

Base Language - Feature #6506

implement CGI WebSpeed support and the standard web-disp.p dispatch functionality

06/08/2022 07:01 PM - Greg Shah

Status: New	Start date:
Priority: Normal	Due date:
Assignee:	% Done: 0%
Category:	Estimated time: 0.00 hour
Target version:	version:
billable: No	
vendor_id: GCD	
Description	
Related issues:	
Related to Base Language - Feature #6432: implement the WEB special stream New	

History

#1 - 06/08/2022 07:02 PM - Greg Shah

- Related to Feature #6432: implement the WEB special stream added

#2 - 06/08/2022 07:10 PM - Greg Shah

This task is to implement the core classic WebSpeed implementation for "CGI Wrapper" usage where a .p or .w can be referenced via HTTP and it results in the invocation of the procedure via CGI. The procedure can include any non-interactive 4GL processing and it writes its results as output to the WEB stream ([#6432](#)). This output is returned back as the HTTP response.

I understand that the core logic for the dispatching is in web-disp.p which is part of Posset (and newer versions in the ADE tools).

We will NOT implement any support (at this time/in this task) for the following:

- Static HTML compiled to a procedure
- Embedded SpeedScript
- HTML Mapping

Marian: Please correct me if I have a misunderstanding on any of this.

#3 - 06/09/2022 02:24 AM - Marian Edu

Greg Shah wrote:

This task is to implement the core classic WebSpeed implementation for "CGI Wrapper" usage where a .p or .w can be referenced via HTTP and it results in the invocation of the procedure via CGI. The procedure can include any non-interactive 4GL processing and it writes its results as output to the WEB stream ([#6432](#)). This output is returned back as the HTTP response.

You are correct, you basically get the output stream, a couple of preprocessor definitions that writes to the said stream and a bunch of API's that gives you access to the web request information - as query string, form data, CGI variables.

I understand that the core logic for the dispatching is in web-disp.p which is part of Posset (and newer versions in the ADE tools).

This is what they use internally, but there is no need to do the exact same thing - years back like in another life I had this little webspool replacement called ganimede (<https://sourceforge.net/projects/ganimede/>) and things haven't changed much in that area since then.

We will NOT implement any support (at this time/in this task) for the following:

- Static HTML compiled to a procedure
- Embedded SpeedScript
- HTML Mapping

Yes, this is covered by the compiler mainly, you could probably pull that out too in conversion phase but imho it's not worth it - that was not largely adopted so I wouldn't bother much about it, it's very old technology anyway so better wait for a client that needs that.

Marian: Please correct me if I have a misunderstanding on any of this.

None that I could think of, that seems about right but as said you probably shouldn't bother about Possenet code nor the web-disp.p.