Reference Documentation: beehive-netui-config.xml File

Table of contents

1. Description

The `beehive-netui-config.xml` file configures the runtime behavior of your NetUI web application. It lives under the WEB-INF directory.

Use `beehive-netui-config.xml` to declare interceptor classes, override the default handler classes, etc.

> **Note:**
> Currently, when you want to add your own settings, you cannot simply create the `beehive-netui-config.xml` file in WEB-INF. You need to create a file that includes the default settings. See Default Settings.

2. Default Settings

To start with a file that includes the default settings, use this version.

3. General Structure

```
<netui-config>
  <expression-languages>
    <default-language>
    <expression-language>
      <name>
      <factory-class>
      <binding-contexts>
        <binding-context>
          <name>
          <factory-class>
  <pageflow-action-interceptors>
    <global>
      <simple-action-interceptor>
        <intercept-path>
        <after-action>
      <action-interceptor>
        <interceptor-class>
        <custom-property>
          <name>
          <value>
    <per-pageflow>
      <pageflow-uri>
      <simple-action-interceptor>
        <intercept-path>
        <after-action>
      <action-interceptor>
        <interceptor-class>
        <custom-property>
      <per-action>
        <action-name>
```

```
        <simple-action-interceptor>
          <intercept-path>
          <after-action>
        <action-interceptor>
          <interceptor-class>
          <custom-property>
<pageflow-handlers>
   <action-forward-handler>
      <handler-class>
      <custom-property>
   <exceptions-handler>
      <handler-class>
      <custom-property>
   <forward-redirect-handler>
      <handler-class>
      <custom-property>
   <login-handler>
      <handler-class>
      <custom-property>
   <reloadable-class-handler>
      <handler-class>
      <custom-property>
<pageflow-config>
   <enable-self-nesting>
   <max-forwards-per-request>
   <max-nesting-stack-depth>
   <ensure-secure-forwards>
   <throw-session-expired-exception>
   <multipart-handler>
   <prevent-cache>
   <module-config-locators>
      <module-config-locator>
        <description>
        <locator-class>
<pageflow-factories>
   <flowcontroller-factory>
      <factory-class>
      <custom-property>
   <faces-backing-bean-factory>
      <factory-class>
      <custom-property>
<default-shared-flow-refs>
   <shared-flow-ref>
      <name>
      <type>
<type-converters>
   <type-converter>
      <type>
      <converter-class>
<jsp-tag-config>
   <doctype>
   <id-javascript>
   <tree-image-location>
   <tree-renderer-class>
```

```
<url-config>
   <url-encode-urls>
   <html-amp-entity>
   <templated-url-formatter-class>
<iterator-factories>
   <iterator-factory>
      <name>
      <factory-class>
<request-interceptors>
   <global>
      <request-interceptor>
         <interceptor-class>
         <custom-property>
<prefix-handlers>
   <prefix-handler>
      <name>
      <handler-class>
```

4. Elements

4.1. <action-forward-handler>

Syntax

```
<action-forward-handler>
  <handler-class> xsd:string </handler-class> [occurrences: 1]
  <custom-property>
    <name> xsd:string </name> [occurrences: 1]
    <value> xsd:string </value> [occurrences: 1]
  </custom-property> [occurrences: 0-*]
</action-forward-handler>
```

Parents:

<pageflow-handlers>

Children:

<handler-class>, <custom-property>

4.2. <action-interceptor>

Syntax

```
<action-interceptor>
  <interceptor-class> xsd:string </interceptor-class> [occurrences: 1]
  <custom-property>
    <name> xsd:string </name> [occurrences: 1]
    <value> xsd:string </value> [occurrences: 1]
  </custom-property> [occurrences: 0-*]
</action-interceptor>
```

Parents:

<global>, <per-pageflow>, <per-action>

Children:

<interceptor-class>, <custom-property>

4.3. <action-name>

Syntax

```
<action-name> xsd:string <action-name> [occurrences: 1]
```

Parents:

<per-action>

Children:

4.4. <after-action>

Syntax

```
 <after-action> xsd:boolean <after-action> [occurrences: 0-1]
```

Parents:

<simple-action-interceptor>

Children:

4.5. <binding-context>

Syntax

```
<binding-context> [occurrences: 0-*]
  <name> xsd:string </name> [occurrences: 1]
  <factory-class> xsd:string </factory-class> [occurrences: 1]
</binding-context>
```

Parents:

<binding-contexts>

Children:

<name>, <factory-class>

4.6. <binding-contexts>

Syntax

```
</binding-contexts> [occurrences: 0-1]
  <binding-context> [occurrences: 0-*]
    <name> xsd:string </name> [occurrences: 1]
    <factory-class> xsd:string </factory-class> [occurrences: 1]
  </binding-context>
</binding-contexts>
```

