Subject: U++ infrastructure server...
Posted by mirek on Wed, 09 Jul 2008 12:27:12 GMT
View Forum Message <> Reply to Message

Well, I am getting to the upgrade of that uvs2 server sitting on my LAN to something more powerful to run "nightly builds" etc...

Well, seems like I am finally upgrading my main desktop and will move it to this purpose. So very soon there will be Core2 CPU with 320GB HD (will add more if necessarry) a 2GB ram available for the task.

Now the question is what exactly to install.

Basic idea is that U++ core developers should have sufficient access to this machine to work with all release and testing scripts. I am now thinking about single "upp" user with SSH keys...

We will need Ubuntu32, Ubuntu64 and Win32 environments. The question of course is how.

Win32 IMO can be quite well managed using Wine. Anyway, the problem still is 32/64.

IMO the options are VMs (this would also allow having XP installed.. but the question is how it should be remotely managed) or chroot.

In any case, the base system will be Ubuntu 64 most likely.

Thoughts?

Mirek

Subject: Re: U++ infrastructure server...
Posted by mdelfede on Wed, 09 Jul 2008 15:21:50 GMT
View Forum Message <> Reply to Message

I think (thought I must test it, but) that it's possible to build 32 bit upp on ubuntu 64 bit.
It should be possible because, for example, wine is 32 bit and it can be built on 64, given needed libraries are installed.
So, one problem should go away
BTW, you can even run upp 32 bit on 64 bit machine, if you've got the needed 32 bit libs installed.
The opposite (building for 64 bit on 32) if possible it's quite more difficult.

For windows builds, I think with wine we wouldn't have problems neither. You can make a batch file and run with wine through cmd.exe (or command.com). Easy task too... I'm using wine quite a lot for autocad and I can set it up well.

About access to server, you could setup also a per-developer username/password, and limit access to sensible stuff just when needed. It sounds paranoid, maybe, but for my experience more harm can come by mistakes than by viruses/malware

With acl you can fine-tune user access on server, even it's not too simple to setup.... but the setup is just once.
With that way you could also separate docs access from main three access, giving more users the ability to contribute to documentation without leaving code access open to all.

Ciao

Max

Subject: Re: U++ infrastructure server...
Posted by captainc on Wed, 09 Jul 2008 16:20:14 GMT
View Forum Message <> Reply to Message

While I'm not an official Upp developer, I suggest the use of VMs. I have built Upp and Upp apps with great success using Windows XP in a VirtualBox VM with Ubuntu as the Host. Compilation speeds are close to native. The only snag is that the VM only allows for using a single core per operating system. Other VMs might be able to take advantage of multi-core processors. In any case, this won't be an issue if you have multiple VMs running, which would then be able to utilize the dual-core processor.

VirtualBox has the fastest VM solution I have seen when using a UI. It beats the pants off of VMWare for this, but I believe a Xen like solution would be best if you didn't use the desktop UI capabilities.

On a final note, I have Vista 64-bit running on one machine and a 20mbit/5mbit internet connection. I would not be opposed to setting something up if you require a build for Vista 64 platform.

Subject: Re: U++ infrastructure server...
Posted by mdelfede on Wed, 09 Jul 2008 17:31:52 GMT
View Forum Message <> Reply to Message

captainc wrote on Wed, 09 July 2008 18:20While I'm not an official Upp developer, I suggest the use of VMs. I have built Upp and Upp apps with great success using Windows XP in a VirtualBox VM with Ubuntu as the Host. Compilation speeds are close to native.

I've been using vmware on ubuntu host for about 1.5 years, after the *very last* virus attack on win xp... and I was very happy with it too.
But, it does one caveat... You must assign a ram size to the machine, which is locked by it (AFAIK...).
In particulaw with windoze guests, it becomes quickly memory hungry.

Quote:

The only snag is that the VM only allows for using a single core per operating system. Other VMs might be able to take advantage of multi-core processors. In any case, this won't be an issue if you have multiple VMs running, which would then be able to utilize the dual-core processor.

