



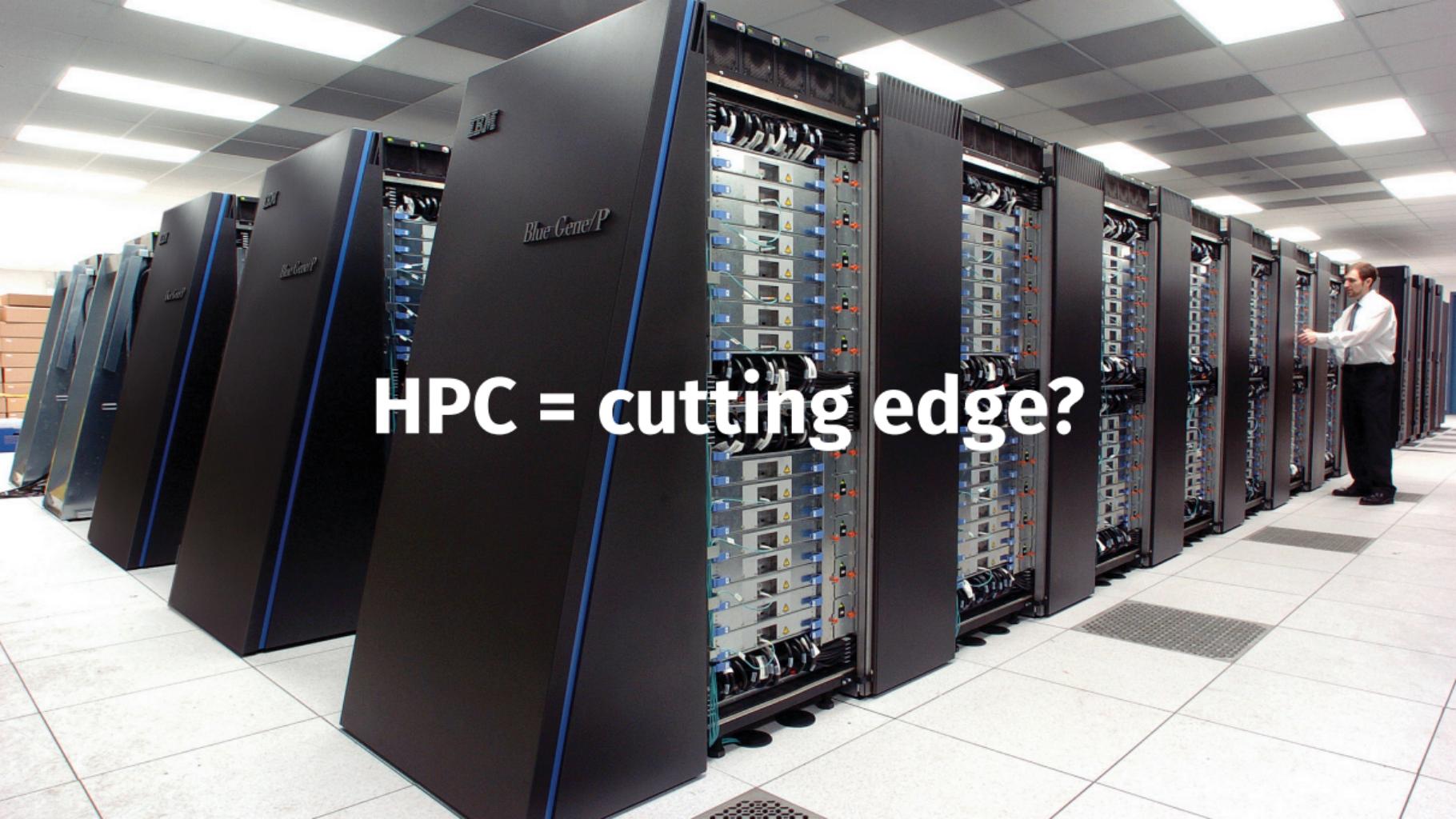
Continuous Integration & Continuous Delivery for HPC with Guix

Ludovic Courtès

CASTIEL Webinar

27 September 2024

Inria



HPC = cutting edge?