Parents:

<expression-language>

Children:

<binding-context>

4.7. <converter-class>

Syntax

```
<converter-class> xsd:string </converter-class> [occurrences: 1]
```

Parents:

<type-converter>

Children:

4.8. <custom-property>

Syntax

```
<custom-property>
  <name> xsd:string </name> [occurrences: 1]
  <value> xsd:string </value> [occurrences: 1]
</custom-property>
```

Parents:

<action-interceptor>, <request-interceptor>, <action-forward-handler>,
<exceptions-handler>, <forward-redirect-handler>, <login-handler>,
<reloadable-class-handler> <flowcontroller-factory> <faces-backing-bean-factory>

Children:

<name>, <value>

4.9. <default-language>

Syntax

```
<default-language> xsd:string </default-language> [occurrences: 1]
```

Parents:

<expression-language>

Children:

4.10. <default-shared-flow-refs>

Allows JSPs to access shared flows even when there is no page flow that declares the shared flow. This configuration element declares default shared flow references that can be used across the web app.

Syntax

```
<default-shared-flow-refs>
  <shared-flow-ref>
    <name> xsd:string </name> [occurrences: 1]
    <type> xsd:string </type> [occurrences: 1]
  </shared-flow-ref> [occurrences: 0-*]
</default-shared-flow-refs>
```

Parents:

<netui-config>

Children:

<shared-flow-ref>

4.11. <description>

Syntax

```
<description> xsd:string </description> [occurrences: 0-1]
```

Parents:

<module-config-locator>

Children:

4.12. <doctype>

Sets the WebApp default HTML format. The currently supported formats are HTML 4.01 Transitional DTD (Loose), HTML 4.01 Transitional DTD with Quirks mode on, and XHTML 1.0 Transitional. The default value is *html4-loose-quirks*.

Syntax

```
<doctype> [ html4-loose | html4-loose-quirks | xhtml1-transitional ]
</doctype> [occurrences: 0-1]
```

Parents:

<jsp-tag-config>

Children:

4.13. <enable-self-nesting>

When this value is set to `false` (the default), a nested page flow that is *anywhere* on the nesting stack will be *reactivated* if one of its actions is hit, and all page flows higher up the stack will be destroyed. When this value is set to `true`, a *new* instance of the nested page flow will end up at the top of the stack. As an example, consider the following stack of page flows:

```
A, B, C (top)
```

When an action in B is hit, one of two things will happen:

*   If `enable-self-nesting` is set to `false`, then B will be reactivated and C will be destroyed. The nesting stack will look like this:
    ```
    A, B (top)
    ```
*   If `enable-self-nesting` is set to `true`, then a new instance of B will be at the top of the stack, which will look like this:
    ```
    A, B, C, B (top)
    ```

Note that setting this value to `true` is **not back-button-friendly**. Hitting a Page Flow action after using the browser back button can often end up executing an action on a nested page flow in the middle of the nesting stack.

Syntax

Page 10

```
<enable-self-nesting> xsd:boolean </enable-self-nesting> [occurrences: 0-1]
```

Parents:

<pageflow-config>

Children:

4.14. <ensure-secure-forwards>

For some server implementations, security checks are executed only when an URL is called directly, while security checks for server forwards are ignored.

When this element is set to `true`, server forwards are guaranteed to be checked.

Syntax

```
<ensure-secure-forwards> xsd:boolean </ensure-secure-forwards>
[occurrences: 0-1]
```

Parents:

<pageflow-config>

Children:

4.15. <exceptions-handler>

Typically, exceptions thrown by the controller class are delivered in a wrapper exception class. The class named here, can be used to unwrap these exceptions and handle the underlying exception in a more fine-grained manner.

Syntax

```
<exceptions-handler>
  <handler-class> xsd:string </handler-class> [occurrences: 1]
  <custom-property>
    <name> xsd:string </name> [occurrences: 1]
    <value> xsd:string </value> [occurrences: 1]
  </custom-property> [occurrences: 0-*]
</exceptions-handler>
```

Parents:

<pageflow-handlers>

Children:

<handler-class>, <custom-property>

4.16. <expression-language>

Syntax

```
  <expression-language> [occurrences: 1-*]
    <name> xsd:string </name> [occurrences: 1]
    <factory-class> xsd:string </factory-class> [occurrences: 1]
    <binding-contexts> binding-contexts </binding-contexts> [occurrences:
0-1]
  </expression-language>
```

Parents:

<pageflow-handlers>

Children:

<name>, <factory-class>, <binding-contexts>

4.17. <expression-languages>

Syntax

```
<expression-languages> [occurrences: 1]
  <default-language> xsd:string </default-language> [occurrences: 1]
  <expression-language> [occurrences: 1-*]
    <name> xsd:string </name> [occurrences: 1]
    <factory-class> xsd:string </factory-class> [occurrences: 1]
    <binding-contexts> binding-contexts </binding-contexts> [occurrences:
0-1]
  </expression-language>
</expression-languages>
```

