

Compiled Once. Published Once. Mounted Everywhere.

CernVM-FS, the Compute Canada model, and EESSI - how scientific software actually reaches the machines that run it

ndexr.io

2026-08-08

Part I - The problem nobody funds

Enterprise Linux is old on purpose

Your cluster runs an enterprise distribution because the vendor supports the interconnect, the parallel filesystem and the GPU driver against **that kernel and no other**. The cost is a frozen userland.

Distribution	Kernel	GCC	glibc	Python
RHEL / Rocky 8	4.18	8.4	2.28	3.9.2
RHEL / Rocky 9	5.14	11.2.1	2.34	3.9.10
Fedora 41	6.11	14.2.1	2.40	3.13.0

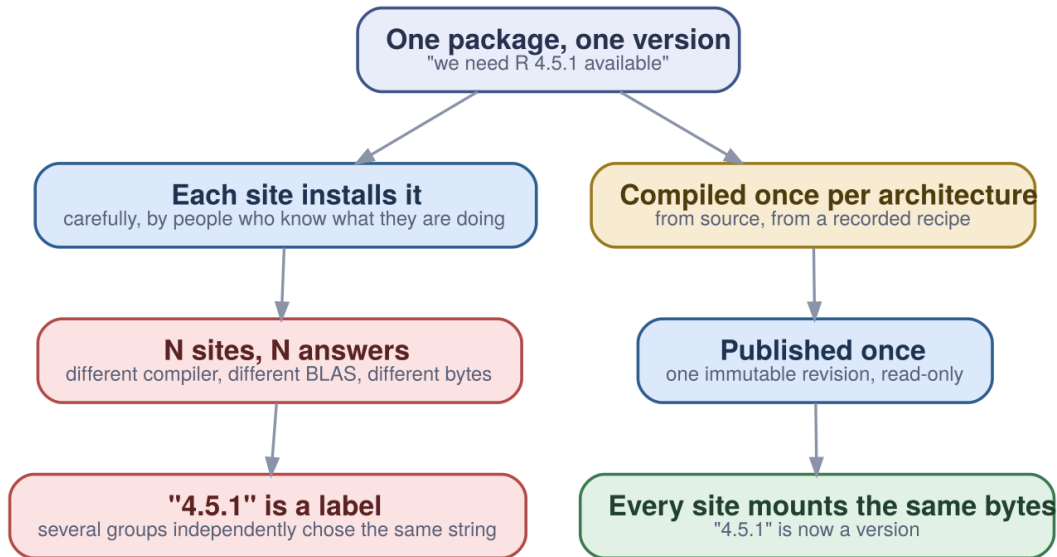
The first two rows are what you run. The third is what your users' documentation assumes.

So everybody installs everything, everywhere

- ▶ **The user's move fails** - `sudo dnf install ...` on a login node, and the sudoers lecture
- ▶ **The administrator's move does not scale** - one modulefile, then another version, then a second compiler, then a third MPI. All written by hand.
- ▶ **And the site next door is doing it too** - same packages, same sources, different results

The effort is real, the people are good, and almost all of it is duplicated.

The axis is not build-or-mount



It is **how many times**. The tree gets built either way.

The four requirements, written before the tools

Users should be presented with an interface that is as consistent and as easy to use as possible across all sites. It should also offer optimal performance.

— Digital Research Alliance of Canada

1. A way to **distribute** it → CernVM-FS
2. Independence from the **host OS** → a compatibility layer
3. **Automated installation** - humans are not consistent → EasyBuild
4. An interface that survives **ten thousand combinations** → Lmod, hierarchical

“Isn’t this what containers are for?”

Two things the word *reproducible* is doing at once:

- ▶ **Repeatability** - running the same bits again. Containers are excellent at this.
- ▶ **Accountability** - saying what those bits *are*. A container cannot touch it.

A container is a distribution format, not a derivation. Freezing an unknown does not make it known.

And, separately, the practical failure

- ▶ **The image grows with the catalogue** - the unit of change is the whole image, not the package
- ▶ **You ship what might be used, not what is** - importing one Python library touches ~3,500 files out of hundreds of thousands

Containers stay in the picture. They carry an OS and a **mount point**, not a stack.

