Subject: Re: changed CoWork and derived WorkQueue
Posted by kohait00 on Tue, 09 Feb 2010 14:14:22 GMT

View Forum Message <> Reply to Message

well, maybe it's because *I* have too little MT experience, i am a bit of a rookie on that indeed, still learning.

my situation was basicly that i wanted to keep the gui responsive (old stuff) offloading work (i realize that CoWork is mainly a loop parallelizer, for independant work). but as soon as the offloaded work is kind of depandant on each other or the order of execution, you only have the possibility to either PostCallback the work using the GUI thread (which for large work affects GUI respnsiiveness) or to use a 1-thread-workqueue, which executes in background in order of comits. both solutions *are* a WorkQueue, even PostCallback. With current CoWork this is *not* possible. for stated reason: CoWork is "only" a loop parallelizer..not a work scheduler. with the given changes, one could flexibilize that.

anyhow..i wont push that, dont feel bothered. you decide. i am working with the modified CoWork, and it seems to me relatively more fluent to post to WorkQueue (1 trhead CoWork) than to a threadpool CoWork.