Subject: Re: Sharing and Locking
Posted by gridem on Sun, 07 Mar 2010 15:38:36 GMT
View Forum Message <> Reply to Message

Mirek, thank you for your answer.
luzr wrote on Sun, 07 March 2010 16:40
1) Ptr and Pte are VERY different beasts as compared to shared_ptr (but you probably know that).

Yes, it different but tries to solve the same kind of problems IMO. I think that shared_ptr has more cleared semantics than Pte/Ptr. May be the reason of this is that I used shared_ptr a lot before and try to use Pte/Ptr like shared_ptr.
luzr wrote on Sun, 07 March 2010 16:40
2) They DO use atomic operations, therefore within its modus operandi, they are completely MT safe (as is or should be the whole U++ now).

Mirek

So I mean that Pte uses Mutex (more precisely, StaticMutex) and in general it can have some problems in high concurency application. Lock-free implementation like atomic operations produces better performance results in general.

The original problem starts from the task to provide the cache in high-loaded service with limited amount of memory.