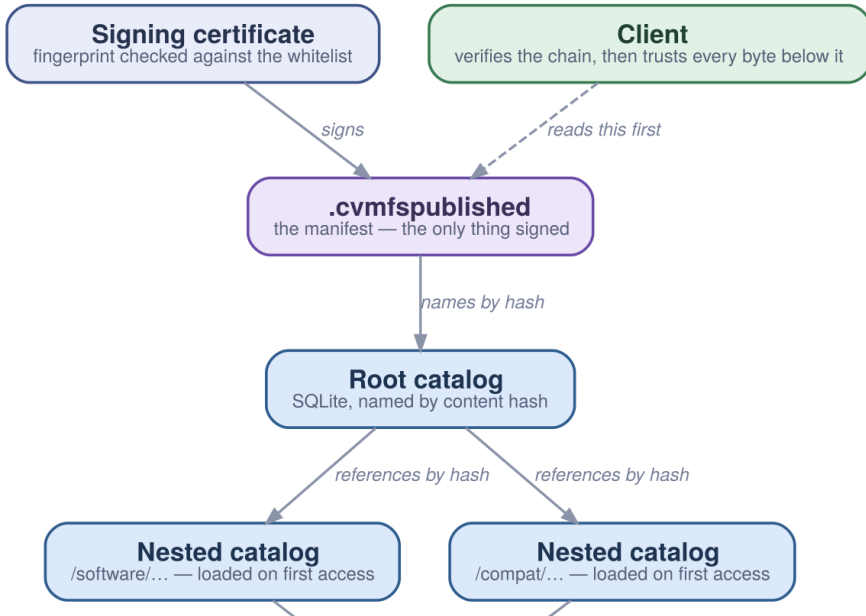
Part II - The delivery substrate

CernVM-FS: software, streamed

A read-only directory under `/cvmfs` that behaves like an on-demand streaming service. Nothing installed, nothing downloaded up front - files arrive when a process opens them.

- ▶ **A filesystem, not a package manager** - FUSE, so every tool that opens a file can use it
- ▶ **Content-addressed** - identical bytes stored once, everywhere
- ▶ **Compressed on the server**, transparent on the client
- ▶ **Ordinary HTTP** - the cache is Squid, the mirror is Apache

Integrity: one signature, covering everything



The manifest is the entire entry point

```
C7d1e...      # root catalog, by content hash
B1179648      # its size in bytes
Xf3a9...      # hash of the signing certificate
T1747526400    # when this revision was published
D240          # how long a client may cache it
S5348         # revision number
Nsoftware.eessi.io # fully qualified repository name
--           # everything below is the signature
```

The transport does not need to be trusted. A proxy, a mirror or a disk can be wrong without the client being wrong.

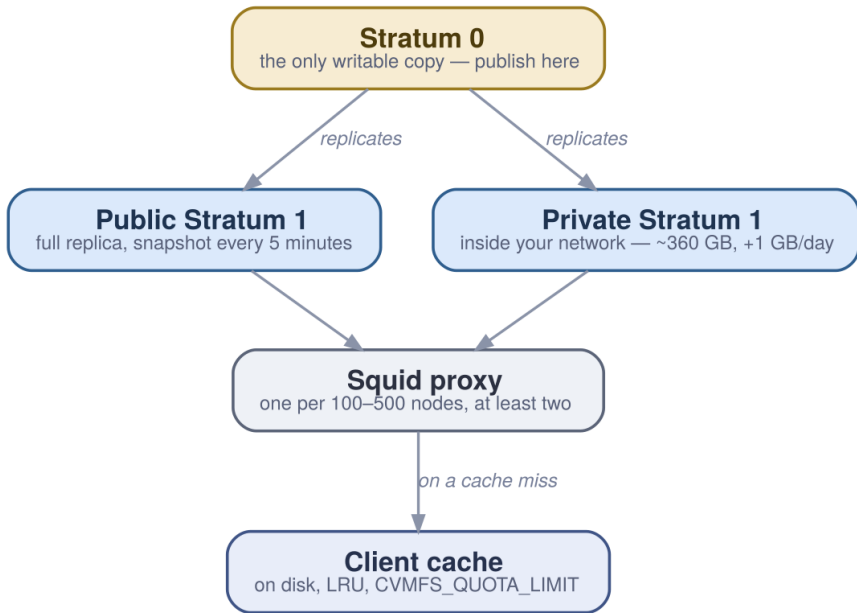
Publishing is a transaction

```
cvmfs_server transaction  software.example.org
#   ... install ...
cvmfs_server publish -a r-4.5.1 -m 'R 4.5.1, foss/2023a' software.example.org

cvmfs_server abort          software.example.org    # throws it all away
cvmfs_server rollback -t   r-4.5.1                # go back to a named revision
```

Every publish is an immutable revision. There is no half-published state for a client to catch.