Here is an example of loading a module on a Linux machine under bash.

```
% module load gcc/3.1.1  
% which gcc  
/usr/local/gcc/3.1.1/linux/bin/gcc
```

Now we'll switch to a different version of the module

```
% module switch gcc gcc/3.2.0  
% which gcc  
/usr/local/gcc/3.2.0/linux/bin/gcc
```



The Conda logo features the word "CONDA" in large, bold, green letters. The letter "C" has a white diamond pattern on its left side.



① 208 Open ✓ 308 Closed

Author ▾

Labels ▾

Projects ▾

Milestones ▾

Assignee ▾

Sort ▾

① Installation issue: xfd build-error

#11526 opened 18 hours ago by huqy

① Installation issue: openmpi (any version) on mac build-error

#11515 opened 4 days ago by luca-heltai

① Could not install elfutils build-error

#11501 opened 5 days ago by jczech07

① Installation issue: mumps (serial), error "/bin/sh: line 0: fc: -h: invalid option"

build-error

#11498 opened 5 days ago by samfux84

① Spack points to incorrect cray-libsci in LANL environment build-error

4

#11491 opened 6 days ago by floquet

① Installation issue: range-v3 build-error



2

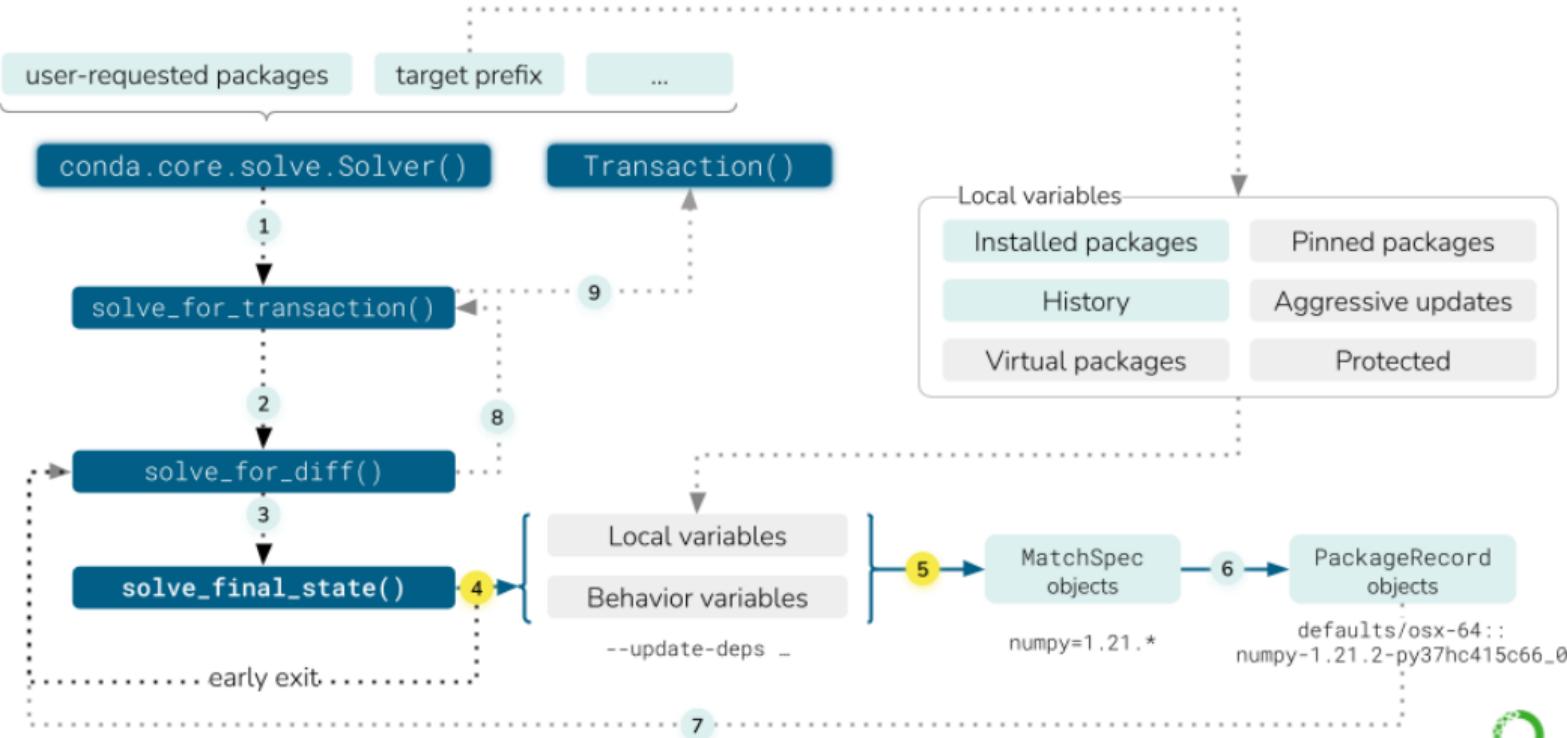
#11481 opened 6 days ago by chissg

① Installation issue: boost build-error

1

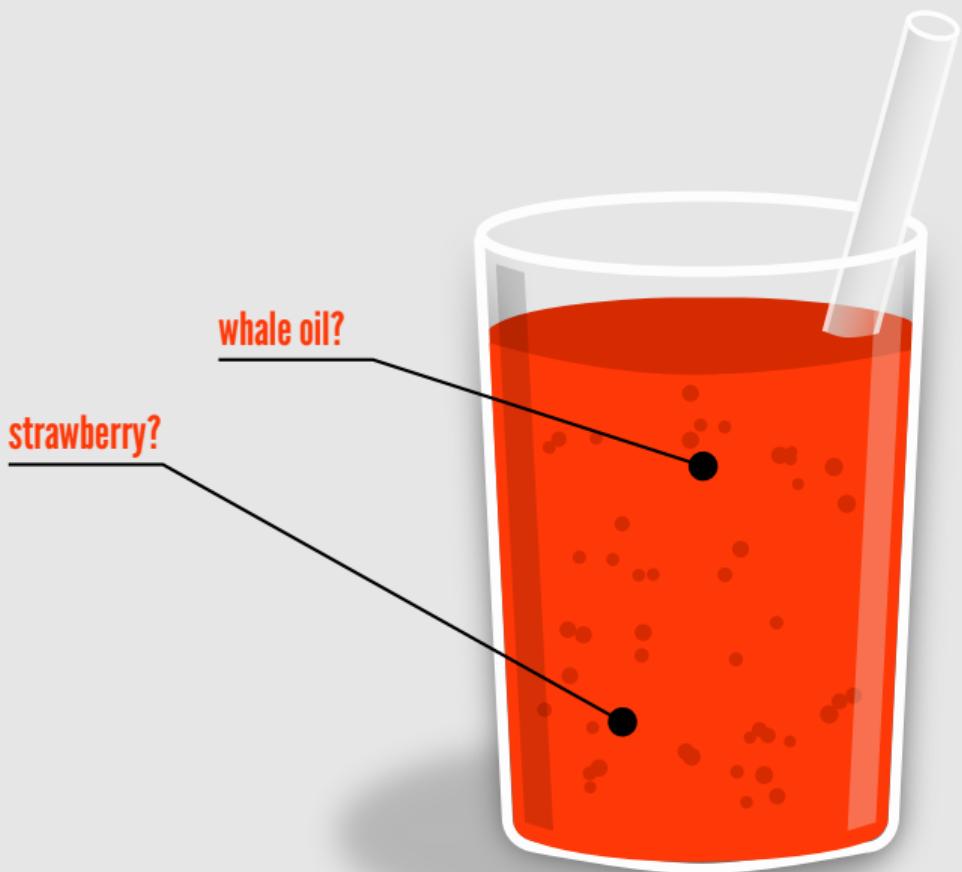
#11467 opened 7 days ago by abc19899

Inside the Solver: formulating the problem



Containers to the rescue?





