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.