Four tiers, and what each one is for



Sizing, in one slide

- ▶ **Stratum 0** - the only writable copy. Small, controlled, not load-bearing for readers.
- ▶ **Stratum 1** - full replica. Private one 360 GB for EESSI, +1 GB/day. Your disconnect insurance.
- ▶ **Proxy** - one per **100-500 nodes**, at least two. This is what makes MPI start-up bearable.
- ▶ **Client cache** - on the SSD, big enough. Where almost every read is answered.

The entire client configuration

```
# /etc/cvmfs/default.local
CVMFS_HTTP_PROXY="http://proxy-a:3128|http://proxy-b:3128"
CVMFS_QUOTA_LIMIT=10000          # 10 GB of local cache
CVMFS_CACHE_BASE=/ssd/cvmfs      # the fastest disk the node has

sudo cvmfs_config reload
ls /cvmfs/software.eessi.io      # mounted on demand by autofs
```


What it scales to (CERN, May 2026)

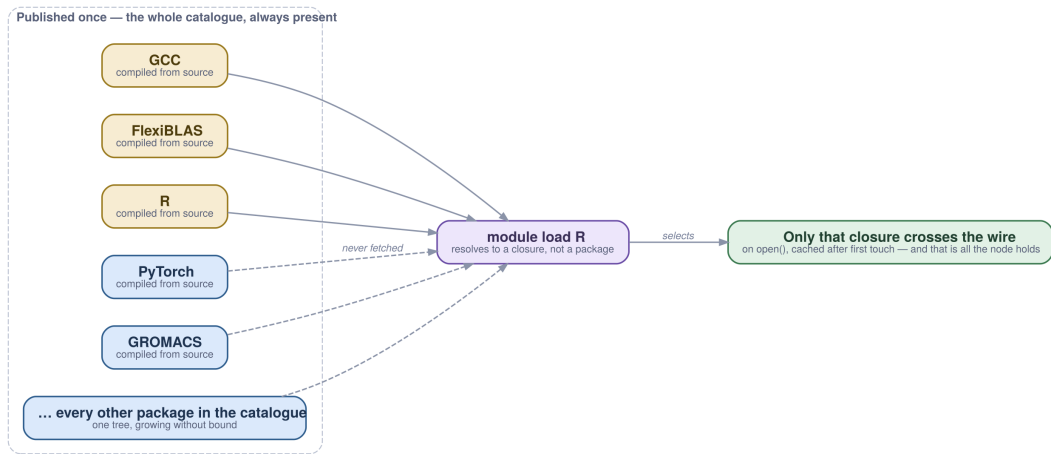
- ▶ ~**15** Stratum 1 mirror servers
- ▶ > **33 billion** files in the /cvmfs tree
- ▶ ~**2 PB** accessible - proven to 100 PB
- ▶ ~**290** repositories, including 8,000+ container images

It is faster than the filesystem you already own

Importing one Python package touches ~3,500 files / ~1.1 GB, most of them tiny.

Where the software lives	Time to import
Local SSD or ramdisk	~2 s (the floor)
GPFS / Lustre	30-60 s , even with a hot pagepool
CernVM-FS, cold cache, near Stratum 1 + proxy	dramatically better
CernVM-FS, warm client cache	approx. local disk

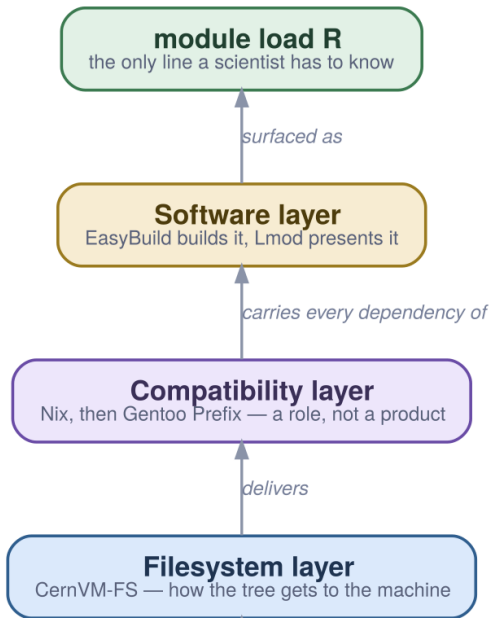
Why: only the closure crosses the wire



The catalogue grows without bound because nothing pays for the parts it does not open.