Containers lack transparency

courtesy of Ricardo Wurmus

```
Bootstrap: library
From: ubuntu:18.04

%setup
    touch /file1
    touch ${SINGULARITY_ROOTFS}/file2

%files
    /file1
    /file1 /opt

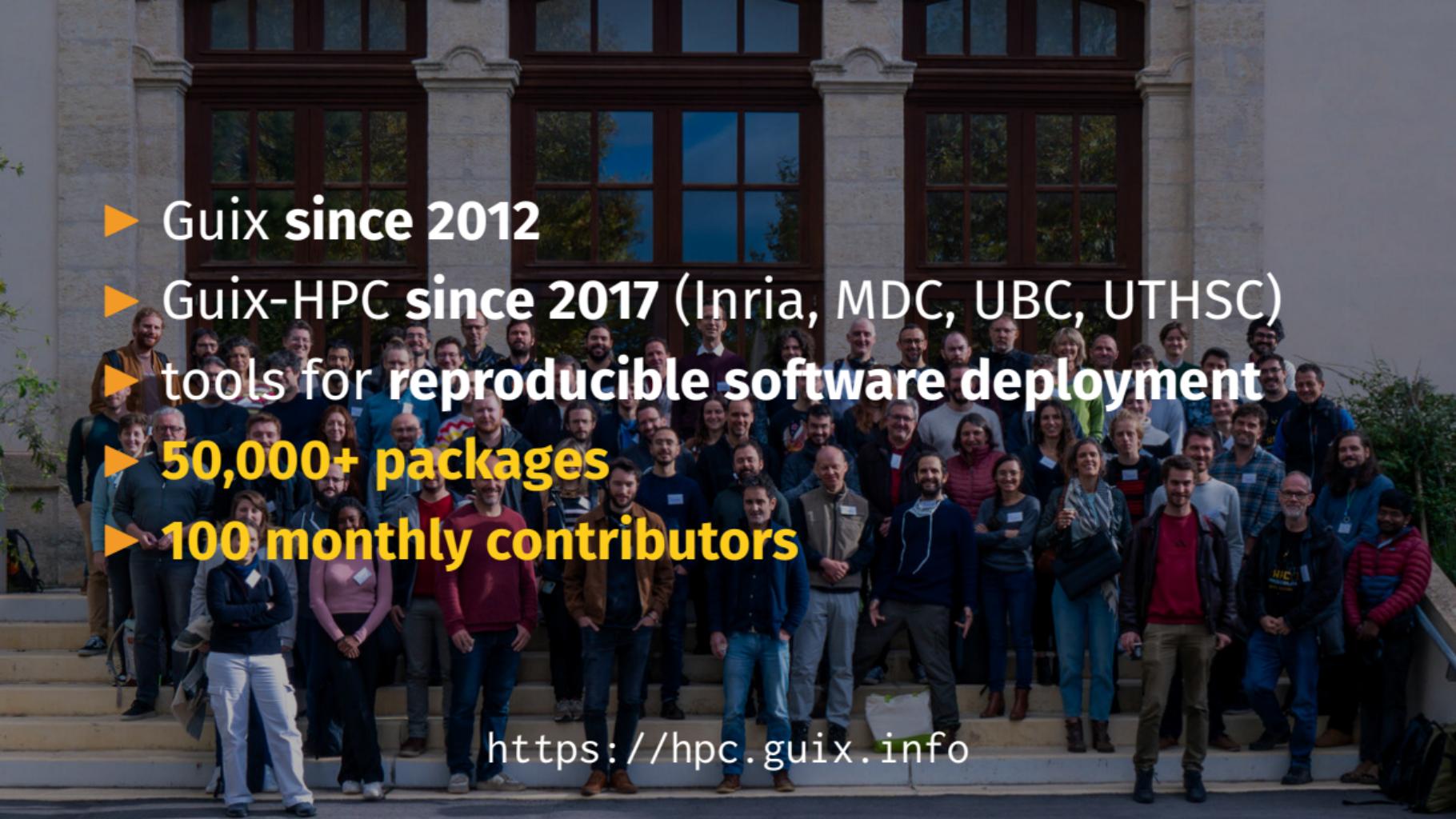
%environment
    export LISTEN_PORT=12345
    export LC_ALL=C

%post
    apt-get update && apt-get install -y netcat
    NOW=`date`
    echo "export NOW=\"${NOW}\"" >> $SINGULARITY_ENVIRONMENT

%runscript
    echo "Container was created $NOW"
    echo "Arguments received: $*"
    exec echo "$@"
```



