Threading in LibreOffice & COOL

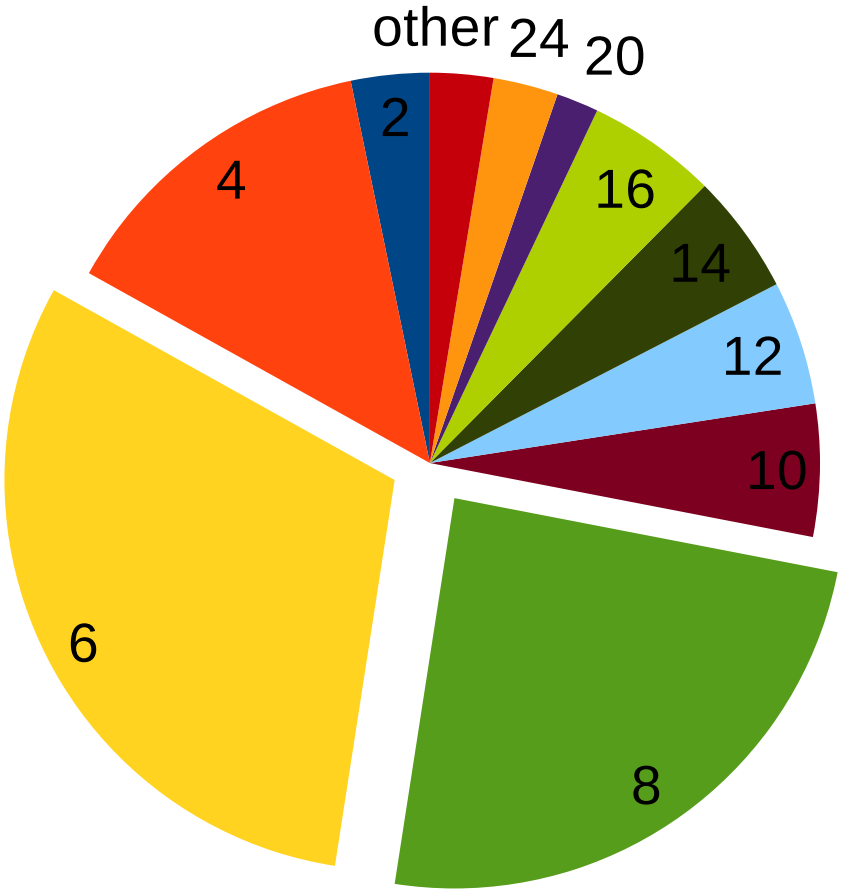
Michael Meeks

“Stand at the crossroads and look; ask for the ancient paths, ask where the good way is, and walk in it, and you will find rest for your souls…” - Jeremiah 6:16

Collabora Online

Steam Hardware Survey # of threads

Almost everyone has at least 4 threads, 50% have >8 threads



Collabora Online

Some sample threads ...

We use lots of threads:

- soffice.bin - VCL main thread
- PipeIPC → remote command argument thread
- configmgrWriter
- salhelper::Time
- comphelper::ThreadPool::getSharedOptimalPool() ...
- On Linux: gdbus, gmain, dconf worker etc.

Windows: worse → GDI + real-time timer threads

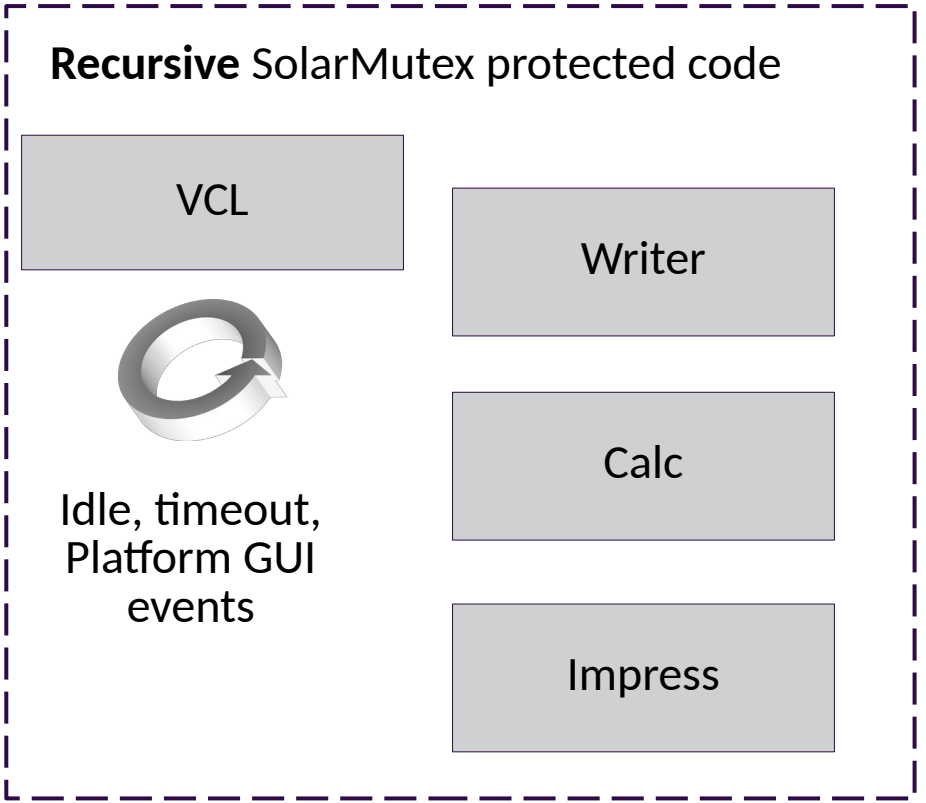
Approximately 1+epsilon – zero concurrency

Collabora Online

A whole thread for timing ?