Part III - One architecture, two implementations

Four layers



...and the middle one is a role, not a product

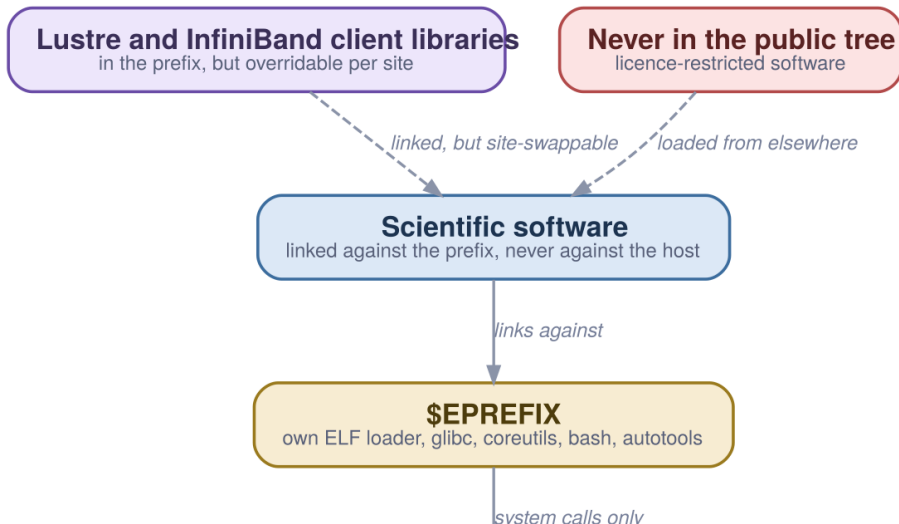
- ▶ Canada filled the compatibility slot with **Nix**
- ▶ Canada then replaced it with **Gentoo Prefix**
- ▶ EESSI chose **Gentoo Prefix** from the start

Three configurations, two programmes, one working architecture each time - and the layers above and below did not change when it was swapped.

This converts a technology bet into an architectural choice.

The compatibility layer

A full OS userland installed somewhere other than / - its own loader, glibc, coreutils, bash, autotools - built from source, without root.



Why Nix was replaced, in their words

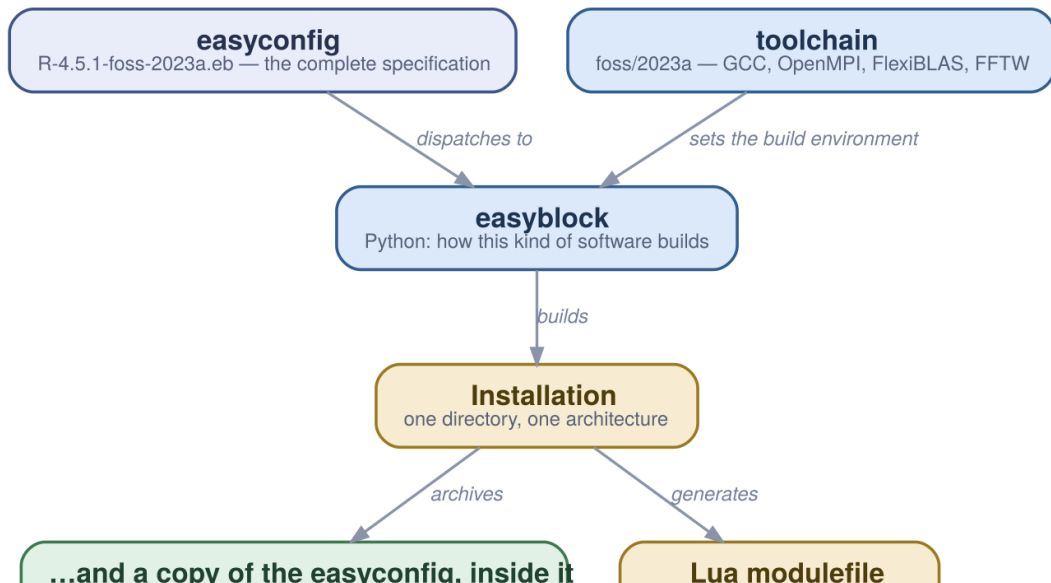
- ▶ **Store hashes leaked upward** through CMake, qmake and Python virtualenvs
- ▶ **Garbage collection was destructive** - the feature, from the other side
Gentoo Prefix: no symlinks, no store leak, minimal solution.
— *Digital Research Alliance of Canada, on the 2020 migration*