<https://hpc.guix.info>

- 
- ▶ Guix since 2012
 - ▶ Guix-HPC since 2017 (Inria, MDC, UBC, UTHSC)
 - ▶ tools for **reproducible software deployment**
 - ▶ **50,000+ packages**
 - ▶ **100 monthly contributors**

<https://hpc.guix.info>

- ▶ **PlaFRIM** (FR): Inria Bordeaux (3,000+ cores)
- ▶ **GriCAD** (FR): Grenoble (1,000+ cores)
- ▶ **GLICID** (FR): Nantes (4,000+ cores)
- ▶ **Grid'5000** (FR): 8 sites (12,000+ cores)
- ▶ **Max Delbrück Center** (DE): 250-node cluster + workstations
- ▶ **UMC Utrecht** (NL): 68-node cluster (1,000+ cores)
- ▶ ...

- ▶ **PlaFRIM** (FR): Inria Bordeaux (3,000+ cores)
- ▶ **GriCAD** (FR): Grenoble (1,000+ cores)
- ▶ **GLICID** (FR): Nantes (4,000+ cores)
- ▶ **Grid'5000** (FR): 8 sites (12,000+ cores)
- ▶ **Max Delbrück Center** (DE): 250-node cluster + workstations
- ▶ **UMC Utrecht** (NL): 68-node cluster (1,000+ cores)
- ▶ ...

```
guix install gcc-toolchain openmpi hwloc
```

```
source ~/.guix-profile/etc/profile
```

```
guix package --roll-back
```

```
guix shell python python-numpy
```

```
guix shell python python-numpy \  
-- python3 -c 'import numpy'
```

```
guix shell -D petsc git
```

```
guix shell -D petsc git --container
```

```
guix shell --manifest=manifest.scm -C
```

```
(specifications->manifest
'("gcc-toolchain" "coreutils" "grep" "sed"
"openmpi" "openblas" "maphys++"))
```

```
$ guix shell -D petsc git \  
--export-manifest
```

```
$ guix shell -D petsc git \  
--export-manifest
```

; ; What follows is a "manifest" equivalent
; ; to the command line you gave.

```
(concatenate-manifests  
  (list (specifications->manifest (list "git"))  
        (package->development-manifest  
         (specification->package "petsc"))))
```

```
bob@laptop$ guix shell --manifest=manifest.scm
bob@laptop$ guix describe
guix cabba9e
repository URL: https://git.sv.gnu.org/git/guix.git
commit: cabba9e15900d20927c1f69c6c87d7d2a62040fe
```

```
bob@laptop$ guix shell --manifest=manifest.scm
bob@laptop$ guix describe
guix cabba9e
repository URL: https://git.savannah.gnu.org/git/guix.git
commit: cabba9e15900d20927c1f69c6c87d7d2a62040fe
```

```
alice@supercomp$ guix pull --commit=cabba9e
alice@supercomp$ guix shell --manifest=manifest.scm
```



travel in space and time!

