



A fully unprivileged CernVM-FS

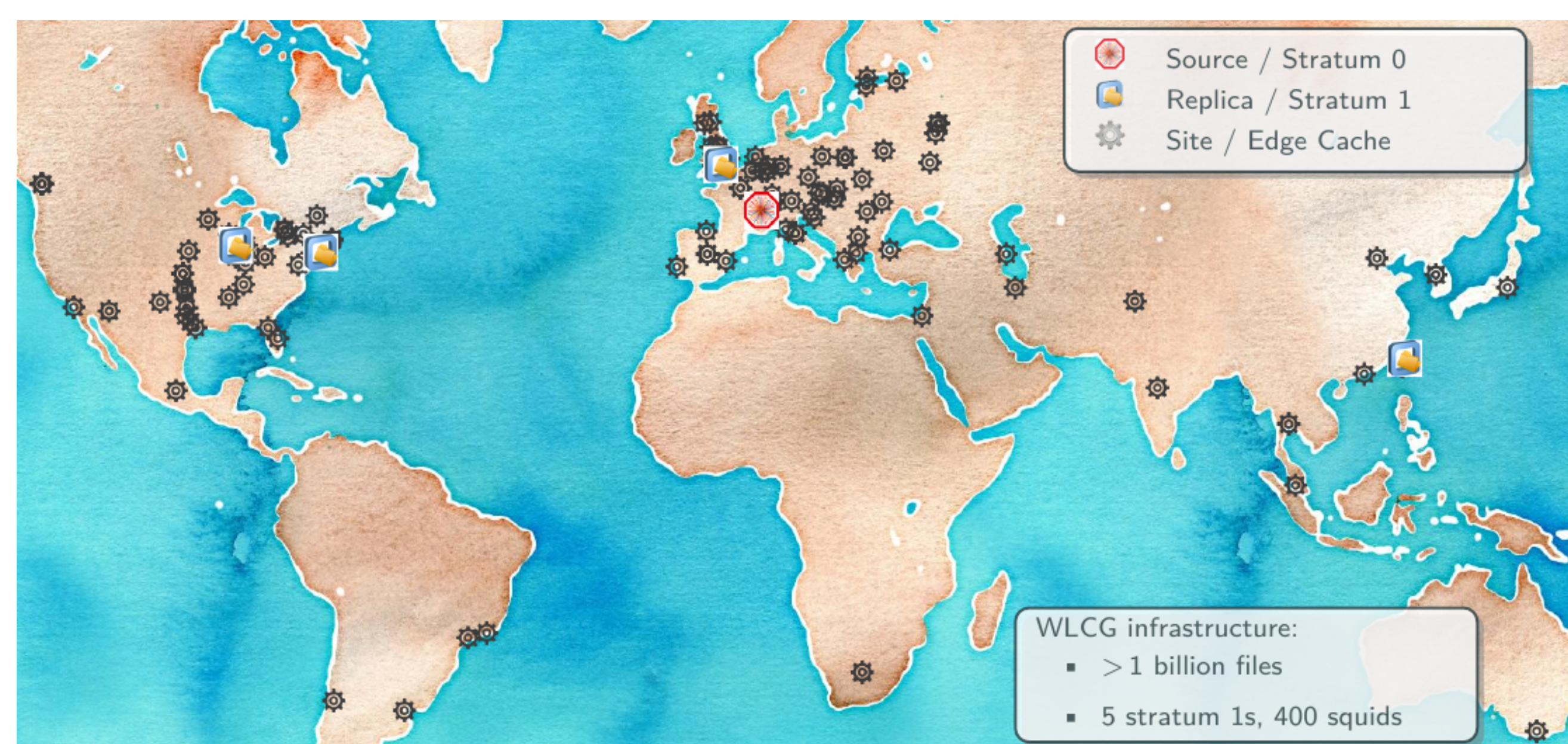
J Blomer¹, D Dykstra², G Ganis¹, S Mosciatti¹, J Priessnitz¹

