

Wt - Feature #7385

More Dynamic Paths

12/30/2019 11:33 PM - Aaron Wright

Status:	New	Start date:	12/30/2019
Priority:	Normal	Due date:	
Assignee:		% Done:	0%
Category:		Estimated time:	0.00 hour
Target version:			
Description			
<p>I'm using WServer::addResource to add several resources to my application, which works great, but I've found that there are several aspects of the path where I want the resource to be at that I would like to be more dynamic.</p> <p>Currently I add each resource twice so that I can handle the trailing slash in a consistent manner inside of the resource:</p> <pre>Wt::WServer server(argc, argv); server.addResource(&resource, "/foo"); server.addResource(&resource, "/foo/");</pre> <p>I also have resources with natural spaces in the path that i would like to use either a dash or underscore:</p> <pre>Wt::WServer server(argc, argv); server.addResource(&resource, "/foo-bar"); server.addResource(&resource, "/foo_bar"); server.addResource(&resource, "/foo-bar/"); server.addResource(&resource, "/foo_bar/");</pre> <p>This is a little laborious. If regular expressions were supported, this could be expressed in one line:</p> <pre>Wt::WServer server(argc, argv); server.addResource(&resource, "/foo[-_]bar/?");</pre> <p>I attempted to do this myself in a sort of meta-resource, but I wasn't excited about re-implementing the dynamic path (urlParams) feature that already exists, and which I already use.</p>			

History

#1 - 01/02/2020 03:25 PM - Roel Standaert

I think using regex matches could be quite costly if we do that by default. We're also currently doing hierarchical matching rather than just taking the first added resource that matches, so I'm not sure how this would fit in that system. Maybe it's an option to accept a regex object, or have some flag indicating that it's a regex. I think that std::regex is sadly implemented quite inefficiently in most STL implementations, though.

#2 - 01/06/2020 05:17 PM - Aaron Wright

I had similar doubts about regex. I'm currently trying to implement this as a WResource, which also handles dispatching based on the method (GET, POST, etc). Routing web API calls in a generic manner has proven to be a challenge so far. I just wanted to get this on your radar in case you had better ideas.