldFilterRegexps`: a list of String filters (interpreted as regular expressions) restricting the classes, for example '^id$'
- `classFilterRegexps`: a list of String filters (interpreted as regular expressions) restricting the example '.*Race'
- `showCanonicalJavaClassNames`: boolean, removing the package prefix to Java classes (from
- `showGrailsInternalClasses`: boolean, removing Controllers and Services provided by the G
- `filterGrailsFields`: boolean
- `diagramType`: either 'DOMAIN' (uml class diagrams of the Grails Domain objects) or 'LAYERS the Grails Controllers and Services)
- `renderingEngine`: either 'LOCAL_PLANT_UML' (generate diagrams via embeded 'WEB_PLANT_UML' (generate diagrams via online plantUML service)

Developed by Igor Rosenberg.