Parents:

<netui-config>

Children:

<default-language>, <expression-language>

4.18. <faces-backing-bean-factory>

This setting configures a factory for creating "backing beans" for JavaServer Faces pages.
The referenced class must extend
org.apache.beehive.netui.pageflow.FacesBackingBeanFactory

Syntax

```
<faces-backing-bean-factory> [occurrences: 0-*]
    <factory-class> xsd:string </factory-class> [occurrences:
1]

    <custom-property> [occurrences: 0-*]
        <name> xsd:string </name> [occurrences: 1]
        <value> xsd:string </value> [occurrences: 1]
    </custom-property>
</faces-backing-bean-factory>
```

Parents:

<pageflow-factories>

Children:

<factory-class>, <custom-property>

4.19. <factory-class>

This setting configures a factory class and its custom properties.

Syntax

```
<factory-class> xsd:string </factory-class> [occurrences: 1]
```

Parents:

<binding-contexts>, <expression-language>, <iterator-factory> <flowcontroller-factory>, <faces-backing-bean-factory>

Children:

None.

4.20. <flowcontroller-factory>

This setting configures a factory for creating page flows and shared flows. The referenced class must extend org.apache.beehive.netui.pageflow.FlowControllerFactory

Syntax

```
<flowcontroller-factory> [occurrences: 0-*]
    <factory-class> xsd:string </factory-class> [occurrences:
1]

    <custom-property> [occurrences: 0-*]
        <name> xsd:string </name> [occurrences: 1]
```

```
              <value> xsd:string </value> [occurrences: 1]
            </custom-property>
          </flowcontroller-factory>
```

Parents:

<pageflow-factories>

Children:

<factory-class>, <custom-property>

4.21. <forward-redirect-handler>

Syntax

```
<forward-redirect-handler>
  <handler-class> xsd:string </handler-class> [occurrences: 1]
  <custom-property>
    <name> xsd:string </name> [occurrences: 1]
    <value> xsd:string </value> [occurrences: 1]
  </custom-property> [occurrences: 0-*]
</forward-redirect-handler>
```

Parents:

<pageflow-handlers>

Children:

Children:

<handler-class>, <custom-property>

4.22. <global>

Syntax

```
<global> [occurrences: 0-1]
  <simple-action-interceptor>
    <intercept-path> xsd:string </intercept-path> [occurrences: 1]
    <after-action> xsd:boolean </after-action> [occurrences: 0-1]
  </simple-action-interceptor> [occurrences: 0-*]
  <action-interceptor>
    <interceptor-class> xsd:string </interceptor-class> [occurrences: 1]
    <custom-property>
      <name> xsd:string </name> [occurrences: 1]
      <value> xsd:string </value> [occurrences: 1]
    </custom-property> [occurrences: 0-*]
  </action-interceptor> [occurrences: 0-*]
</global>
```

```
<global>
  <request-interceptor>
    <interceptor-class> xsd:string </interceptor-class> [occurrences: 1]
    <custom-property>
      <name> xsd:string </name> [occurrences: 1]
      <value> xsd:string </value> [occurrences: 1]
    </custom-property> [occurrences: 0-*]
  </request-interceptor> [occurrences: 0-*]
</global>
```

Parents:

<pageflow-action-interceptors>, <request-interceptors>

Children:

<simple-action-interceptor>, <action-interceptor>, <request-interceptor>

4.23. <handler-class>

Syntax

```
<handler-class> xsd:string </handler-class> [occurrences: 1]
```

Parents:

<action-forward-handler>, <exceptions-handler>, <forward-redirect-handler>, <login-handler>, <reloadable-class-handler>, <prefix-handler>

Children:

4.24. <html-amp-entity>

The url parameter separators of & (single amphersand) are encoded to & (ampersand followed by amp;) by default according to the html 401 specifications set forth by W3C. This item can be set to false and the parameter separators will not be encoded.

Syntax

Syntax

```
<html-amp-entity> xsd:boolean </html-amp-entity> [occurrences: 0-1]
```

Parents:

<url-config>

Children:

4.25. <id-javascript>

This configuration setting will change the type of support for JavaScript. The default is to output only the Beehive NetUI JavaScript support. There are two legacy mode options that output pre-Beehive JavaScript support. When *id-javaScript* is set to **legacy** the pre-Beehive lookup methods are written out along with the current Beehive lookup methods. When the value is set to **legacyOnly** only the pre-beehive lookup methods are written out. This is a pure backward compatibility mode for pre-beehive NetUI behavior.

