NativeSPRequestMapHostRegex

The `<HostRegex>` element is used to apply content rules to requests to any virtual host that matches a regular expression. The input expression is constructed as:
`scheme``://`*`hostname`*`:`*`port`*

Attributes

Content Specifiers

- `regex` (string)
  - Required attribute, specifies the regular expression to apply.

- `ignoreCase` (boolean) (defaults to true)
  - Controls case option in regex engine, should be left on in most cases.
  - Deprecated in V2.6.0

Version 2.6 and Above

- `caseSensitive` (boolean) (defaults to false).
  - If false, the case-ignoring option is set for the regular expression engine.

Content Settings

XML attributes corresponding to request mapper properties are used.

Child Elements

Access Control

One of the following elements can be used to attach an access control policy to the resource. This is a violation of the axiom that the SP doesn't do access control, but it's really just a call-out that has some predefined plugins you can use as examples to create more.

- `<htaccess>`
  - Enables Apache `.htaccess` support during the authorization phase. This is automatic and implicit for the "Native" request mapper, but can be enabled by hand if the "XML" request mapper is used. Note that this will fail for non-Apache servers.

- `<AccessControlProvider>`
  - Attaches a custom access control policy supported by a plugin.

- `<AccessControl>`
  - Attaches an access control policy using the sample XML-based plugin provided with the SP. This is just a short-hand for embedding the policy in the element above, if you want the policy inside the same file.

If no element is included (or inherited or implicitly enabled), any access control is left to the resource.

If an error occurs when processing this element, a dummy policy to deny access is installed to prevent accidental exposure.

Nested Content Specifiers

Zero or more of these "overrides" to match specific content on the virtual host can be included.

- `<Path>`
  - Matches requests whose first path component is an exact match for the element.

- `<PathRegex>`
  - Matches requests with a path that matches the element's expression.

- `<Query>`
  - Matches requests containing a query string parameter satisfying the element.

Matching is done within a `<HostRegex>` element as follows:

1. First, by examining `<Path>` elements in order.
2. Then, by checking any `<PathRegex>` elements in order against the part of the path that was not matched in the first step.
3. Finally, by examining any `<Query>` elements in order.

Once a matching child element is found, the process steps "into" that element and no other siblings will be applied. Thus, siblings cannot overlap.

For more details on how the request mapping process works, see the request mapper HOWTO.