
Subject: Helper function to help dealing with GUI thread related issues

Posted by [Didier](#) on Thu, 01 May 2014 23:44:40 GMT

[View Forum Message](#) <> [Reply to Message](#)

Hi all,

Doing some corrections to an application I ran into the classic GUI thread Vs other thread (trying to do GUI things) problem

My problem was the following: I have an import thread that imports data into a DB and from time to time it needs to ask questions in order to solve conflicts ... so it creates dialogs ... and this can only be done in GUI thread (as stated in documentation, `GuiLock` is not sufficient for this). So I decided to do a helper function that can be used by others, it does the following:

- * Puts the originating thread in wait state
- * runs the dialog in the GUI thread
- * Once the dialog finished, it releases the originating thread

All you need to use it is a Callback to the GUI action you need to be done, all the synchronisation is done automatically

```
void GuiExecWait(Callback cb, int execDelay_ms = 20) {  
    Mutex mtx;  
    mtx.Enter();  
    {  
        // enter new context to avoid mtx destruction before cond destruction ( due to bad compiler  
optimizations )  
        ConditionVariable cond;  
        cb << callback( &cond, &ConditionVariable::Signal );  
        SetTimeCallback(execDelay_ms, cb);  
        cond.Wait(mtx);  
    }  
}
```

Maybe this helper function can be added to core ?