¹CERN ²Fermilab

jblomer@cern.ch

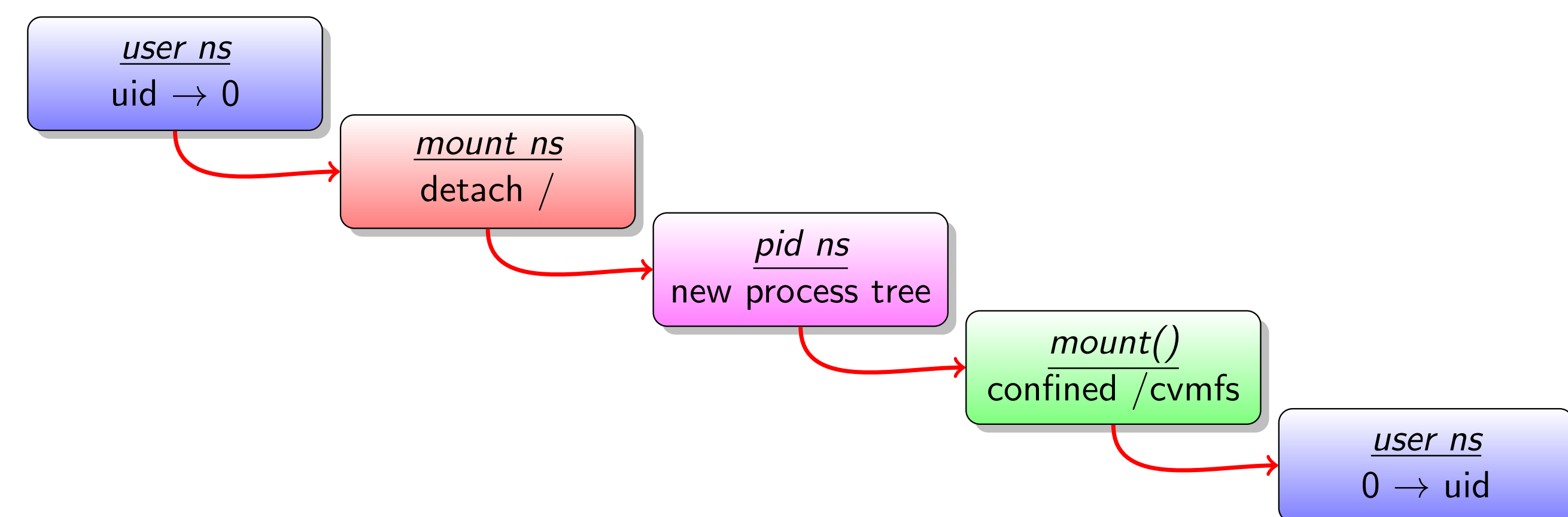
CernVM-FS – Status and Deployment

The CernVM File System provides the **software and container distribution backbone** for most High Energy and Nuclear Physics experiments [1]. Its key features include a POSIX compliant interface, HTTP transport, multi-level caching, versioning, strong consistency, and end-to-end data integrity.



New Feature: Mounts in User Namespaces

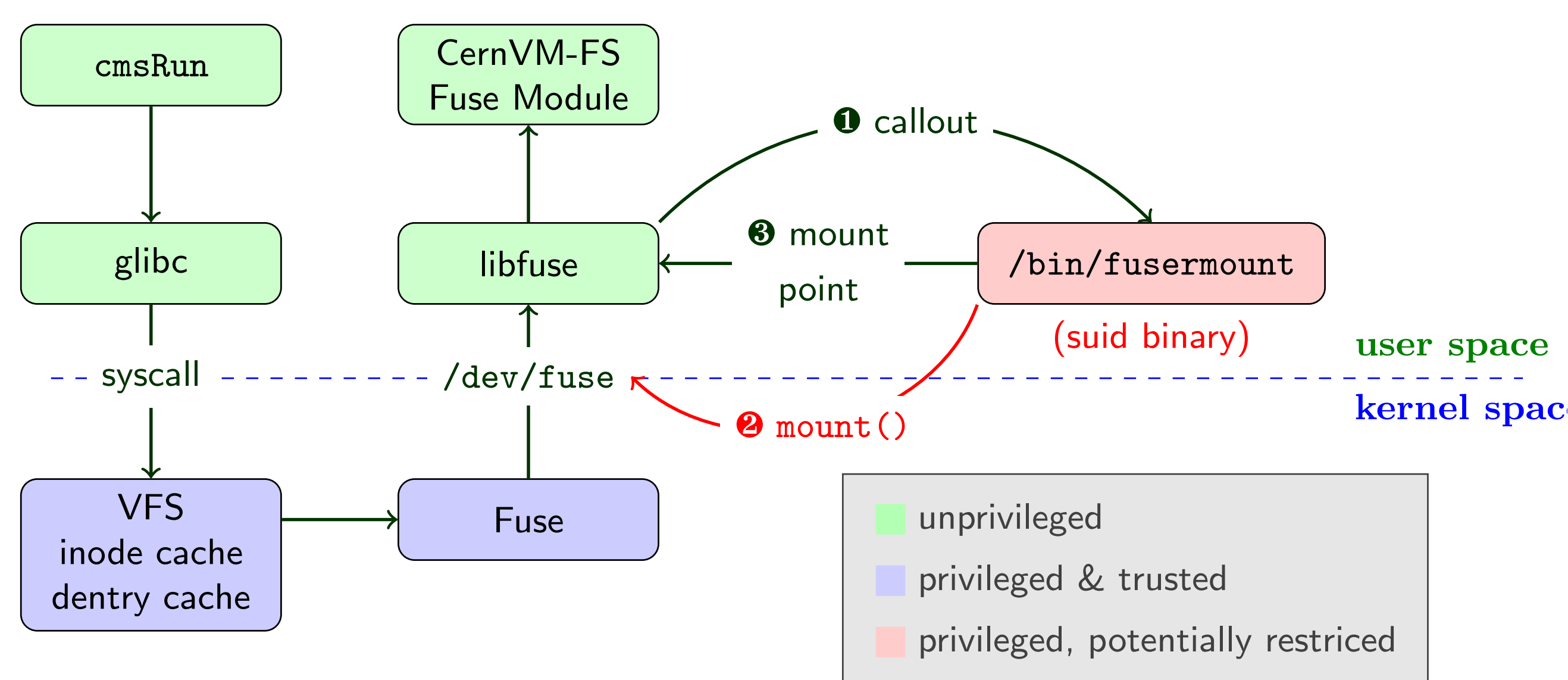
As of Linux kernel version 4.18 (e.g. EL8), FUSE mounts are **unprivileged in user name spaces**. In combination with other namespace, a CernVM-FS container environment can be spawned:



