

Background (auto)-save

Interruption free saving of large documents

Michael Meeks

CEO

michael.meeks@collabora.com



Collabora
Online



LibreOffice Technology

Technical Day

COOL
days



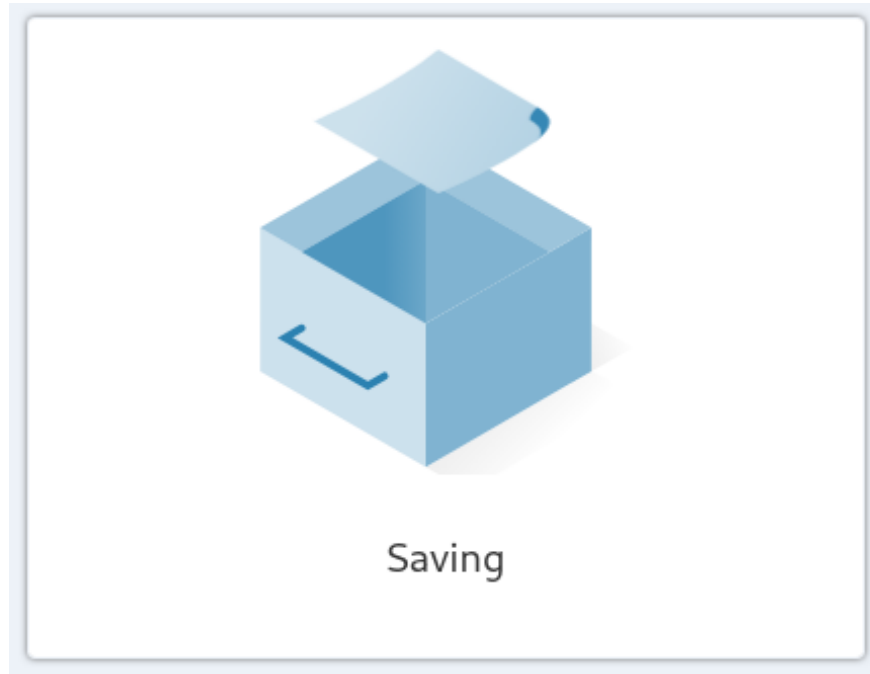
**The favourite
animated overlay:**



Beautiful saving animation

Document moving into a drawer ...

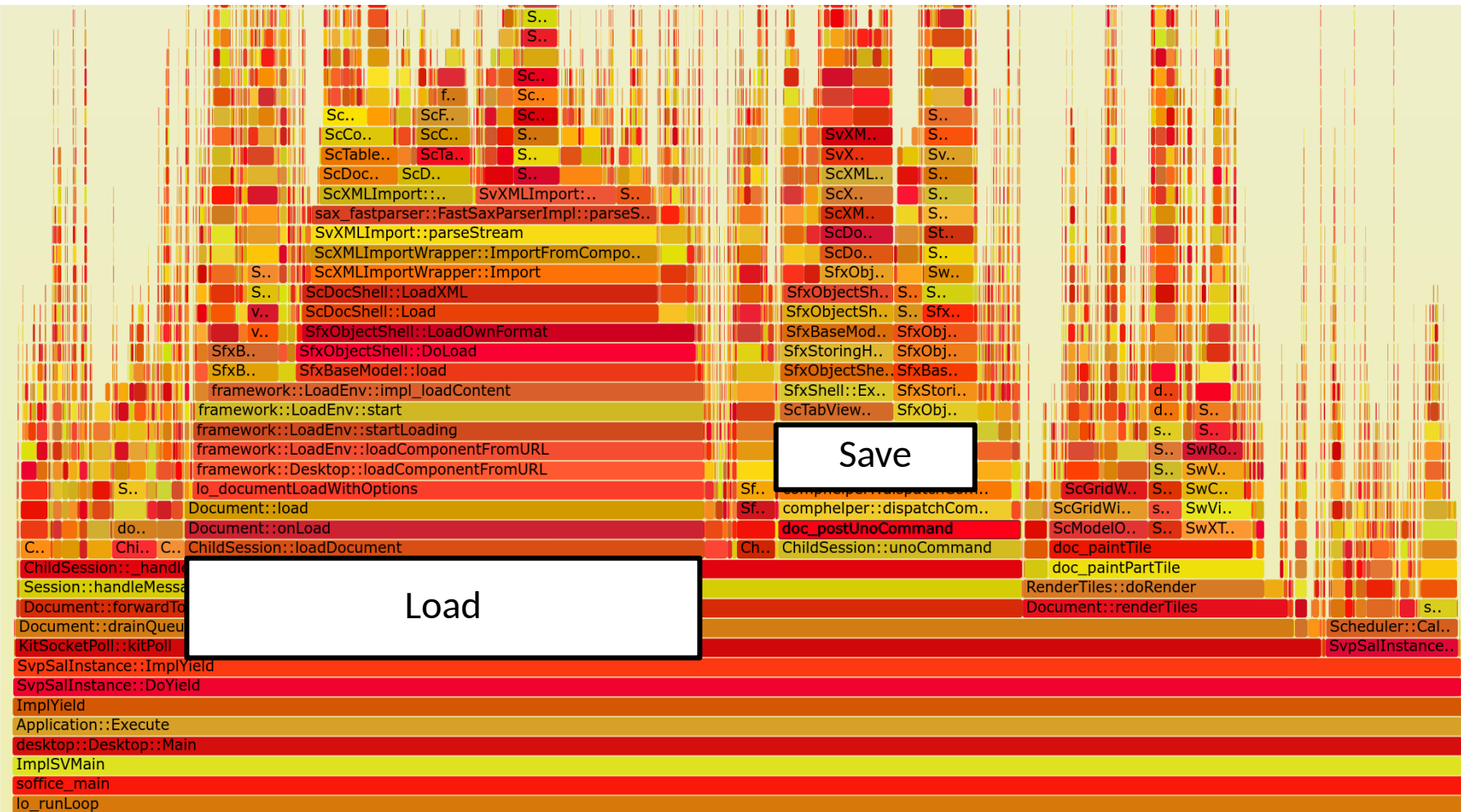
- But ...
- Blocks editing not just for you but for everyone