Redeploy environments: 2 files, 2 commands

1. guix describe -f channels > **channels.scm**
2. guix time-machine -C **channels.scm** -- \
shell -m **manifest.scm**

Interoperability.

```
$ guix pack \  
      python python-numpy python-scipy  
...  
/gnu/store/...-pack.tar.gz
```

```
$ guix pack --relocatable \
    python python-numpy python-scipy
...
/gnu/store/...-pack.tar.gz
```

<https://hpc.guix.info/blog/2020/05/faster-relocatable-packs-with-fakechroot/>

```
$ guix pack --format=squashfs \
    python python-numpy python-scipy
...
/gnu/store/...-singularity-image.tar.gz
```

```
$ guix pack --format=docker \  
  python python-numpy python-scipy  
...  
/gnu/store/...-docker-image.tar.gz
```



ENVIRONMENT
MODULES

```
guix module create -o /opt/modules \  
gcc-toolchain openmpi netcdf gromacs
```

<https://hpc.guix.info/blog/2022/05/back-to-the-future-modules-for-guix-packages/>

```
guix module create -o /opt/modules \  
--tune=skylake-avx512 \  
gcc-toolchain openmpi netcdf gromacs
```

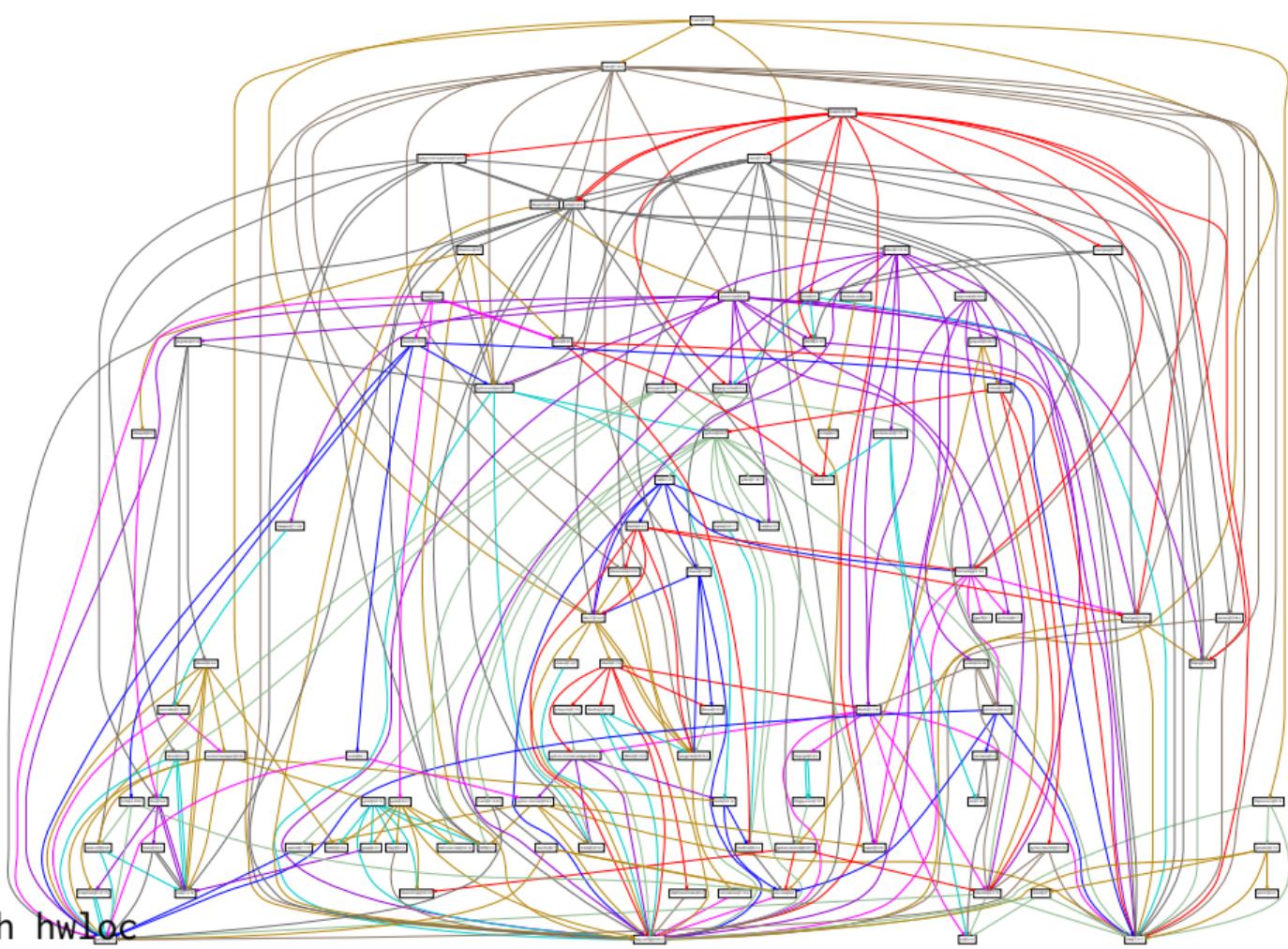
[https://hpc.guix.info/blog/2022/05/back-to-the-
future-modules-for-guix-packages/](https://hpc.guix.info/blog/2022/05/back-to-the-future-modules-for-guix-packages/)

```
guix module create -o /opt/modules \  
--manifest=manifest.scm
```

