
Subject: Re: Thelde continuously using CPU
Posted by [mirek](#) on Thu, 14 Sep 2023 09:43:51 GMT
[View Forum Message](#) <> [Reply to Message](#)

jjacksonRIAB wrote on Wed, 13 September 2023 16:53 After running it I saw the event loop, that was not the issue. I checked the other threads:

ide/background.cpp

```
void GatherAllFiles(const String& path, Index<String>& filei, VectorMap<String, String>& file)
{
    Sleep(0); // This is supposed to be superlazy
    for(FindFile ff(path + "/*.*"); ff && !Thread::IsShutdownThreads(); ff.Next())
        if(ff.IsFolder() && *ff.GetName() != '.')
            GatherAllFiles(ff.GetPath(), filei, file);
        else
            if(ff.IsFile()) {
                String p = NormalizePath(ff.GetPath());
                String lp = ToLower(p);
                if(filei.Find(lp) < 0) {
                    filei.Add(lp);
                    file.Add(GetFileName(p), p);
                }
            }
}
```

That Sleep(0) seems to be not very superlazy at all. If I set it to 10 the CPU usage drops. Could this be that Linux handles sleep differently from Windows? I can see the unix version is using nanosleep and I'm not sure what that does with a param of 0 but I'm assuming from what I've read it does nothing... which instead of making it lazy makes it expensive.

That is weird. If you look one level up

```
void IdeBackgroundThread()
{
    while(!Thread::IsShutdownThreads()) {
        VectorMap<String, String> file;
        Index<String> dir;
        Index<String> filei;

        for(FindFile ff(ConfigFile("*.var")); ff && !Thread::IsShutdownThreads(); ff.Next()) {
            VectorMap<String, String> var;
            LoadVarFile(ff.GetPath(), var);
            for(String d : Split(var.Get("UPP", ""), ','))
```

```

    dir.FindAdd(NormalizePath(d));
    Sleep(0);
}
for(String d : dir)
    GatherAllFiles(d, filei, file);
{
    Mutex::Lock ____(s_allfiles_lock);
    s_allfiles = pick(file);
    s_allnests = dir.PickKeys();
}
for(int i = 0; i < 10 && !Thread::IsShutdownThreads(); i++)
    Sleep(100);
}
}

```

It should wait before doing this for 1 second. The whole purpose of the exercise is to have somewhat actual list of all files of all assemblies (this is then used in comparison menu where now all files with the same name that are somewhat accessible through any assembly are listed - simplifies comparison sources between branches/versions). Putting Sleep(10) into GatherFiles would make it too long to happen.

Can you experiment with that loop at the end? IDK, maybe IsShutdownThreads is broken?

BTW, the idea behind Sleep(0) is to give up CPU if there is more important work to do (this is Thread::StartNice).

Mirek
