Subject: Webtoolkit
Posted by mirek on Fri, 22 Feb 2008 12:53:07 GMT
View Forum Message <> Reply to Message

This sounds interesting:

http://www.webtoolkit.eu/wt/doc/tutorial/wt-sdj.xhtml

I would be interesting to do using U++ style...

Mirek

Subject: Re: Webtoolkit
Posted by zsolt on Fri, 22 Feb 2008 14:46:10 GMT
View Forum Message <> Reply to Message

I was thinking about a reimplementation of Wt as well.
The idea is very-very good, but the current implementation and API of Wt are extremly complex, if you know U++ style of programming.

Btw I would prefer a much more simple design in the first iteration:
* A scalable web server class (basically single threaded, but passing dynamic requests to threads)
* A simple infrastructure to handle requests, sessions
* An infrastructure to register cpp files and map them to URLs (something like test registration in UnitTest++)
* A simple templating engine to generate HTML.

Subject: Re: Webtoolkit
Posted by Mindtraveller on Fri, 22 Feb 2008 16:32:52 GMT
View Forum Message <> Reply to Message

Some time ago I had an idea of making site. Yes, C++ would be very attractive way of making site (at least for me who knows C++ and doesn`t know much of PHP).

The main problem I`ve met was actually not generating answers and parsing html (but that was a problem too). The main thing was supporting a number of browsers. This is very specific web-only thing, which requires strong JS+HTML+CSS experience. So, the thing actually required most was visual style of making site pages which would support different browsers without actually programming these differences. I mean something like theIDE template editor, but for pages (like Macromedia Dreamweaver visual editor?).
Cross-platform. Cross-browser.

Subject: Re: Webtoolkit

Posted by mr_ped on Fri, 22 Feb 2008 18:27:22 GMT
View Forum Message <> Reply to Message

I'm certainly interested into these things too.

But I'm not sure I would like to see U++ custom web server, I mean I would prefer something well used, robust and powerful, like apache for example.

That mixed with TheIDE support and some easy way how to run C++ modules (or some easy tutorial) would really made my day.

And thinking about some common classes for parsing GET/POST variables (especially to handle the security aspect of these things easily and almost in fool-proof way, so you don't have to think about security checks every time you need some GET variable from user) makes me excited even more.

I wonder if the WT is the only existing solution.
Actually I have strong suspicions that to write a c++ module for apache is definitely possible and probably encouraged, if you know apache well enough... but my knowledge about apache is on "using it" level only, did some local installations of it too, but never really bothered to take a look how it works.

Maybe all we need is some good tutorial how to set up existing technologies to work together with U++, and couple of simple U++ - like classes for making parsing and HTML output easy.

Subject: Re: Webtoolkit
Posted by mirek on Fri, 22 Feb 2008 18:54:48 GMT
View Forum Message <> Reply to Message

Well, Tom has implemented a couple of websites/webserves using C++/U++ and it really is not that much complicated.

I have experience with only single intranet application, which was abandoned later anyway. In fact, if I remember well, Tom had somewhat similar system for his apps as Webtoolkit; but I have found direct generation of websites much more simple.... (without emulating "widgets"). But of course, perhaps depends on the kind of application.

Mirek

Subject: Re: Webtoolkit
Posted by mirek on Fri, 22 Feb 2008 18:58:33 GMT
View Forum Message <> Reply to Message

mr_ped wrote on Fri, 22 February 2008 13:27I'm certainly interested into these things too.

But I'm not sure I would like to see U++ custom web server, I mean I would prefer something well

used, robust and powerful, like apache for example.

Well, the problem there is that not too much is left for apache to do in such case. Of course, if you have mixed content (some pages C++, some static or other code), it has some merit. Otherwise, parsing http requests is quite easy.

Quote:
And thinking about some common classes for parsing GET/POST variables (especially to handle the security aspect of these things easily and almost in fool-proof way, so you don't have to think about security checks every time you need some GET variable from user) makes me excited even more.

Check HttpServer in the Web.

Quote:
Maybe all we need is some good tutorial how to set up existing technologies to work together with U++, and couple of simple U++ - like classes for making parsing and HTML output easy.

All is already there. OTOH, I suspect the idea of HTML expressions is a little bit over the head (tried to repeat SqlExp success there, but HTML seems to be another kind of beast...).

BTW, I seem to be working for internet company now, maybe there will be some new web programming means in U++ soon (Unfortunately, so far I am working on some backend with PHP frontend, so not direct HTML for now

Mirek

Subject: Re: Webtoolkit
Posted by zsolt on Fri, 22 Feb 2008 22:51:31 GMT
View Forum Message <> Reply to Message

Quote:But I'm not sure I would like to see U++ custom web server
I'm sure, I would.
One of the most important problems with web-application development environments (Perl, PHP) is, that you have no useful debugger.

Just imagine, that you could run and debug your U++ web app inside TheIDE. Development of an apache module is a nightmare compared to it.

The other problem with apache is, that it uses a process for every client socket, so you have to save every session information to a file or to database.

