

# Release Notes 2.0.0

[Struts 2.0.0](#) is the first distribution of WebWork 2 as an Apache Struts product.

- If you are a Maven user, you might want to get started using the [Maven Archetype](#).
- Another quick-start entry point is the **blank** application. Rename and deploy the WAR as a starting point for your own development.

## Known Issues

- For known issues, see [Struts 2.0.1](#)

## Key Snippets

Maven Artifact ID

```
<dependency>
  <groupId>org.apache.struts</groupId>
  <artifactId>struts2-core</artifactId>
  <version>2.0.0</version>
</dependency>
```

Struts Configuration XML

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE struts PUBLIC
  "-//Apache Software Foundation//DTD Struts Configuration 2.0//EN"
  "http://struts.apache.org/dtds/struts-2.0.dtd">
<struts>
  <include file="struts-default.xml"/>
  <package name="support" extends="struts-default">
    <action name="$name" class="$class">
      <result>$location</result>
    </action>
  </package>
</struts>
```

XWork Validation XML

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE validators PUBLIC
  "-//OpenSymphony Group//XWork Validator 1.0.2//EN"
  "http://www.opensymphony.com/xwork/xwork-validator-1.0.2.dtd">
<validators>
  <field name="$name">
    <field-validator type="$type">
      <message>$message</message>
    </field-validator>
  </field>
</validators>
```

## New Features and Extensions

- [Can we use Acegi Security with the framework](#) extension is available.
- [CheckBox Interceptor](#) detects unchecked checkboxes and adds a parameter with a default value (usually 'false').
- [JavaServer Faces](#) extension is available.
- A `struts.xml` file can be placed in a JAR and automatically plugged into an application, so that modules can be self-contained and automatically configured.
  - In the case of Freemarker and Velocity modules, the templates can also be loaded from the classpath, so the entire module can be plugged in as a single JAR.
- [DebuggingInterceptor](#) displays the value stack at runtime.

## Key Changes

Depending on whether you are coming from a Struts 1 or a WebWork 2 background, your perspective on the key changes will differ.

### From Struts 1

- See [Comparing Struts 1 and 2](#)
- ⚠ More notes to come ...

### From WebWork 2

- Struts 2 is dependant on XWork 2 (nightly build).
- Java 1.5 is required to build and deploy Struts 2.
  - For Java 1.4 deployments, RetroWeaver or RetroTranslator may be used to provide Java 1.4 support
  - A pre-packaged set of RetroTranslator JARs is being provided with the distribution.

For other changes from WebWork 2, see the "Members to rename", "New property settings", "New features or feature changes", and "Removed or deprecated".

### Members to rename

WebWork 2	Struts 2
<code>com.opensymphony.xwork.*</code>	<code>com.opensymphony.xwork2.*</code>
<code>com.opensymphony.webwork.*</code>	<code>org.apache.struts2.*</code>
<code>xwork.xml</code>	<code>struts.xml</code>
<code>webwork.properties</code>	<code>struts.properties</code>
<code>DispatcherUtil</code>	<code>Dispatcher</code>
<code>com.opensymphony.webwork.config.Configuration</code>	<code>org.apache.struts2.config.Settings</code>

The tag prefix conventions in the example applications have changed.

<b>JSP</b>	s:	<code>&lt;s:form ...&gt;</code>
<b>Freemarker</b>	s.	<code>&lt;@s.form ...&gt;</code>
<b>Velocity</b>	s	<code>#sform ( ... )</code>

### New property settings

<b>struts.enable. DynamicMethodInvocation</b>	Enable support for the hardwired <code>action!alias</code> syntax (Default is <code>TRUE</code> )
---	---



It is recommended that all applications specify the `DynamicMethodInvocation` setting for now, since the default may change. The recommended approach to action aliasing is to use [wildcard aliases](#) instead of the hardwired bang.

### New features and feature changes

- Various changes to `ConfigurationManager`
  - `ConfigurationManager` is no longer a static factory. It is now an instance created through `Dispatcher`. Custom configuration could be done through `DispatcherListener`.
  - Custom configuration to `ConfigurationManager` and `Configuration` cannot be done statically anymore, instead use `Dispatcher`'s `DispatcherListener` to achieve the same effect.
- The `prepare` interceptor now uses reflection to call `prepare_Method_` where *method* is the action method configured for the particular action in `struts.xml`.

```

<action name="myAction" method="insert" ....>
    ....
</action>

// with the above configuration, and PrepareInterceptor in play,
// callsequence will be
1] prepareInsert() (If it exists)
2] prepare() (Unless Interceptor's alwaysInvokePrepare parameter is set to false)
3] insert()

```

- DefaultWorkflowInterceptor (named workflow in struts-default.xml) now uses reflection to call validateMethod on the action class that implements Validateable interface where method is the action method configured for the particular action in struts.xml.

```

<action name="myAction" method="insert" ...>
    ...
</action>

// with the above configuration, and DefaultWorkflowInterceptor in play,
// call sequence for action that implements Validateable interface will be
1] validateInsert()
2] validate() (unless Interceptor's alwaysInvokeValidate parameter is set to false)
3] insert()

```

- DatePicker tag is now using dojo's (limited in terms functionality and internationalization)
- [Tiles integration](#) extension is available.
- [Wildcards](#) can be specified in action mappings.
- [MessageStoreInterceptor](#) is introduced to allow field errors / action errors and messages to be store and retrieve through session, resulting them to be preservable across request.


## Removed or deprecated

<b>AroundInterceptor</b>	The AroundInterceptor has been removed. If your application extends the AroundInterceptor, either import the class into your source code form WebWork 2 (pursuant to the Open Symphony License) and modify it to server as your own base class, or rewrite your interceptor.
<b>oldSyntax</b>	Support for the "oldSyntax" is removed.
<b>Rich text editor tag</b>	<p>Rich text editor tag has been removed (a possible replacement would be to use the textarea tag with theme="ajax", this will used dojo's rich text editor)</p> <pre>&lt;s:textarea theme="ajax" /&gt;</pre>
<b>doActionMethod</b>	<p>The convention of trying a "do" form of an action method is not supported.</p> <pre> &lt;action name="..." method="submit"&gt;     ... &lt;/action&gt; </pre> <p>In WebWork,</p> <ul style="list-style-type: none"> <li>• try to execute submit method in the action, <i>fail</i></li> <li>• try to execute doSubmit method in the action if Step 1, <i>fail</i></li> <li>• <i>fail</i></li> </ul> <p>In Struts,</p> <ul style="list-style-type: none"> <li>• try to execute submit method in the action, <i>fail</i></li> <li>• <i>fail</i></li> </ul>
<b>default method</b>	Calling the "default" method via "doDefault" is not supported.

## Detail

- [Release Notes \(to date\)](#)

Pending and open issues

type	key	summary	status	resolution
<div><div> Unable to locate Jira server for this macro. It may be due to Application Link configuration.</div></div>				

Other resources

- [Commit Logs \(Struts 1 and Struts 2\)](#)
- [Source Code Repository \(includes change browsing\)](#)