- vcl::Timer:
  - Relies on the main-loop spinning …
- eg. File import:
  - Want to update / re-render status bar every say 50ms
  - Fetching accurate time – (was) surprisingly expensive …
  - Instead have a waiting thread & a simple atomic to check.
- So we get a timer thread event-loop
  - Just sleeps, calls a callback, sleeps etc. …

Overview of threading ...



Task specific helper threads ...

BitmapScaleSuperFilter, GraphicFilter::ImportGraphics ...

UNO components / APIs (in theory thread-safe)

configmgr

XFastParser

Decompress + Parse + Tokenize Thread

Simple example … package/

Deflation is slow

- Ideally we all use **zstd - ODF needs to catch up**

Lots of streams in files

- rZipOut.addDeflatingThreadTask …
  - Do the deflation in parallel and async – finish up before writing.

But … ODF's content.xml …

- Aggregates all sheets of a giant spreadsheet into one XML file (!?) …

```
if (estimatedSize > 1000000)
{
    // Use ThreadDeflater which will split the stream
    // into blocks and compress them in threads, but
    // not in background (i.e. writeStream() will block).
    // This is suitable for large data.
```

Collabora Online

More fun example … calc formulae

Ideally lock-less & highly concurrent …

- Cut out a ‘safe’ part of the formula engine …
- Find a long formula-group – down a column, with a complex formula
- ScFormulaCell::CheckComputeDependencies
  - check they are all already calculated => we don’t recurse.
- ScFormulaCell::InterpretFormulaGroupThreading …
  - Try to have independent copies of some data we need
  - Try to ensure locking on eg. SvNumberFormatter
  - Subset to ignore corners: volatile formulae / macros etc. …

Even more fun example … calc XLSX

Parallel sheet parsing …

- Problem: much of the code must be protected by SolarMutex
- Problem: the very inner loop simply parsing / importing cells is 95% of the work
- Problem: the SolarMutex is recursive – and always held over calc.

Solution – some sort of inside-out locking

- SolarMutexReleaser → spawn threads
  - Thread takes SolarMutex
  - Releases it only around the performance critical pieces.

COOL bits ...

Lots of threading ... & message passing

Thread outline …

SocketPoll

General Threads:

- **accept_poll →** accepts incoming HTTP / websockets
- **websrv_poll →** serves web requests / upgrades websockets
- **prisoner_poll** → listens for & checks UDS connections from Kits
- **Asyncdns →** no good C / async APIs here.
- **coolwsd** → keeps watch & cleans-up
- **Poco** – waitQueue thread
- **Admin** → manages the admin console work … & killing misbehaving pids
- **Remotefontconfig …** → does that.

Per Document threads …

coolwsd

- **Documentbroker_001**
  - **DocumentBroker + ClientSessions + WOPI etc.**
  - ~all work in a single thread – consuming events …

ForKit

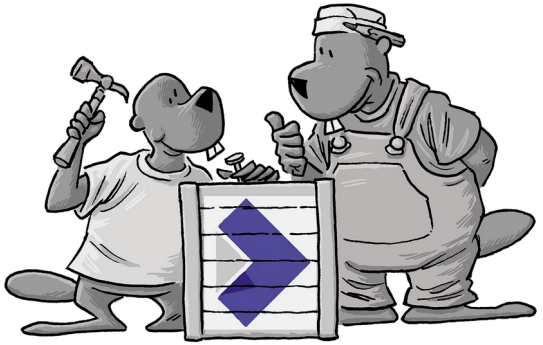
- **No (zero, none, nada) threads → [!] …**

Kit – process per document - LibreOfficeKit threads plus …

- Kitbroker_001 → ‘ImplSVMain’ thread – Unipoll …
- ThreadPool::wor → tile + sideshow compression thread pool

Collabora Online

Conclusions



Threading

- Widely and correctly considered a disaster area
- Used sparingly & carefully …
- Threading code changes need triple review here
- Never parallelize before profiling
- Locking granularity is not your friend …
- code-locking of ‘leaf’ stacks with no callbacks ‘works…’
- Pausing execution to parallelize a task – less risky.

Message passing – rocks …

Collabora Online

We are hiring please talk to me ...

Thank you & Questions ?

Collabora Online

Oh, that my words were recorded, that they were written on a scroll, that they were inscribed with an iron tool on lead, or engraved in rock for ever! I know that my Redeemer lives, and that in the end he will stand upon the earth. And though this body has been destroyed yet in my flesh I will see God, I myself will see him, with my own eyes - I and not another. How my heart yearns within me. - Job 19: 23-27