APACHE SLING - KEY CONCEPT

Q 1. How Sling will Call the Script on the basis of url?

Sling first looks up at the resource identified by the URL - typically a path inside the JCR repository, which is annotated by the sling:resourceType property which defines the resource type of that resource. Using this resource type (which is kind of a relative path, eg. "myblog/comment"), scripts or servlets are looked up.

During the Resource Resolution step, the client request URI (as being returned by [HttpServletRequest.getRequestURI() is decomposed into the following parts (in this order):

Resource Path - For existing resources the resource path is the longest match (also considering its mappings) pointing to a resource where the next character is either a dot (.) or it is the full request URI. Otherwise (for a path not matching any existing resource) the resource path ends at the first dot (.) in the request url. The exact logic for retrieving the resource path is implemented at ResourceResolver.resolve (HttpServletRequest,String). It is impossible to tell from just looking at the request URI where the resource path part ends. You have to know the underlying resource structure to know how a URL is decomposed. You cannot safely assume that the resource path will always end at the first dot.

Selectors - If the first character in the request URL after the resource path is a dot (.), the string after the dot up to but not including the last dot before the next slash character or the end of the request URL comprises the selectors. If the resource path spans the complete request URL no selectors exist. If only one dot follows the resource path before the end of the request URL or the next slash, also no selectors exist.

Extension - The string after the last dot after the resource path in the request URL but before the end of the request URL or the next slash after the resource path in the request URL is the extension.

Suffix - If the request URL contains a slash character after the resource path and optional selectors and extension, a path starting with the slash up to the end of the request URL is the suffix path. Otherwise, the suffix path is empty. Note, that after the resource path at least a dot must be in the URL to let Sling detect the suffix.

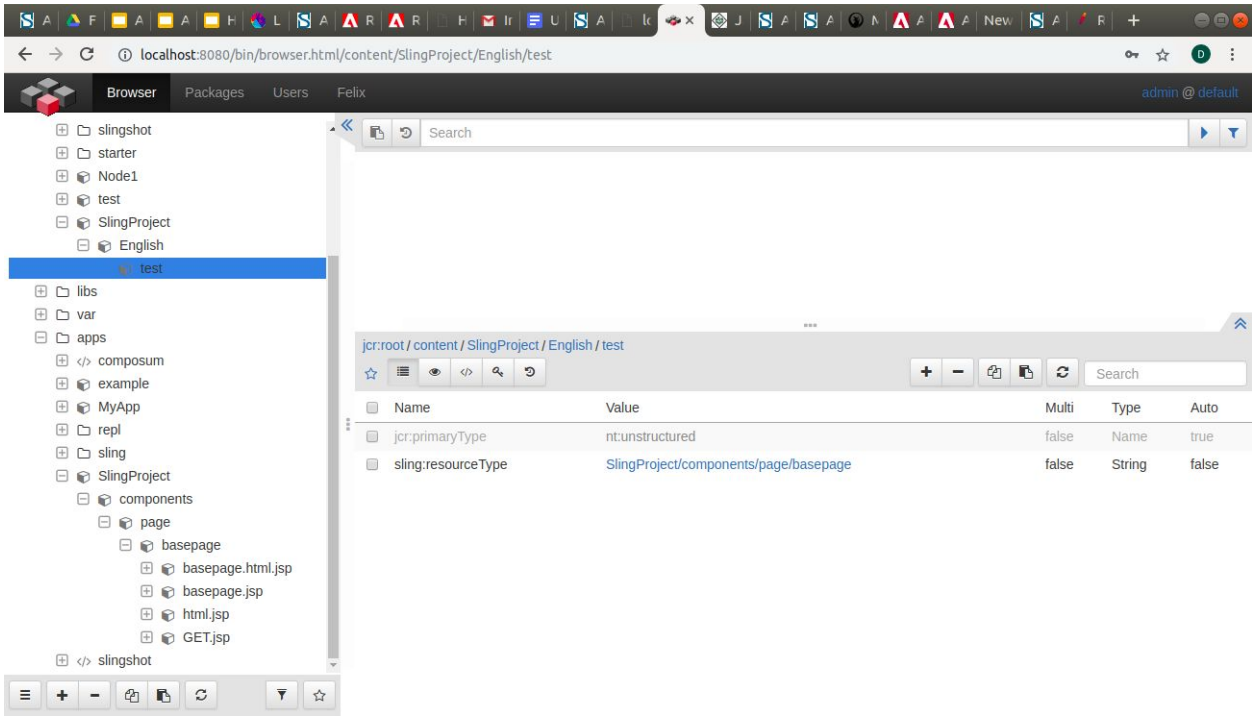
Rules for Script Path Prioritization

1. The more request selectors are matched, the better.
2. A script including the request extension matches better than one without a request extension.
3. A script found earlier matches better than a script found later in the processing order. This means, that script closer to the original resource type in the resource type hierarchy is considered earlier.

