

# Configuring custom jexl java expressions

|                           |                                               |                                |                                          |                                         |                                              |
|---------------------------|-----------------------------------------------|--------------------------------|------------------------------------------|-----------------------------------------|----------------------------------------------|
| <a href="#">Wiki Home</a> | <a href="#">Grouper Release Announcements</a> | <a href="#">Grouper Guides</a> | <a href="#">Grouper Deployment Guide</a> | <a href="#">Community Contributions</a> | <a href="#">Internal Developer Resources</a> |
|---------------------------|-----------------------------------------------|--------------------------------|------------------------------------------|-----------------------------------------|----------------------------------------------|

[JEXL is expression language in Java.](#)

Depending on the API and what the JEXL is used for, the call might allow static class references. For example PSPNG allows this.

First you need to compile a class. You can do this outside of Grouper, or checkout the grouper code or make a maven dependency or compile in the container if you like.

This example uses Grouper libraries, so lets just make a simple maven dependency in eclipse

pom.xml

```
<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 https://maven.apache.org/xsd/maven-4.0.0.xsd">
  <modelVersion>4.0.0</modelVersion>
  <groupId>sampleJexl</groupId>
  <artifactId>sampleJexl</artifactId>
  <version>0.0.1-SNAPSHOT</version>
  <dependencies>
    <!-- https://mvnrepository.com/artifact/edu.internet2.middleware.grouper/grouper -->
    <dependency>
      <groupId>edu.internet2.middleware.grouper</groupId>
      <artifactId>grouper</artifactId>
      <version>2.5.35</version>
    </dependency>
  </dependencies>
  <build>
    <sourceDirectory>src</sourceDirectory>
    <plugins>
      <plugin>
        <artifactId>maven-compiler-plugin</artifactId>
        <version>3.8.1</version>
        <configuration>
          <source>1.8</source>
          <target>1.8</target>
        </configuration>
      </plugin>
    </plugins>
  </build>
</project>
```

Make a class

```

package sampleJexl;

import edu.internet2.middleware.grouper.util.GrouperUtil;

public class SampleClass {

    public static void main(String[] args) {

    }

    public static String tableauADGroupName(String groupName) {
        /* Strip contextual stuff */
        String tableauName = groupName.replace("app:tableau:org:", "")
            .replaceAll(":roles:", ":")
            .replaceAll(":protected:", ":")
            .replaceAll(":projects:", ":");

        /*
            convert some:prefix:blah:blah_suffix
            to just some:prefix:blah_suffix
            That is squeeze out the last folder before the group.
        */
        /* by indexes */
        /*
            int lastColon = tableauName.lastIndexOf(":");
            int prevColon = tableauName.lastIndexOf(":", lastColon - 1);
            tableauName = tableauName.substring(0, prevColon) + tableauName.substring(lastColon);
        */
        /* With grouper util methods */
        String parentFolder = GrouperUtil.parentStemNameFromName(tableauName);
        String grandParentFolder = GrouperUtil.parentStemNameFromName(parentFolder);
        String groupSuffix = GrouperUtil.suffixAfterChar(tableauName, ':');
        tableauName = grandParentFolder + ":" + groupSuffix;
        /* _ and : turn into - */
        tableauName = tableauName.replaceAll("[_:]", "-");
        /* Take the last 48 chars, or the whole string,
            so the full group name is <= 64 chars as required by AD
        */
        int cutoff = Math.max(tableauName.length() - 48, 0);
        tableauName = tableauName.substring(cutoff);
        /* Add the prefix */
        tableauName = "Grouper-Tableau-" + tableauName;
        return tableauName;
    }
}

```

Test it

Note, to invoke static methods, put the fully qualified package name, classname, and method name. In this case the package name is sampleJexl, the classname is SampleClass, and the method is tableauADGroupName

```

public static void main(String[] args) {

    // Grouper-Tableau-Enterprise-HR-LaborRelations_project_leaders
    String result = GrouperUtil.substituteExpressionLanguage(
        "${sampleJexl.SampleClass.tableauADGroupName('app:tableau:org:Enterprise:HR:projects:LaborRelations:roles:LaborRelations_project_leaders')}}",
        null, true, false, false);

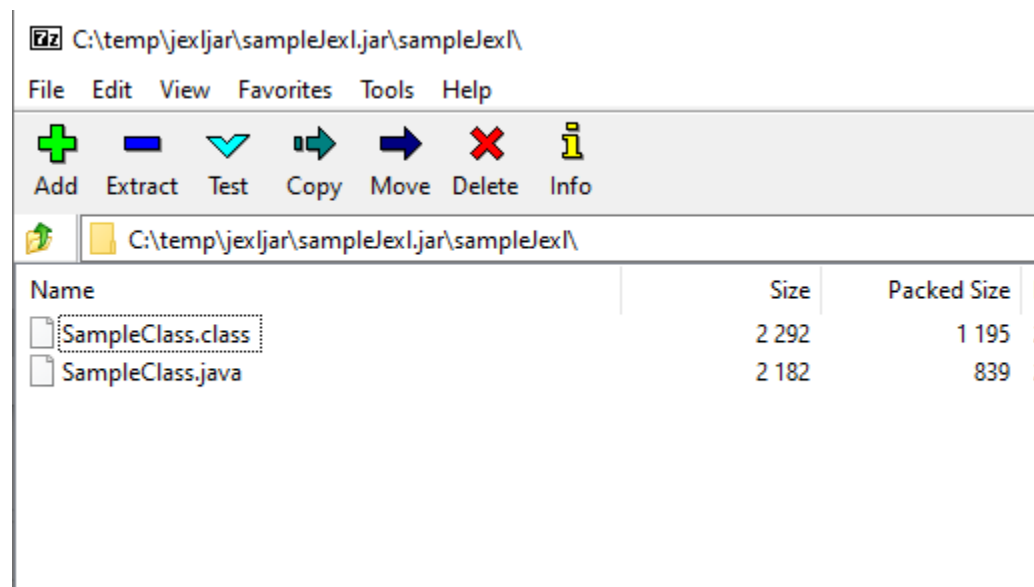
    if (!"Grouper-Tableau-Enterprise-HR-LaborRelations-project-leaders".equals(result)) {
        throw new RuntimeException(result);
    }

    System.out.println("Success");
}

```

Im just going to copy the classfile to a dir and source and make a zip, rename to jar

```
mchyzer@ISC20-0637-WL:/mnt/c/temp/jexljar$ find .
./sampleJexl
./sampleJexl/SampleClass.class
./sampleJexl/SampleClass.java
mchyzer@ISC20-0637-WL:/mnt/c/temp/jexljar$
```



Add this jar to slashRoot/opt/grouper/grouperWebapp/WEB-INF/lib or put in subimage