VmWare can use all processors on a single machine... but I agree that it's not the most important stuff, in particular with a build server. You don't need a lightning speed for building, IMO.

Quote:
VirtualBox has the fastest VM solution I have seen when using a UI. It beats the pants off of VMWare for this, but I believe a Xen like solution would be best if you didn't use the desktop UI capabilities.

On a final note, I have Vista 64-bit running on one machine and a 20mbit/5mbit internet connection. I would not be opposed to setting something up if you require a build for Vista 64 platform.

I'm using wine since I could run autocad on it, so by now I can see the difference.... and wine is usually a bit faster (and less memory hungry) than VM.
But you touched the *only* true caveat of wine... It's limited to 32 bit windows apps. So, no build for 64 bit on it...

Max

Subject: Re: U++ infrastructure server...
Posted by mezise on Wed, 09 Jul 2008 20:13:18 GMT
View Forum Message <> Reply to Message

VMs will be better for build platform if MacOS and other operating systems are meant to be supported by Upp.

Subject: Re: U++ infrastructure server...
Posted by mirek on Wed, 09 Jul 2008 20:35:04 GMT
View Forum Message <> Reply to Message

mdelfede wrote on Wed, 09 July 2008 13:31
VmWare can use all processors on a single machine... but I agree that it's not the most important stuff, in particular with a build server. You don't need a lightning speed for building, IMO.

It might be important for unit testing.

Mirek

Subject: Re: U++ infrastructure server...
Posted by mr_ped on Thu, 10 Jul 2008 08:56:55 GMT
View Forum Message <> Reply to Message

luzr wrote on Wed, 09 July 2008 22:35mdelfede wrote on Wed, 09 July 2008 13:31
VmWare can use all processors on a single machine... but I agree that it's not the most important stuff, in particular with a build server. You don't need a lightning speed for building, IMO.

It might be important for unit testing.

Mirek

It's just game of words as the technology is pretty much the same, but you mean automated tests, right?
Because unit tests, if they should be run after each build after every little change, they have to take 2-5seconds at most (I'm usually around 0.1s to 0.5s in my small projects) to not make you sad. Usually I can cover 100% of code with O(1) and O(small N) tests, and any thorough O(n) and more tests I move into automated application tests, which I don't run after every build like Unit tests. Those I run only when I did finish some step of development.

Anyway, if the tests are not written to use multi-core, they will not benefit from it anyway. And the easiest way to use multi-core is to run multiple single core tests at the same time. If we assign each platform single core (and I think you will sooner run out of cores, than out of platforms), I think the overall performance will be ok. Even if some core will be bored occasionally.

Subject: Re: U++ infrastructure server...
Posted by mirek on Thu, 10 Jul 2008 13:07:11 GMT
View Forum Message <> Reply to Message

mr_ped wrote on Thu, 10 July 2008 04:56luzr wrote on Wed, 09 July 2008 22:35mdelfede wrote on Wed, 09 July 2008 13:31
VmWare can use all processors on a single machine... but I agree that it's not the most important stuff, in particular with a build server. You don't need a lightning speed for building, IMO.

It might be important for unit testing.

Mirek

It's just game of words as the technology is pretty much the same, but you mean automated tests, right?
Because unit tests, if they should be run after each build after every little change, they have to take 2-5seconds at most (I'm usually around 0.1s to 0.5s in my small projects) to not make you sad. Usually I can cover 100% of code with O(1) and O(small N) tests, and any thorough O(n) and

more tests I move into automated application tests, which I don't run after every build like Unit tests. Those I run only when I did finish some step of development.

Anyway, if the tests are not written to use multi-core, they will not benefit from it anyway. And the easiest way to use multi-core is to run multiple single core tests at the same time. If we assign each platform single core (and I think you will sooner run out of cores, than out of platforms), I think the overall performance will be ok. Even if some core will be bored occasionally.

Yep, automated testing.

Anyway, the real point is that we are about to test MT stuff too. Means we need multithreaded tests that really run on multiple cores. Some bugs are revealed only this way.

Mirek