

Wt - Feature #2754

WServer.addApplication()

03/05/2014 01:51 AM - I. Lazaridis

Status: New	Start date: 03/05/2014
Priority: Normal	Due date:
Assignee:	% Done: 0%
Category:	Estimated time: 0.00 hour
Target version:	
Description	
Currently, one has to write:	
<pre>server.addEntryPoint(Application, boost::bind(&createApplication, _1, _2), BlogUrl);</pre>	
This could be simplified to:	
<pre>server.addApplication(boost::bind(&createApplication, _1, _2), BlogUrl);</pre>	

History

#1 - 03/11/2014 03:06 AM - I. Lazaridis

from [RE: Blog Examples and Wt Best Practices](#)

> > In this high-level code, one should not be forced to use "boost::bind(&createApplication, _1, blogDb)". This is just too much (this could be simplified, by changing the API call).

> > I don't understand the comment about boost::bind(): how could the API be changed to avoid this?

>

> It starts with saying: "I want this, thus it's simpler for users". Then it's either solvable, or not. The ideal thing would be something like:

>

> server.add(MyWtApplication, "/mypath")

>

> (I'll open a topic on this, as this interests me technically, too)

And MyWtApplication is a class or a function?

Don't know yet. The matter is general (at this point) and the question is:

If there is a solution to this, would you accept it?

#2 - 03/11/2014 10:17 AM - Koen Deforche

I. Lazaridis wrote:

> And MyWtApplication is a class or a function?

Don't know yet. The matter is general (at this point) and the question is:

If there is a solution to this, would you accept it?

Yes.

Although I would probably be inclined to rather use this to simplify WRun() as for example used in the hello world example, than to simplify WServer.

The idea would then be to simplify:

```
WApplication *createApplication(const WEnvironment& env)
{
    return new HelloApplication(env);
}
```

```
}  
  
int main(int argc, char **argv)  
{  
    return WRun(argc, argv, &createApplication);  
}
```

with

```
int main(int argc, char **argv)  
{  
    return WRun(argc, argv, HelloApplication);  
}
```

Regards,

koen

#3 - 03/11/2014 11:26 AM - I. Lazaridis

This sounds like a better starting point.

With no additional parameters, one can already write this:

```
int main(int argc, char **argv)  
{  
    return WRun(argc, argv, HelloApplication::create);  
}
```

HelloApplication has to provide a static create function:

```
static WApplication *create(const WEnvironment& env)  
{  
    return new HelloApplication(env);  
}
```

The next step would be to try to eliminate the "::create" part in the WRun (possibly not possible).

Then to find a way for handling the additional parameters without the need of bind (and without any magic), etc. etc.

I know myself, I will not find peace if I don't find a solution for this (which makes everyone happy, users, you, me).