Subject: Assist - rescan code vs caching in files....
Posted by mirek on Wed, 10 Sep 2008 09:48:38 GMT
View Forum Message <> Reply to Message

Seeking for advice

Assist++ has quite complex mechanism that stores parsed information into files (in 'cfg' directory) when theide exists and reloads it from there on theide start.

Recently I have started to worry if this is a good idea.

Well, the primary reason to worry is that the mechanism for storing and retrieving is quite complex and fragile.

Maybe it would be OK to rescan the code each time theide starts?

So the question is:

- if you load "ide" package, how long it takes initial start sequence (most of it is spent loading assist++ from file cache).

- can you compare this to invoking "Assist/Rescan code"?

Mirek

Subject: Re: Assist - rescan code vs caching in files....
Posted by mr_ped on Wed, 10 Sep 2008 10:09:47 GMT
View Forum Message <> Reply to Message

First time opening "ide" package:
14s (src files uncached, and probably not scanned)
Rescan:
4s (since here src files are probably cached by OS in RAM)
Exit+Reopen IDE:
1s
Rescan:
4s

BTW, what about full rescan after start of IDE in the background?
In case that one will take around 10-30s even with more intelligent parser, I think that's perfectly acceptable solution, I usually stare into the source files for more than a minute or two before I catch up where I left and start to write something.

edit: the important thing is to let me "stare" at them, that's the reason why it should run in background. The current modal foreground rescan or load after opening package annoys me. It's just 1-5s, but can be zero in ideal world of background scanning .

Subject: Re: Assist - rescan code vs caching in files....
Posted by mirek on Wed, 10 Sep 2008 10:54:55 GMT
View Forum Message <> Reply to Message

mr_ped wrote on Wed, 10 September 2008 06:09
BTW, what about full rescan after start of IDE in the background?

Well, I am afraid that making more inteligent caching system is much easier than scanning in background

Moreover, I believe that in process of "staring", you would soon start requiring class browser....

Quote:
edit: the important thing is to let me "stare" at them, that's the reason why it should run in background. The current modal foreground rescan or load after opening package annoys me. It's just 1-5s, but can be zero in ideal world of background scanning .

Well, I think I will fix the caching

Mirek

Subject: Re: Assist - rescan code vs caching in files....
Posted by kodos on Wed, 10 Sep 2008 11:17:13 GMT
View Forum Message <> Reply to Message

I don't know if it is worth to put an effort in the caching. After all you don't open a project every 5 minutes. And with the caching there could be problems with updates of the code from outside TheIDE, couldn't there?

Subject: Re: Assist - rescan code vs caching in files....
Posted by mirek on Wed, 10 Sep 2008 12:17:36 GMT
View Forum Message <> Reply to Message

kodos wrote on Wed, 10 September 2008 07:17I don't know if it is worth to put an effort in the caching. After all you don't open a project every 5 minutes. And with the caching there could be problems with updates of the code from outside TheIDE, couldn't there?

Not really. Compared to other assist++ problems, this one is trivial

Mirek