What it actually costs to run a prefix

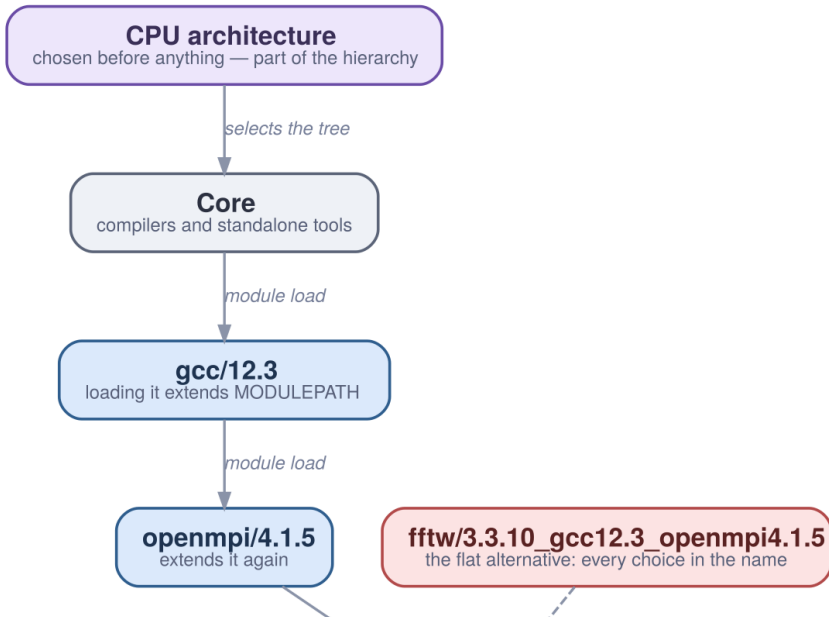
1. Downloaded binaries need `patchelf --set-interpreter`
2. A stale `LD_LIBRARY_PATH=/usr/lib64` in a `.bashrc` breaks most tools
3. Anaconda and Julia ship binaries that do not always work - hence 17,000+ wheels
4. `$EPREFIX/var` versus `/var` - `who` and `last` read an empty file
5. One central `libstdc++`, so GCCcore builds collapse to one module level

Every site hits these in the same order. That is a much better position than a set nobody has written down.

EasyBuild records it



Lmod makes it choosable



Architecture belongs in the hierarchy, not the name

```
$ module load fftw
```

Lmod has detected the following error: These module(s) exist but cannot be loaded as requested: "fftw"

```
$ module spider fftw/3.3.10
```

You will need to load all module(s) on one of the lines below first:

```
gcc/12.3  openmpi/4.1.5
```

```
$ module load gcc/12.3 openmpi/4.1.5 fftw/3.3.10
```

~1,000 applications $\times$ 4 CPU generations $\times$ 4 GPU generations = 10,000+ permutations. The user sees fftw/3.3.10.

The Canadian stack, in numbers

- ▶ 5 regional consortia, 38 institutions, ~18,000 accounts, 6 national systems
- ▶ ~**1,000** scientific applications, **10,000+** version/CPU/toolchain permutations
- ▶ **17,000+** Python wheels
- ▶ One catalogue, one scheduler, one way of installing things - since **2017**

Getting in

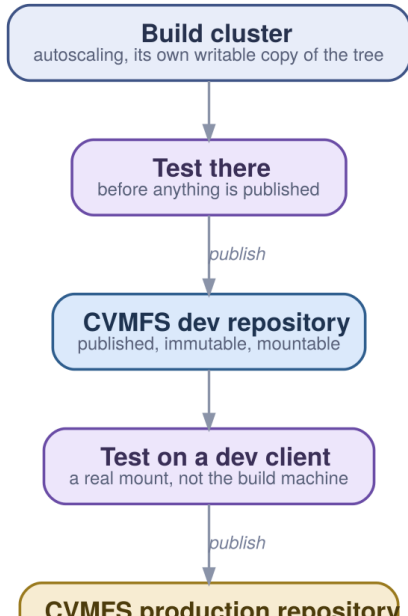
```
# the Alliance stack - public, mountable anywhere
source /cvmfs/soft.computecanada.ca/config/profile/bash.sh
module load StdEnv/2023

# EESSI - in the default CernVM-FS config since November 2023
source /cvmfs/software.eessi.io/versions/2023.06/init/bash
module load R/4.3.2-gfbf-2023a
```

