

MyFaces Developer Notes

{scrollbar}

This document deals with issues that are important for MyFaces developers and custom component developers.

Component developer guidelines

(same as "SVN guidelines", see below)

SVN guidelines

Code style

See <http://www.apache.org/dev/styleguide.html> for the Apache-Wide Coding Style Guide. Everything that is not documented there should be written according to Suns Java Code Conventions at <http://java.sun.com/docs/codeconv/html/CodeConvTOC.doc.html>

Here is a formatter profile for eclipse 3.1: [apache-codeconv.xml](#) . Import it through Window->Preferences->Java->Code Style->Formatter

Here is a formatter profile for IntelliJ IDEA 7.0 (7718): [maven-idea-codestyle.xml](#). Import it through File->Import Settings ... This should add a new profile called "MyFaces"

Error and exception handling

Whenever an exception is caught, the following applies:

- If there is no chance to get the system back to a stable state, a message with severity "fatal" should be logged and a RuntimeException (or derived class) should be thrown. Vice versa: Whenever a RuntimeException (or derived class) is thrown from within a catch clause, a message with severity "fatal" must have been logged right before.
- If the exception is severe, but there is a chance to continue processing, a message with severity "error" or "warning" should be logged.

Logging

Except in the JSF API (javax.faces.*) classes where there must be no dependencies to additional libraries, commons-logging is used for logging. Commons-logging should be used in the recommended way, i.e. each class has its own private static logger.

Naming

Naming is a fundamental concern. The name of a class, method or field should tell as much as possible of it's meaning and usage. Avoid short names like "Helper", "calc" or "check". Use "FinancialConsultance", "calculateMyYearlyIncome" and "isMyIncomeTooLow" instead.

Special comments

Addition special comments (see 10.5.4 in CodeConv)

- **TODO:** something that is not yet ready
- **HACK:** something that is bogus but (meanwhile) necessary to make it work
- **XXX:** something that is bogus but works
- **FIXME:** something that is bogus and broken

Version Control

- Every developer may upload files, if (s)he is confident, that this file is usable for others.
- Uploaded java files must be compilable, of course.
- Changes to java files must leave the whole project in a state where it is still compilable.
- Foreign files should only be committed (changed) after consulting and/or informing the original author. The same applies for deleting, of course.
Exception: Trivial changes such as spelling mistakes in comments, etc.
- Just to be sure: Don't forget the message(s) on commit 😊

Communication

- Main communication instrument among !MyFaces developers is the !MyFaces developer mailing list (dev@myfaces.apache.org).
- Every active !MyFaces developer must subscribe to this list.

Copyright

Every file should start with the following copyright notice:

```
java /* * Licensed to the Apache Software Foundation (ASF) under one * or more contributor license agreements. See the NOTICE file * distributed with this work for additional information * regarding copyright ownership. The ASF licenses this file * to you under the Apache License, Version 2.0 (the * "License"); you may not use this file except in compliance * with the License. You may obtain a copy of the License at * * http://www.apache.org/licenses /LICENSE-2.0 * * Unless required by applicable law or agreed to in writing, * software distributed under the License is distributed on an * "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY * KIND, either express or implied. See the License for the * specific language governing permissions and limitations * under the License. */
```

Class and Interface comment

Every class and interface should have a javadoc similar to this:

```
java /** * Very detailed description goes here... ;-) * * @author Bug Rogers */
```

SVN Settings

Developers should be setting their autoprops so that files have an svn:eol-style of 'native'. Developers working on *nix systems are "lucky" because native on *nix is what SVN calls native, but the autoprops should still be set so that keywords are replaced as well. This is described on the [Struts SVN wiki page](#).

Issue Tracker guidelines

Closing issues

Issues should be marked as closed at the same time that an issue is marked as resolved.

{scrollbar}