

XSPFormValidator

One of the [Cocoon BuiltInLogicsheets](#).

This [XSP Logicsheet](#) is designed to be used in parallel with the `FormValidatorAction`.

See [FormValidationUsingCocoon](#) for additional background information.

The namespace for this logicsheet is <http://apache.org/xsp/form-validator/2.0>

The following sections describe the purpose of each element.

Descriptor

The `descriptor` element is required if you need to

- query attributes from the form descriptor
- declare that the descriptor file should be re-read if modified

The element has several required attributes

- `name` – the URI to the descriptor file. Using uses the `cocoon context:/ scheme`
- `constraint-set` – the name of the constraint set to read
- `reloadable` – declares that file should be reloaded; default is `true`

Other elements from the tag library should be nested inside this one, if access is required to the descriptor file.

The element can be used multiple times on the page, e.g. referencing different constraint sets.

Results

The `results` element simply returns a Map of all validation results, e.g. for direct manipulation in embedded code.

Validate

A grouping element that provides a named context for other nested elements, e.g. `is-ok`, etc. Should have a `name` attribute that identifies the parameter being used.

Generally, if an element is not used within a `validate` element then it's context can be defined by adding a `name` attribute to the individual element.

Is Elements

There are several tests that can be applied to an element to determine the outcome of validation:

- `is-ok` – successfully validated
- `is-toosmall` – either value or length (strings only) was too small
- `is-error` – general error indicator
- `is-toolarge` – either value or length (strings only) was too large
- `is-nomatch` – parameter failed to match regex
- `is-null` – parameter was null
- `is-notpresent` – parameter was missing

On Elements

Nested markup is included if the parameter was successfully validated. There are `on` elements for each of the scenarios listed above. E.g. `on-ok`, `on-null`, etc.

Get Attribute

The `get-attribute` element must have a `descriptor` element as an ancestor, and provides access to attributes given in the form descriptor.

There's a slight departure from the usual syntax here. The parameter being read from the descriptor file is identified by a `parameter` attribute. If the `get-attribute` element is used within a `validate` element, then its context will be used. The `name` attribute on this element identifies an attribute on the parameter in the descriptor, *not* the parameter itself.

Additional informations can be found on [2.1 version here](#). [<http://cocoon.apache.org/2.1/userdocs/xsp/logicsheet-forms.html>]