

Wt - Feature #491

allow threads to be created by library user

08/31/2010 07:04 PM - Wim Dumon

| | | | |
|--|--------|------------------------|------------|
| Status: | New | Start date: | 08/31/2010 |
| Priority: | Normal | Due date: | |
| Assignee: | | % Done: | 0% |
| Category: | | Estimated time: | 0.00 hour |
| Target version: | | | |
| Description | | | |
| See http://redmine.emweb.be/boards/2/topics/601 | | | |
| Allow user to create his own thread. As such he may specify stack sizes, priorities, processor affinity, store thread local data, ... When fully configured, the newly created thread should call some Wt (WServer?) entry point. | | | |

History

#1 - 09/02/2010 07:24 PM - Koen Deforche

Hey Wim,

in what way can a user not do all this while handling a request ?

i.e. check the value of a particular thread local variable, if it is 0, then do all this and set a value for that thread local variable (so that you do not repeat this work again).

The overhead of checking the value for that thread local variable on each request is negligible compared to how many times the library does this internally (when calling `WApplication::instance()`).

koen

#2 - 09/21/2010 01:49 AM - Christophe Delépine

Hi Koen,

I have my own "Thread" C class that offers communication services (messages, synchronization..).

When handling a request, i need to send messages and synchronize with my application's threads.

Since Wt creates threads by itself, i cannot use any method from my Thread class since the thread in which the request is processed is not a Thread instance but a Wt thread.

Therefore i had to use a non multithreaded build of Wt in order to have control on the creation of the thread that runs the web server and event loop.

Besides, it is more natural to assign priority, processor affinity.. at the very beginning of the thread main entry point and not wait for a request that may arrive much later or never..

Christophe