Old environments stay mountable

Environment	Compiler	MPI	What changed
StdEnv/2016.4	GCC 5.4 / Intel 2016.4	OMPI 2.1.1	First unified environment
StdEnv/2018.3	GCC 7.3 / Intel 2018.3	OMPI 3.1.2	First AVX-512
StdEnv/2020	GCC 9.3 / Intel 2020.1	OMPI 4.0.3	Nix → Gentoo Prefix
StdEnv/2023	GCC 12.3 / Intel 2023.1	OMPI 4.1.5	GCC default; min AVX2; FlexiBLAS; CUDA 12

Nothing is promoted by being edited

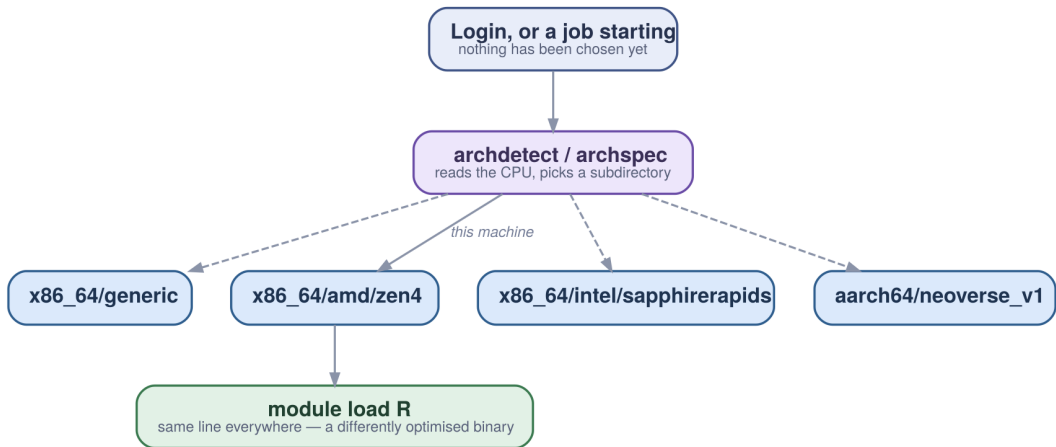


EESSI: the same architecture, new hardware

Architecture	Targets
x86_64 / AMD	zen2, zen3, zen4, zen5
x86_64 / Intel	haswell, skylake_avx512, cascadelake, icelake, sapphirerapids
aarch64	neoverse_n1, neoverse_v1, a64fx, nvidia/grace
riscv64	separate development repository

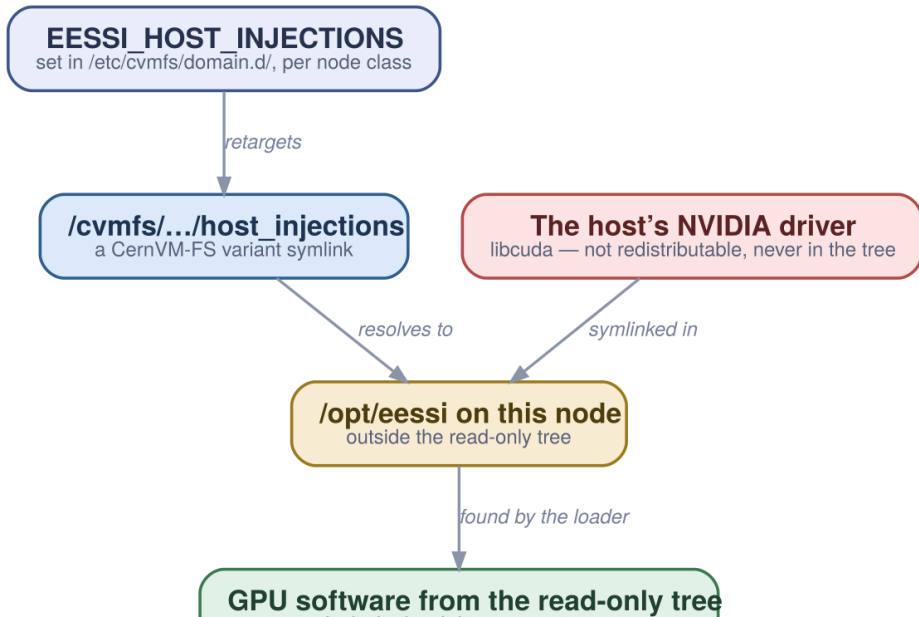
Validated with a **ReFrame** test suite a site runs against its own mount.