Q. 2. Have a page /content/SlingProject/English/test.html which has resourceType SlingProject/components/page/basepage?

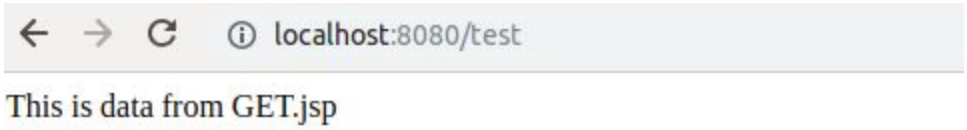
We have 4 script files at "SlingProject/components/page/basepage":
1. basepage.html.jsp
2. basepage.jsp
3. html.jsp
4. GET.jsp

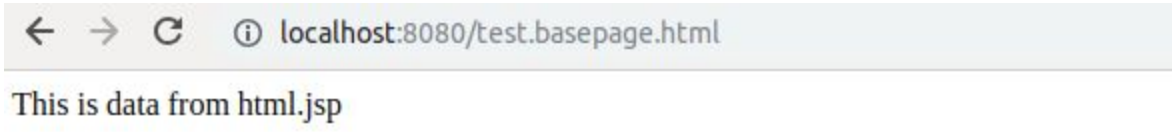
Structure

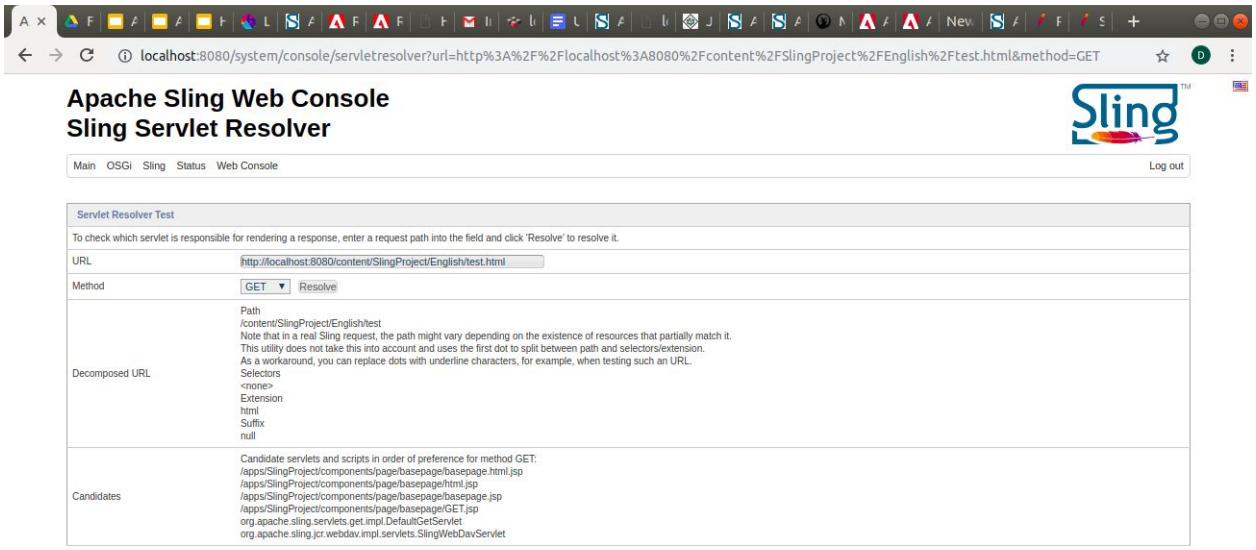


Outputs


localhost:8080/test.html
This is data from html.jsp


localhost:8080/test
This is data from GET.jsp


localhost:8080/test.basepage.html
This is data from html.jsp


localhost:8080/system/console/servletresolver?url=http%3A%2F%2Flocalhost%3A8080%2Fcontent%2FSlingProject%2FEnglish%2Ftest.html&method=GET
Apache Sling Web Console
Sling Servlet Resolver
Main OSGi Sling Status Web Console
Log out
Servlet Resolver Test
To check which servlet is responsible for rendering a response, enter a request path into the field and click 'Resolve' to resolve it.
URL
http://localhost:8080/content/SlingProject/English/test.html
Method
GET
Resolve
Decomposed URL
Path
/content/SlingProject/English/test
Note that in a real Sling request, the path might vary depending on the existence of resources that partially match it.
This utility does not take this into account and uses the first dot to split between path and selectors/extension.
As a workaround, you can replace dots with underline characters, for example, when testing such an URL.
Selectors
<none>
Extension
html
Suffix
null
Candidates
Candidate servlets and scripts in order of preference for method GET:
/apps/SlingProject/components/page/basepage/basepage.html.jsp
/apps/SlingProject/components/page/basepage/html.jsp
/apps/SlingProject/components/page/basepage/basepage.jsp
/apps/SlingProject/components/page/basepage/GET.jsp
org.apache.sling.servlets.get.impl.DefaultGetServlet
org.apache.sling.jcr.webdav.impl.servlets.SlingWebDavServlet