

StrutsClassicRelease130Notes

THIS PAGE PROVIDED AS A REFERENCE ONLY. DO NOT EDIT THIS PAGE. EDIT THE COPY IN SVN UNDER CORE/USERGUIDE, AND THEN COPY THE OUTPUT HERE.

RELEASE NOTES FOR STRUTS CLASSIC 1.3.0

Introduction

This section contains release notes for changes that have taken place to Struts Classic since Version 1.2.7. To keep up-to-date on all changes to Struts, subscribe to the (commits at struts.apache.org) list. To preview our plans for upcoming changes, please visit the Roadmap page.

Notes on upgrading are maintained in the Wiki Upgrade pages. The wiki is a community maintained resource - please feel free to add your input so that everyone can benefit from the collective experience.

For the version requirements of each library, see the Installation chapter. Highlights of Changes

The purpose of this section is to highlight the new features since the Version 1.2.7 release. For more detail, see the Project Info reports for each subproject, which include a complete changelog and list of external dependencies.

You can also access to the Struts source repository and change logs directly through both web browser and Subversion client interfaces. People

On the volunteer front, two new committers, Wendy Smoak and Gary [VanMatre](#), accepted invitations to join us. Struts Classic

Since Struts 1.2.7, we have subdivided Struts into several subprojects, each with its own release cycle. The codebases that were part of Struts 1.2.7 will inherit version 1.3.0, but, hereafter, subproject revisions will increment independently of each other.

Since many teams will use more than one Struts subproject, we are bundling several of the subprojects in a single distribution called "Struts Classic". The seven subprojects included in the Struts Classic 1.3.0 distribution are: Applications, Core, EL, Plugins, Site, Taglibs, and Tiles.

The Struts website and Struts Classic distribution are being built with Maven. But, of course, you can continue to build your own projects any way you like!) Struts Applications

The applications have been moved into their own subproject, the builds have been "Mavenized", but the set of example applications is essentially unchanged.

- In [MailReader](#), fix invalid password limit error.

Struts Core

Major changes to Struts Core include

- Composable Request Processor
- Arbitrary configuration properties
- Enhanced Global Exception Handlers
- Extends attribute
- "isCommitted" Exception Handling
- Postback Actions
- Wildcard [ActionConfig](#) properties

Composable Request Processor

In prior versions, the request processing gauntlet is represented as a series of methods. While it was easy to override the methods to provide different functionality, it was not easy to use multiple extension that each wanted to override the request processor in a different way.

For Struts 1.3, we turned the request processor methods into Commands that are part of a flexible Chain of Commands. Rather than subclassing a monolithic object, we can now just replace Commands. Commands can also be inserted or removed, if needed, to extend or streamline the request processing gauntlet to meets the needs of different kinds of applications.

We fully expect the Struts 1.3 request processor to be backwardly compatible with earlier versions. But, to be prudent, we wanted to release 1.3 with the Composable Request Processor before making additional changes.

We've made a few other changes and improvements in Struts 1.3.x, but the two big-ticket items are subprojects and the new request processor.

Arbitrary configuration properties

Most every Struts configuration element now accepts a map of key/value pairs. Arbitrary configuration properties let us externalize more of our configurations, encouraging reuse.

Arbitrary Configuration should make it easier for everyone to write reusable chunks of code, for [ActionMappings](#), Exception Handlers, and other members. Enhanced Global Exception Handlers

We now support use of global exception handlers in cases when no [ActionConfig](#) has yet been identified, as in with "preprocessing" commands in a custom request processing chain. In the case when there is no [ActionConfig](#), the [AbstractExceptionHandler](#) will now call a new method in [ModuleConfig](#) which provides similar logic for finding a globally mapped exception handler for the given exception class or any of its superclasses. Extends attribute

In the Struts configuration file, we can now use the extends attribute to adopt default settings from another element, as we already can in the Tiles configuration files.

Extends makes using XML elements much more like object-orientated programming. You can setup a base element, and then only specify the behavior that changes. Extends lets us set default values and then "program by difference". When elements are not relentlessly chanting the same things over and over, it's easier to see the forest for the trees. "isCommitted" Exception Handling

When using Tiles, sometimes the response is already committed when an exception occurs. Now, the [ExceptionHandler](#) tests response.isCommitted, and, if true, includes the configured view path, rather than forwarding to it. The updated [ExceptionHandler](#) accepts new configuration attributes which let you choose alternate behavior, or no behavior at all. See the [JavaDoc](#) for details. Postback Forms

The action attribute of the Form taglib is now optional. If omitted, the original URI of the original request is used.

Wildcard [ActionConfig](#) properties

Wildcards can now be used in the properties of an [ActionConfig](#). This makes it easier to pass multiple request-time values to the Action without complicating the "parameter" attribute. Struts EL

Aside from addressing deprecations, there was one minor change to Struts EL.

- Remove reference to html:html's locale attr in TestELHtmlTag

Struts Plugins

This subproject was extracted from the Plugins package within Struts Core. The code itself is unchanged. Struts Site

The new Site subproject hosts the top-layer of the Struts website, and serves as a portal to the other subprojects. Struts Taglibs

We're changed the way we generate our reference documentation for the taglibs, but we think you will find the new reference just as useful as the old. Otherwise, we made only minor fixes to the trusty Struts Taglibs.

- Update error message for when value is null to specify the id of the bean being defined.
- Updated tag to filter value field to prevent it from breaking HTML if the value has reserved characters in it.
- Changed logic so that the "must specify only one of..." message only get's thrown if the user is specifying more than one of the available options. The code was throwing the message if the user didn't specify any at all and should be flowing down to the "no value specified" error message. Update "can contain only one of..." message to specify which bean has the error.
- Update [FormTag](#) to provide better debug message in the scenario where an action mapping is missing a form bean name.
- Update [TagUtils](#) to provide a more specific error message in the case where properties on a formbean are not found. Basically this change handles the scenario where the Constants.BEAN_KEY value was being displayed as the bean name. Now, the actual bean name will be displayed.
- Changing rendering of the form name to use the 'id' attribute when in XHTML strict mode. If an 'id' is already identified with the 'tyleld' attribute, an exception is thrown.
- Changed dynamic focus javascript to choose form using 'id' attribute in XHTML mode
- Changed the "action" attribute of the Form taglib to be optional. If omitted, the original URI of the request is used.
- Filter anchors out when determining action mapping name.
- Fix [MessagesTag](#) so that non-resource messages are possible for multiple messages, instead of just the first one.
- Added escape for single quotes in javascript values

Struts Tiles

Tiles is being refactored into a standalone package that can be used with or without Struts. For more on how this affects Tiles users, see the Tiles subproject site.

Next: Installation