Subject: Re: Webtoolkit
Posted by zsolt on Fri, 22 Feb 2008 22:53:24 GMT
View Forum Message <> Reply to Message

Quote:OTOH, I suspect the idea of HTML expressions is a little bit over the head
I agree.

Subject: Re: Webtoolkit
Posted by mr_ped on Fri, 22 Feb 2008 23:42:18 GMT
View Forum Message <> Reply to Message

Thank you all for advices and new directions where to look.

Basically I'm just toying around with idea of creating yet another web multi user game, so my main problems with custom web server are:
- will it be secure enough? (let's say I would have 1k - 10k registered players, I bet at least 0.1% of them will be capable to try all kind of exploits)
- will it handle hundreds to thousands users simultaneously?
- how difficult is to maintain per user sessions in C++ (with connection to previous point), and whole authorization stuff?

I know very little about these things right now, so I have to learn few things about web servers if I ever will try to do that project, still if you have some patience with me and ideas what should I check/study, thank you.

Subject: Re: Webtoolkit
Posted by zsolt on Sat, 23 Feb 2008 18:34:36 GMT
View Forum Message <> Reply to Message

mr_ped wrote on Sat, 23 February 2008 00:42
- will it handle hundreds to thousands users simultaneously?

Check The C10K problem document.
This is the best starting point to every developer, starting a new server project. It describes all type of server applications and all of their strengths and drawbacks. It explains why apache like servers have so pure performance.

And check the best socket library if you want to develop a really efficient server app. Maybe it would be included in U++ with an U++ wrapper layer.

Subject: Re: Webtoolkit
Posted by mr_ped on Sun, 24 Feb 2008 01:38:50 GMT

"Check The C10K problem document."

Thank you, it gave me lot of new information. Unfortunately it just made me sure I don't want to code my own web server at all.
I would prefer to get all those "handle 10k clients in 200 threads" for granted from somebody else, and focus only on the application itself.

Of course your point about debugging the application directly in TheIDE is correct. So in ideal world ... what about producing an apache (or other powerful web server) module in TheIDE and having the option to quickly to connect it on custom U++ web server with same API? I think it will turn out to be impossible for many many reasons (or possible only in case the new module would use very limited API set, thus making it useless in general), but I'm writing it here to show my way of thinking and what is interesting for me.

Back to topic, I also read something more about WT itself, and I don't need anything like that right now. Maybe one day it will be become very handy, because it may hide all those nasty browser difference from developer. Than again with late efforts of Mozilla, Opera and KHTML/WebKit to conform to W3C standards I don't feel the pressure to worry about browser incompatibilities. (I will reserve my right to ignore hopeless existence of IE .. for me it is dead and I don't care anymore)

I think doing simple HelloWorld.html in C++ is overkill, even with WT. I mean you still have to define some class, some constructor, etc... And I bet if I ever do some web project, it will still consist of many many simple html pages, even if there will be some horrible complex beast hidden inside.
So far I think I should rather look how to mix PHP and C++ in efficient and simple manner, that would suite my taste much more (and hopefully my needs too).

Subject: Re: Webtoolkit
Posted by mirek on Sun, 24 Feb 2008 07:55:58 GMT

mr_ped wrote on Sat, 23 February 2008 20:38"Check The C10K problem document."

Thank you, it gave me lot of new information. Unfortunately it just made me sure I don't want to code my own web server at all.
I would prefer to get all those "handle 10k clients in 200 threads" for granted from somebody else, and focus only on the application itself.

Actually, Apache is not much of help here. If you want or need to store or cache state info in memory (and you want if you want to be fast), you will need those threads anyway.

Quote:
Of course your point about debugging the application directly in TheIDE is correct. So in ideal world ... what about producing an apache (or other powerful web server) module in TheIDE and

having the option to quickly to connect it on custom U++ web server with same API?

Yep. This is how existing Web infrastructure works, albeit not for Apache (but Win32 ISAPI) - they can run as standalone http server or as ISAPI module, based on commandline parameter. In fact, this part is relatively easy...

Quote:
So far I think I should rather look how to mix PHP and C++ in efficient and simple manner, that would suite my taste much more (and hopefully my needs too).

It is easy to do PHP style programming in C++. After all, HTML is just text and we have String, do not we?

Mirek

Subject: Re: Webtoolkit
Posted by cbpporter on Sun, 24 Feb 2008 21:16:12 GMT
View Forum Message <> Reply to Message

It is easy to do PHP style programming in C++. After all, HTML is just text and we have String, do not we?
We do, there's no arguing here . But still, if this becomes a reality, I would like to see more than HTML/text output. A C++ Servlet or "echo" equivalent wouldn't be that great. I would like to see a framework which removes as much of the messy job of browser "programing" as possible. For example, from the myriad of web toolkits that exist, I think that Tapestry 4.x offer a nice balance between high level code, small quantities of glue code, and dynamic HTML that is based on templates which can be created in in a visual editor or fed directly to an online HTML/CSS compatibility checker.