Namespace mounts enable CernVM-FS in unprivileged containers!

Privileges for File Systems in User Space

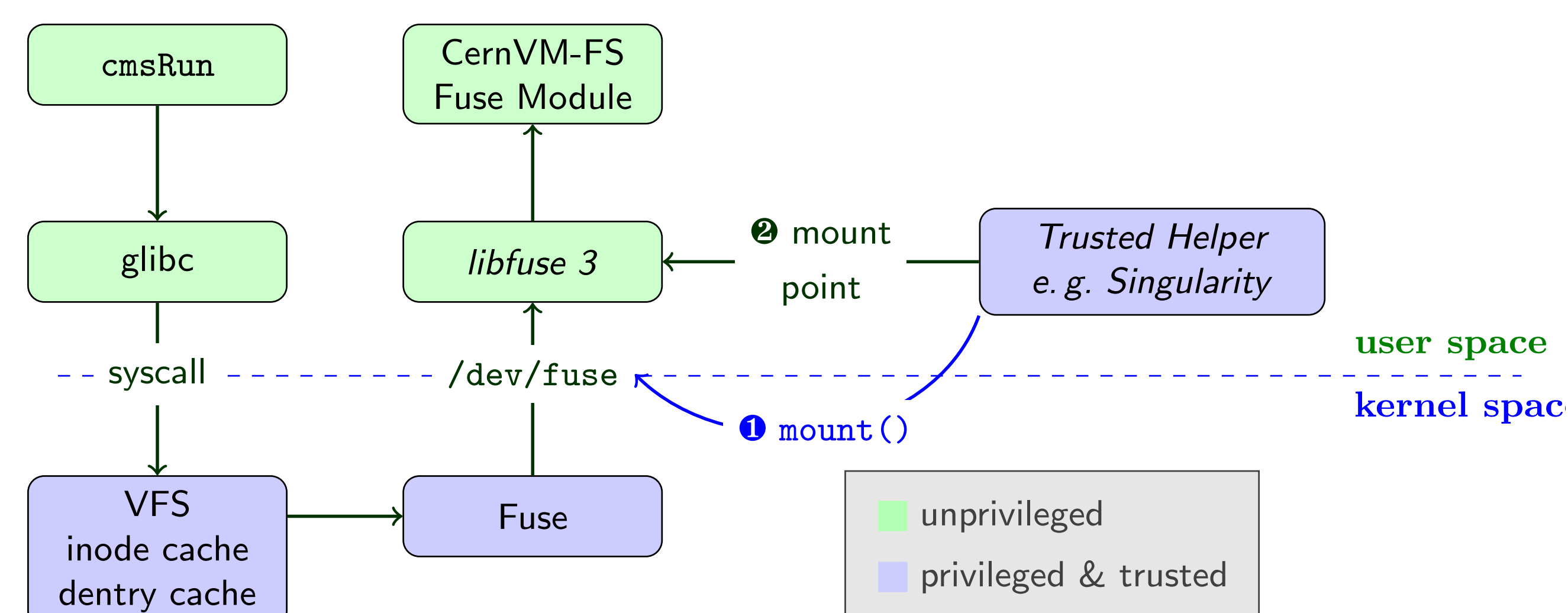
CernVM-FS is implemented as a **file system in user-space (FUSE)** [2], which permits its execution without any elevated privileges. Yet, mounting the file system in the first place is handled by a privileged **suid helper program** that is installed by the fuse package on most systems.