Syntax
```
<id-javascript> [ default | legacy | legacyOnly ] </id-javascript>
[occurrences: 0-1]
```
Parents:

<jsp-tag-config>

Children:

4.26. <interceptor-class>

Syntax
```
<interceptor-class> xsd:string <interceptor-class> [occurrences: 1]
```
Parents:

<action-interceptor>, <request-interceptor>

Children:

4.27. <intercept-path>

Syntax
```
<intercept-path> xsd:string </intercept-path> [occurrences: 1]
```
Parents:

<simple-action-interceptor>

Children:

4.28. <iterator-factories>

Syntax

```
<iterator-factories>
  <iterator-factory> [occurrences: 0-*]
    ...
  </iterator-factory>
</iterator-factories>
```

Parents:

<netui-config>

Children:

<iterator-factory>

4.29. <iterator-factory>

Syntax

```
<iterator-factory> [occurrences: 0-*]
  <name> xsd:string </name> [occurrences: 1]
  <factory-class> xsd:string </factory-class> [occurrences: 1]
</iterator-factory>
```

Parents:

<iterator-factories>

Children:

<name>, <factory-class>

4.30. <jsp-tag-config>

Syntax

```
<jsp-tag-config>
  <doctype> xsd:string </doctype> [occurrences: 0-1]
  <id-javascript> [ default | legacy | legacyOnly ] </id-javascript>
[occurrences: 0-1]
  <tree-image-location> xsd:string </tree-image-location> [occurrences:
0-1]
</jsp-tag-config>
```

Parents:

<netui-config>

Children:

<doctype>, <id-javascript>, <tree-image-location>,

4.31. <locator-class>

Syntax

```
<locator-class> xsd:string </locator-class> [occurrences: 1]
```

Parents:

<module-config-locator>

Children:

4.32. <login-handler>

Syntax

```
<login-handler>
  <handler-class> xsd:string </handler-class> [occurrences: 1]
  <custom-property>
    <name> xsd:string </name> [occurrences: 1]
    <value> xsd:string </value> [occurrences: 1]
  </custom-property> [occurrences: 0-*]
</login-handler>
```

Parents:

<pageflow-handlers>

Children:

<handler-class>, <custom-property>

4.33. <max-forwards-per-request>

If the number of server forwards exceeds the given count, an error is written to the response and no further forwarding is excuted. This is mainly to prevent infinite loops of server forwards. To reproduce the error, invoke this action in a page flow:

```
    @Jpf.Action(
        forwards={
            @Jpf.Forward(name="self", path="overflow.do")
```

```
        }
    )
    public Forward overflow()
    {
        return new Forward( "self" );
    }
}
```

If this element is omitted, the error will be written to the response after 25 server forwards within a single request.

When an application is in development, a descriptive error is sent; for deployed applications, a 500 response is sent.

Syntax

```
<max-forwards-per-request> xsd:int </max-forwards-per-request>
[occurrences: 0-1]
```

Parents:

<pageflow-config>

Children:

4.34. <max-nesting-stack-depth>

This parameter sets the maximum size of the Page Flow nesting stack. If page flows are repeatedly nested until the stack size exceeds the specified value, an error is written to the response object and any further nesting is not allowed. This helps prevent the nesting stack from consuming large amounts of resources.

When an application is in development, a descriptive error is sent; for deployed applications, a 500 response is sent.

Syntax

```
<max-nesting-stack-depth> xsd:int </max-nesting-stack-depth> [occurrences:
0-1]
```

Parents:

<pageflow-config>

Children:

4.35. <module-config-locator>

Syntax

```
<module-config-locator> [occurrences: 0-*]
  <description> xsd:string </description> [occurrences: 0-1]
  <locator-class> xsd:string </locator-class> [occurrences: 1]
</module-config-locator>
```

Parents:

<module-config-locators>

Children:

<description>, <locator-class>

4.36. <module-config-locators>

Names a class or group of classes that know the location of the web application's Struts module configuration files. By default these configuration files are saved to `WEB-INF/classes/_pageflow/`. In cases where this default location has been overridden or augmented by a custom compilation process, use `<module-config-locators>` to point to a Java class that knows the new location.

Syntax

```
<module-config-locators> [occurrences: 0-1]
  <module-config-locator> [occurrences: 0-*]
    <description> xsd:string </description> [occurrences: 0-1]
    <locator-class> xsd:string </locator-class> [occurrences: 1]
  </module-config-locator>
</module-config-locators>
```

Parents:

<pageflow-config>

Children:

<module-config-locator>

4.37. <multipart-handler>

By default multi-part file uploading is disabled for NetUI web applications. To turn it on, you must (1) explicitly set this element to `memory` or `disk`, or (2) set the `multipartHandler` attribute on a page flow's `@Jpf.Controller` annotation. Doing (1) will enable file uploading across the entire web applcation; doing (2) enables file uploading on a controller-by-controller basis.