The tree carries all of them; the machine picks



“We moved to Arm” stops being a recompile and a revalidation.

The seam: host injections



...and the influence now runs both ways

- ▶ The Alliance's Gentoo Prefix bootstrap is based on **EESSI's** Ansible implementation
- ▶ The Alliance has adopted **EESSI's** variant-symlink approach for CUDA driver libraries

The method has transferred between programmes that share no staff, funding or hardware - and is now transferring back.

Part IV - Running it

What you actually stand up, in order

1. **Client** - a package, a public key, a config file. autofs mounts on demand.
2. **Two proxies**, once you have nodes - one per 100-500 workers
3. **A private Stratum 1**, if a lost link would stop work - one replica, one Apache, one cron job
4. **Diskless nodes** - a loopback file on the shared filesystem, sized ~15% above the cache
5. **Monitor two things** - cache cleanup frequency, and client revision versus Stratum 1

The four error messages, decoded

What it says	What it means
Too many levels of symbolic links	Mount namespace not shared. Almost always a container.
No such file or directory	A catalog could not load. The reason is in the syslog.
Transport endpoint is not connected	The client died and the watchdog did not restart it.
Software caused connection abort	The kernel cut the FUSE connection. Remount.

Five commands worth knowing first

```
cvmfs_config probe software.eessi.io      # is it reachable at all
cvmfs_config stat -v software.eessi.io    # revision, cache use, hit rate
cvmfs_config showconfig software.eessi.io # what it thinks it was told
cvmfs_talk -i software.eessi.io ncleanup24 # cache evictions in 24h
sudo cvmfs_config wipecache               # last resort, not first
```


The boundary - what this does not solve

- ▶ Updates are visible **within minutes**, not instantly
- ▶ The signing **whitelist expires** - an availability dependency on the runbook
- ▶ **Unprivileged mounting** works, but each process gets its own cache
- ▶ **EasyBuild does not sandbox** - bit-for-bit is a target, not a guarantee
- ▶ **Licensed software** does not fit the model
- ▶ **Derivation proves construction, never meaning**

The one that catches people

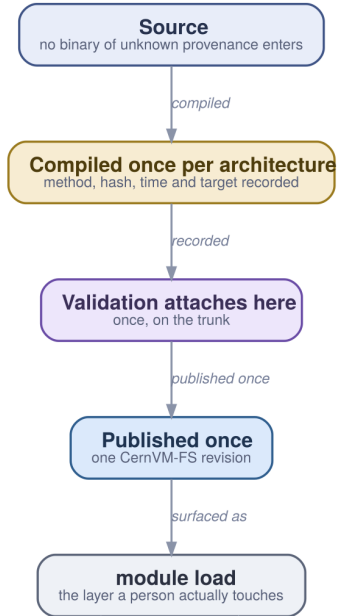
If your nodes have no local disk, do **not** run one client and export it over NFS.

It can be made to work. It is slow. It generates more operational problems than any other configuration in the deployment guidance.

Use a **loopback cache file** on the shared filesystem instead.

Part V - What it settles

Four questions that stop being open



Consequences, not benefits

- ▶ **“Which version is this?” has one answer** - same source, same compiler, same BLAS, same bytes
- ▶ **Approval attaches once, on the trunk** - a second approval path is two things to keep in agreement forever
- ▶ **Reproducing an old result is a mount, not a rebuild**
- ▶ **Changing architecture stops being a project** - Arm is the same line resolving elsewhere

Where this leaves us

None of this is novel and all of it is in production, at national scale, for around eighteen thousand researchers.

The useful conversation is not whether the model works - two national programmes have answered that - but **which parts of it fit the estate you already have.**

<https://what.ndexr.io>