A successful fuse mount returns a file descriptor to /dev/fuse, which is subsequently used by the fully unprivileged **FUSE module**.

New Feature: Pre-mounted File System

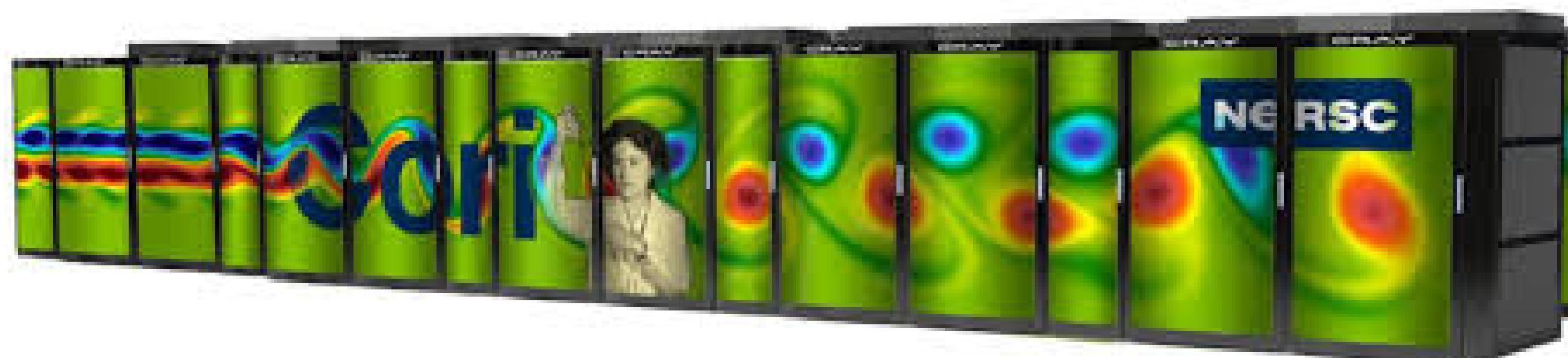
With the new FUSE3 libraries, the task of mounting /dev/fuse can be handed to a trusted, external helper. Support for mounting /dev/fuse has been added to Singularity, which runs as a trusted process on many supercomputers. Fuse 3 support has been added to CernVM-FS. FUSE3 libraries have been backported to EL6 and EL7 platforms.



Pre-mounting is implemented in **Singularity 3.4** and **CernVM-FS 2.7 (tagged)**!

On-Demand Mounts on Opportunistic Resources

The privileged nature of the **mount** system call is a **serious hindrance to running CernVM-FS on opportunistic resource and supercomputers**. While the fuse kernel module is a standard Linux facility, the execution of **suid binaries** is forbidden at some of the biggest supercomputers. Likewise, **suid binaries** are usually not available in containers.



Application 1: "Universal Pilot"

With unprivileged /cvmfs mounts, the CernVM-FS client can be bundled to provide a "universal pilot". The self-extracting bundle would

1. create a new user namespace
2. mount experiment /cvmfs repository
3. run the experiment pilot from /cvmfs
4. pilot optionally mounts additional /cvmfs repositories
5. pilot runs singularity from /cvmfs
6. singularity runs user payload job

Unprivileged mounts are available through the **cvmfsexec** package! [4]

Application 2: On-Demand Publishing

FUSE namespace mounts in concert with **fuse-overlayfs** [3] can provide an on-demand CernVM-FS publisher container:

```
λ cvmfs enter hsf.cvmfs.io /users/joe
...opens a shell in an ephemeral container
with write access to the repository
λ cvmfs publish
...back to read-only mode
```

References

[1] *Towards a serverless CernVM-FS*, EPJ Web Conf 214 (2019)

[2] *libfuse*, <https://github.com/libfuse/libfuse>

[3] *fuse-overlayfs*, <https://github.com/containers/fuse-overlayfs>

[4] *cvmfsexec*, <https://github.com/cvmfs-contrib/cvmfsexec>