Also see:

[Attribute multipartHandler on annotation interface](#)
[org.apache.beehive.netui.pageflow.annotations.Jpf](#)

Syntax

```
<multipart-handler> [ disabled | memory | disk ] </multipart-handler>
[occurrences: 0-1]
```

Parents:

[<pageflow-config>](#)

Children:

4.38. <name>

Syntax

```
<name> xsd:string </name> [occurrences: 1]
```

Parents:

[<expression-language>](#), [<iterator-factory>](#), [<binding-context>](#), [<custom-property>](#),
[<shared-flow-ref>](#), [<prefix-handler>](#)

Children:

4.39. <netui-config>

The top-level element in the `beehive-netui-config.xml` file.

Syntax

```
<netui-config>
  <expression-languages> ... </expression-languages> [occurrences: 1]
  <pageflow-action-interceptors> ... </pageflow-action-interceptors>
[occurrences: 0-1]
  <pageflow-handlers> ... </pageflow-handlers> [occurrences: 0-1]
  <pageflow-factories> ... </pageflow-factories> [occurrences: 0-1]
  <pageflow-config> ... </pageflow-config> [occurrences: 0-1]
  <default-shared-flow-refs> ... </default-shared-flow-refs> [occurrences:
0-1]
  <type-converters> ... </type-converters> [occurrences: 0-1]
  <jsp-tag-config> ... </jsp-tag-config> [occurrences: 0-1]
  <url-config> ... </url-config> [occurrences: 0-1]
  <iterator-factories> ... </iterator-factories> [occurrences: 0-1]
  <request-interceptors> ... </request-interceptors> [occurrences: 0-1]
```

```
  <prefix-handlers> ... </prefix-handlers> [occurrences: 0-1]
</netui-config>
```

Parents:

Children:

<expression-languages>, <pageflow-action-interceptors>, <pageflow-handlers>,
<pageflow-factories>, <pageflow-config>, <default-shared-flow-refs>, <type-converters>,
<jsp-tag-config>, <url-config>, <iterator-factories>, <request-interceptors>,
<prefix-handlers>

4.40. <pageflow-action-interceptors>

To run code before or after Page Flow actions, you configure a
<pageflow-action-interceptor> group. A simple example of this would be a monitoring
infrastructure that keeps track of the number of actions raised. To keep track of the number
of actions raised, register an interceptor that runs the counting code before going to any
action. A more complex example is an interceptor that "injects" a page flow before allowing
you to go to the current one; for instance, it might decide (on the fly) to take you to a nested
page flow that asks you to fill out a "satisfaction survey" before sending you to the
destination page flow.

The action-intercepting class is run twice: both before *and* after the execution of the action.
When run before an action, it can do three things:
- change the destination URI and thus prevent the action from running, or,
- set the destination URI to `null` (no forwarding) and thus prevent the action from
  running, or,
- "inject" an entire nested page flow to run before the action is invoked. If the override
  forward URI is a nested page flow, then it will run until it raises one of its return actions.
  At that point, `afterNestedIntercept` is called on *this interceptor*, which can again
  choose to override the forward or allow the original action to run.

When run after an action, it can do two things:
- change the destination URI that was returned by the action, or,
- set the destination URI to `null` (no forwarding).

Also see:

Interface org.apache.beehive.netui.pageflow.interceptor.ActionInterceptor

Syntax

```
<pageflow-action-interceptors>
```

```
        <global> [occurrences: 0-1]
                <simple-action-interceptor> ...
</simple-action-interceptor> [occurrences: 0-*]
                <action-interceptor> ... </action-interceptor>
[occurrences: 0-*]
        </global>
        <per-pageflow> [occurrences: 0-*]
                <pageflow-uri> xsd:string </pageflow-uri> [occurrences: 1]
                <simple-action-interceptor> ...
</simple-action-interceptor> [occurrences: 0-*]
                <action-interceptor> ... </action-interceptor>
[occurrences: 0-*]
                <per-action> [occurrences: 0-*]
                        <action-name> xsd:string </action-name>
[occurrences: 1]
                        <simple-action-interceptor> ...
</simple-action-interceptor> [occurrences: 0-*]
                        <action-interceptor> ... </action-interceptor>
[occurrences: 0-*]
                </per-action>
        </per-pageflow>
</pageflow-action-interceptors>
```