**Always profile
before optimizing !**



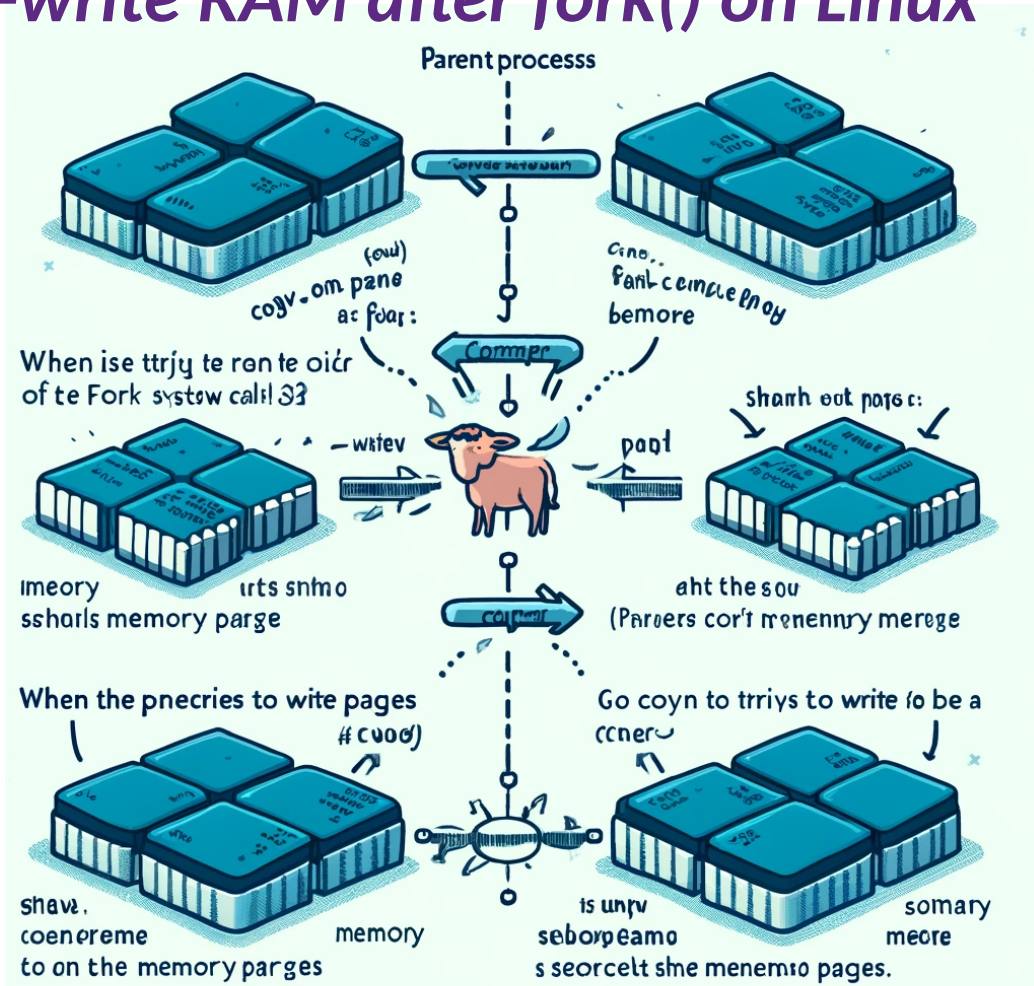
Week long profile of our usage:



**Key concept: unix fork() and
Copy on Write (CoW)**



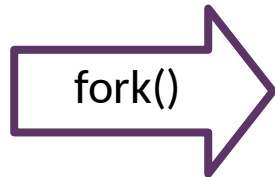
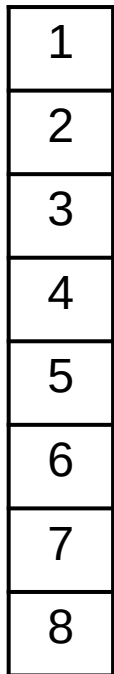
ChatGPT4 (2024-04-03): "create me a diagram illustrating copy-on-write RAM after fork() on Linux"





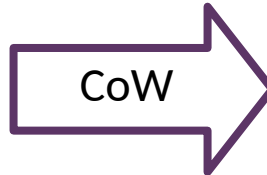
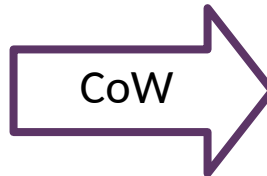
My 2 cents instead: your memory ...

Kit Process

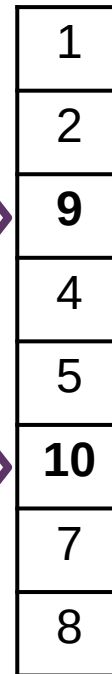


The Multiverse on your CPU ...

Saving Kit Process



Writing Kit Process



Most pages of memory are shared

Two new pages copied from page '3' and page '6' and then modified



So – just fork & save, and exit !

```
pid_t fork(void);
```

DESCRIPTION

`fork()` creates a new process by duplicating the calling process. The new process is referred to as the child process. The calling process is referred to as the parent process.

The child process is an exact duplicate of the parent process except for the following points ...

RETURN VALUE

On success, the PID of the child process is returned in the parent, and 0 is returned in the child.

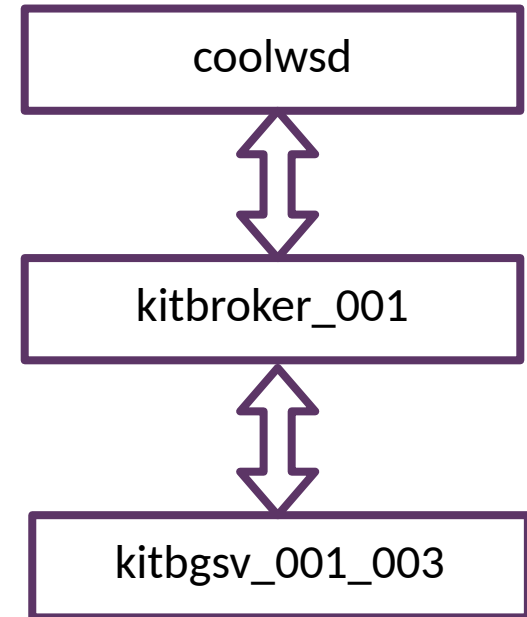


And then the detail ...

Fork issues:

- Only one thread ...
 - New 'joinThreads()' LOK method.
 - Open & readdir /proc/self/task
 - check == 1 thread before fork
- File-descriptors copied
 - Two processes writing to and reading from the same coolwsd websocket.
 - Extremely interesting display corruption.
 - close file-descriptors don't shutdown

New StreamSocket pair to talk to bgsv process:





And then more detail ...

Fork CoW's your memory but ...

- File-system:
 - Temporary files – an issue ...
 - Avoid deleting parent's files during save.
- Touching memory:
 - Don't "trim" memory in the bgsv process, ideally hold off in the parent too.
 - Hard exit the bgsave process without cleaning up / touching memory.

/proc permissions

- Can't we access own /proc/self/fd in forked process eg. parent can't.

How does it stack up ?

- 'Saving' dialog shows up for >700ms save latency.
- TRC Background save process <pid> fork **took 6ms** | kit/Kit.cpp:1449
 - **100x faster**
 - Some cost of CoW'ing pages as/when typing occurs.
- **Avoid heap fragmentation from save**

More work:



TODO bits:

- export COOL_FORCE_BGSAVE=1 to play
 - Or enable background_autosave in coolwsd.xml
- Only one ongoing bgsave
- tracking 'Modified' status reliably.
- Admin Console / tracking of memory usage
- Cleanup of malfunctioning / crashed save Kit processes
- Parameter to force bgsave rather than sync save for eg. unload.
- Extend to eg. PDF export, slide show / export
- UI thoughts:
 - What should we show while auto-saving ?



Collabora
Online



LibreOffice Technology

Technical Day

COOL
days 

Thank you!

By Michael Meeks



@CollaboraOffice
hello@collaboraoffice.com
www.collaboraoffice.com