Parents

<netui-config>

Children

<global>, <per-pageflow>

4.41. <pageflow-config>

Syntax

```
<pageflow-config>
        <enable-self-nesting> xsd:boolean </enable-self-nesting>
[occurrences: 0-1]
        <max-forwards-per-request> xsd:int </max-forwards-per-request>
[occurrences: 0-1]
        <max-nesting-stack-depth> xsd:int </max-nesting-stack-depth>
[occurrences: 0-1]
        <ensure-secure-forwards> xsd:boolean </ensure-secure-forwards>
[occurrences: 0-1]
        <throw-session-expired-exception> xsd:boolean
</throw-session-expired-exception> [occurrences: 0-1]
        <multipart-handler> [ disabled | memory | disk ]
</multipart-handler> [occurrences: 0-1]
        <prevent-cache> [ default | always | inDevMode ] </prevent-cache>
[occurrences: 0-1]
        <module-config-locators> ... </module-config-locators>
[occurrences: 0-1]
</pageflow-config>
```

Parents:

<netui-config>

Children:

<enable-self-nesting>, <max-forwards-per-request>, <max-nesting-stack-depth>, <ensure-secure-forwards>, <throw-session-expired-exception>, <multipart-handler>, <prevent-cache>, <module-config-locators>

Configures the NetUI runtime across the webapp. If you want to disable file-upload, for instance, you set the multipart-handler to "disabled".

4.42. <pageflow-handlers>

These settings let you override base framework behavior. A good example is the LoginHandler. By default, standard Servlet APIs are used to determine whether a user is logged in, and server-specific APIs are used to actually log in a user when `login()` is called inside a page flow. If you want to replace this behavior with your own login scheme (which may look at a User database table for login information), you can provide your own LoginHandler that defines methods like `login()` and `isUserInRole()`.

Syntax

```
<pageflow-handlers>
        <action-forward-handler> [occurrences: 0-*]
                <handler-class> xsd:string </handler-class> [occurrences:
1]
                <custom-property> [occurrences: 0-*]
           <name> xsd:string </name> [occurrences: 1]
           <value> xsd:string </value> [occurrences: 1]
        </custom-property>
        </action-forward-handler>
        <exceptions-handler> [occurrences: 0-*]
                <handler-class> xsd:string </handler-class> [occurrences:
1]
                <custom-property> [occurrences: 0-*]
           <name> xsd:string </name> [occurrences: 1]
           <value> xsd:string </value> [occurrences: 1]
        </custom-property>
        </exceptions-handler>
        <forward-redirect-handler> [occurrences: 0-*]
                <handler-class> xsd:string </handler-class> [occurrences:
1]
                <custom-property> [occurrences: 0-*]
           <name> xsd:string </name> [occurrences: 1]
           <value> xsd:string </value> [occurrences: 1]
        </custom-property>
        </forward-redirect-handler>
```

```
        <login-handler> [occurrences: 0-*]
                <handler-class> xsd:string </handler-class> [occurrences:
1]
                <custom-property> [occurrences: 0-*]
            <name> xsd:string </name> [occurrences: 1]
            <value> xsd:string </value> [occurrences: 1]
        </custom-property>
        </login-handler>
        <reloadable-class-handler> [occurrences: 0-*]
                <handler-class> xsd:string </handler-class> [occurrences:
1]
                <custom-property> [occurrences: 0-*]
            <name> xsd:string </name> [occurrences: 1]
            <value> xsd:string </value> [occurrences: 1]
        </custom-property>
        </reloadable-class-handler>
</pageflow-handlers>
```

Parents:

<u><netui-config></u>

Children:

<u><action-forward-handler></u>, <u><exceptions-handler></u>, <u><forward-redirect-handler></u>, <u><login-handler></u>, <u><reloadable-class-handler></u>

4.43. <pageflow-factories>

These settings let you override the default factories for creating page flows, shared flows, and JavaServer Faces backing beans.

Syntax

```
            <pageflow-factories>
                <flowcontroller-factory> [occurrences: 0-*]
                    <factory-class> xsd:string </factory-class>
[occurrences: 1]
                    <custom-property> [occurrences: 0-*]
                        <name> xsd:string </name> [occurrences: 1]
                        <value> xsd:string </value> [occurrences: 1]
                    </custom-property>
                </flowcontroller-factory>
                <faces-backing-bean-factory> [occurrences: 0-*]
                    <factory-class> xsd:string </factory-class>
[occurrences: 1]
                    <custom-property> [occurrences: 0-*]
                        <name> xsd:string </name> [occurrences: 1]
                        <value> xsd:string </value> [occurrences: 1]
                    </custom-property>
                </faces-backing-bean-factory>
```

```
        </pageflow-factories>
```

Parents:

<netui-config>

Children:

<flowcontroller-factory>, <faces-backing-bean-factory>

4.44. <pageflow-uri>

Syntax

```
<pageflow-uri> xsd:string </pageflow-uri> [occurrences: 1]
```

Parents:

<per-pageflow>

Children:

4.45. <per-action>

Syntax

```
<per-action> [occurrences: 0-*]
        <action-name> xsd:string </action-name> [occurrences: 1]
        <simple-action-interceptor> netui:simple-action-interceptor
</simple-action-interceptor> [occurrences: 0-*]
        <action-interceptor> netui:action-interceptor </action-interceptor>
[occurrences: 0-*]
</per-action>
```

Parents:

<global>

Children:

<action-name>, <simple-action-interceptor>, <action-interceptor>

4.46. <per-pageflow>

Syntax

```
<per-pageflow> [occurrences: 0-*]
        <pageflow-uri> xsd:string </pageflow-uri> [occurrences: 1]
        <simple-action-interceptor> netui:simple-action-interceptor
```

```
</simple-action-interceptor> [occurrences: 0-*]
        <action-interceptor> netui:action-interceptor </action-interceptor>
[occurrences: 0-*]
        <per-action> [occurrences: 0-*]
                <action-name> xsd:string </action-name> [occurrences: 1]
                <simple-action-interceptor> netui:simple-action-interceptor
</simple-action-interceptor> [occurrences: 0-*]
                <action-interceptor> netui:action-interceptor
</action-interceptor> [occurrences: 0-*]
        </per-action>
</per-pageflow>
```

Parents:

<global>

Children:

<pageflow-uri>, <simple-action-interceptor>, <action-interceptor>, <per-action>

4.47. <prefix-handler>

Syntax

```
<prefix-handler>
  <name> xsd:string </name> [occurrences: 1]
  <handler-class> xsd:string </handler-class> [occurrences: 1]
</prefix-handler>
```

Parents:

<prefix-handlers>

Children:

<name>, <handler-class>

4.48. <prefix-handlers>

Syntax

```
<prefix-handlers>
  <prefix-handler> [occurrences: 0-*]
    <name> xsd:string </name> [occurrences: 1]
    <handler-class> xsd:string </handler-class> [occurrences: 1]
  </prefix-handler>
</prefix-handlers>
```

Parents:

<netui-config>

Children:

Page 27

<prefix-handler>

4.49. <prevent-cache>

This setting configures browser caching behavior for pages returned from Page Flow actions. Setting it to `always` causes no-cache headers to be added to the HTTP response, specifying that pages are **never** cached. Setting it to `inDevMode` means that pages may be cached only when the server is in production mode.

Syntax

```
<prevent-cache> [ default | always | inDevMode ] </prevent-cache>
[occurrences: 0-1]
```

Parents:

<pageflow-config>

Children:

4.50. <reloadable-class-handler>

Syntax

```
<reloadable-class-handler>
  <handler-class> xsd:string </handler-class> [occurrences: 1]
  <custom-property>
    <name> xsd:string </name> [occurrences: 1]
    <value> xsd:string </value> [occurrences: 1]
  </custom-property> [occurrences: 0-*]
</reloadable-class-handler>
```

Parents:

<pageflow-handlers>

Children:

<handler-class>, <custom-property>

4.51. <request-interceptor>

Syntax

```
<request-interceptor>
  <interceptor-class> xsd:string </interceptor-class> [occurrences: 1]
  <custom-property>
    <name> xsd:string </name> [occurrences: 1]
```

```
    <value> xsd:string </value> [occurrences: 1]
  </custom-property> [occurrences: 0-*]
</request-interceptor>
```

Parents:

<global>

Children:

<interceptor-class>, <custom-property>

4.52. <request-interceptors>

Syntax

```
<request-interceptors>
  <global> [occurrences: 0-1]
    <request-interceptor> [occurrences: 0-*]
      <interceptor-class> xsd:string </interceptor-class> [occurrences: 1]
      <custom-property> ... </custom-property> [occurrences: 0-*]
    </request-interceptor>
  </global>
</request-interceptors>
```

Parents:

<netui-config>

Children:

<global>

4.53. <shared-flow-ref>

Syntax

```
<shared-flow-ref>
  <name> xsd:string </name> [occurrences: 1]
  <type> xsd:string </type> [occurrences: 1]
</shared-flow-ref>
```

Parents:

<default-shared-flow-refs>

Children:

<name>, <type>

4.54. <simple-action-interceptor>

Syntax

```
<simple-action-interceptor>
  <intercept-path> xsd:string </intercept-path> [occurrences: 1]
  <after-action> xsd:boolean </after-action> [occurrences: 0-1]
</simple-action-interceptor>
```

Parents:

<netui-config>

Children:

4.55. <templated-url-formatter-class>

This setting allows you to override the default url formatting class. The default value is
org.apache.beehive.netui.pageflow.internal.DefaultTemplatedUrlFormatter.

Syntax

```
<templated-url-formatter-class> xsd:string </templated-url-fomatter-class>
[occurrences: 0-1]
```

Parents:

<url-config>

Children:

4.56. <throw-session-expired-exception>

When set to `true` (the default), this causes a `SessionExpiredException` to be thrown
in place of some other Page Flow exceptions when the root cause is most likely the
expiration of the user session.

Also see:

Class org.apache.beehive.netui.pageflow.SessionExpiredException

Syntax

```
<throw-session-expired-exception> xsd:boolean
</throw-session-expired-exception> [occurrences: 0-1]
```

Parents:

<pageflow-config>

Children:

4.57. <tree-image-location>

This configuration sets a path for looking up the default images used by the tree tags. The default location of resources changed in Beehive NetUI from pre-Beehive NetUI. This element allows you to set an absolute path to these image resources.

Syntax

```
<tree-image-location> xsd:string </tree-image-location> [occurrences: 0-1]
```

Parents:

<jsp-tag-config>

Children:

4.58. <tree-renderer-class>

Refers to a custom tree renderer class that overrides the default TreeRenderer.

Syntax

```
<tree-renderer-class> xsd:string </tree-renderer-class> [occurrences: 0-1]
```

Parents:

<jsp-tag-config>

Children:

4.59. <type>

Syntax

```
<type> xsd:string </type> [occurrences: 1]
```

Parents:

<type-converter>, <shared-flow-ref>

Children:

4.60. <type-converter>

Syntax

```
<type-converter> [occurrences: 0-*]
  <type> xsd:string </type> [occurrences: 1]
  <converter-class> xsd:string </converter-class> [occurrences: 1]
</type-converter>
```

Parents:

<type-converters>

Children:

<type>, <converter-class>

4.61. <type-converters>

Syntax

```
<type-converters>
  <type-converter> [occurrences: 0-*]
    <type> xsd:string </type> [occurrences: 1]
    <converter-class> xsd:string </converter-class> [occurrences: 1]
  </type-converter>
</type-converters>
```

Parents:

<netui-config>

Children:

<type-converter>

4.62. <url-config>

Syntax

```
<url-config>
  <url-encode-urls> xsd:boolean </url-encode-urls> [occurrences: 0-1]
  <html-amp-entity> xsd:boolean </html-amp-entity> [occurrences: 0-1]
  <templated-url-formatter-class> xsd:string
</templated-url-formatter-class> [occurrences: 0-1]
</url-config>
```

Parents:

[<netui-config>](#)

Children:

[<url-encode-urls>](#), [<html-amp-entity>](#), [<templated-url-formatter-class>](#)

4.63. <url-encode-urls>

This setting allows you to override the default of encoding the urls for tags with href attributes so that already encoded urls will not be re-encoded.

Syntax

```
<url-encode-urls> xsd:boolean </url-encode-urls> [occurrences: 0-1]
```

Parents:

[<url-config>](#)

Children:

4.64. <value>

Syntax

```
<value> xsd:string </value> [occurrences: 1]
```

Parents:

[<custom-property>](#)

Children:

5. Example

Use the following example `netui-config.xsd` file as a guide.

```
<?xml version="1.0" encoding="UTF-8"?>

<netui-config xmlns="http://beehive.apache.org/netui/2004/server/config">

    <expression-languages>
        <default-language>netuiel</default-language>
        <expression-language>
            <name>netuiel</name>
<factory-class>org.apache.beehive.netui.script.el.ExpressionEvaluatorImpl$NetUIELEngine
```

```
            </expression-language>
        </expression-languages>

        <pageflow-action-interceptors>
            <global>
                <action-interceptor>
                    <!-- This interceptor is run before/after ALL page flow
actions. -->
<interceptor-class>example.GlobalInterceptor</interceptor-class>
                    <custom-property>
                        <name>someCustomProperty</name>
                        <value>someValue</value>
                    </custom-property>
                </action-interceptor>
            </global>
            <per-pageflow>
<pageflow-uri>/example/intercepted1/Controller.java</pageflow-uri>
                <!-- This simple interceptor is run before any action in
/example/nested/SomeNestedController.java.
                    It "injects" a nested page flow before allowing the
original action to run. -->
                <simple-action-interceptor>
<intercept-path>/example/nested/SomeNestedController.java</intercept-path>
                </simple-action-interceptor>
            </per-pageflow>
        </pageflow-action-interceptors>

        <pageflow-handlers>
            <login-handler>
                <handler-class>example.CustomLoginHandler</handler-class>
            </login-handler>
        </pageflow-handlers>

        <pageflow-config>
            <multipart-handler>memory</multipart-handler>
        </pageflow-config>

        <jsp-tag-config>
            <doctype>html4-loose</doctype>
            <id-javascript>Legacy-JavaScript-Only</id-javascript>
        </jsp-tag-config>

        <iterator-factories>
        </iterator-factories>

</netui-config>
```

6. Related Topics

Class org.apache.beehive.netui.util.config.ConfigUtil

[Interface org.apache.beehive.netui.pageflow.interceptor.ActionInterceptor](#)