[https://hpc.guix.info/blog/2022/05/back-to-the-
future-modules-for-guix-packages/](https://hpc.guix.info/blog/2022/05/back-to-the-future-modules-for-guix-packages/)

**Reproducible deployment as a
foundation for robust CI/CD.**

guix graph hwloc



```
$ guix build hwloc
```

isolated build: chroot, separate name spaces, etc.

```
$ guix build hwloc  
/gnu/store/ h2g4sf72... -hwloc-1.11.2
```



hash of **all** the dependencies

```
$ guix build hwloc  
/gnu/store/ h2g4sf72... -hwloc-1.11.2  
  
$ guix gc --references /gnu/store/...-hwloc-1.11.2  
/gnu/store/...-glibc-2.33  
/gnu/store/...-gcc-10.3.0-lib  
/gnu/store/...-hwloc-2.9.0
```

```
$ guix build hwloc
```

```
/gnu/store/ h2g4sf72... -hwloc-1.11.2
```

```
$ guix gc --references /gnu/store/...-hwloc-1.11.2
```

```
/gnu/store/...-glibc-2.33
```

```
/gnu/store/...-gcc-10.3.0-lib
```

```
/gnu/store/...-hwloc-1.11.2
```

bit-identical for everyone*

build processes
chroot, separate UIDs

client commands

guix build hwloc

build daemon

build processes
chroot, separate UIDs

client commands

`guix build hwloc`

build daemon

RPCS



build processes

chroot, separate UIDs

CMake, GCC, etc.

CMake, GCC, etc.

CMake, GCC, etc.

client commands

guix build hwloc

build daemon

RPCS

```
(define python  
  (package ...))
```

test

```
guix build python  
/gnu/store/...-python-3.9.6
```

git push

Git repository

```
(define python  
  (package ...))
```

test

```
guix build python  
/gnu/store/...-python-3.9.6
```

git push

user

guix pull

Git repository

```
(define python  
  (package ...))
```

test

```
guix build python  
/gnu/store/...-python-3.9.6
```

git push

Git repository

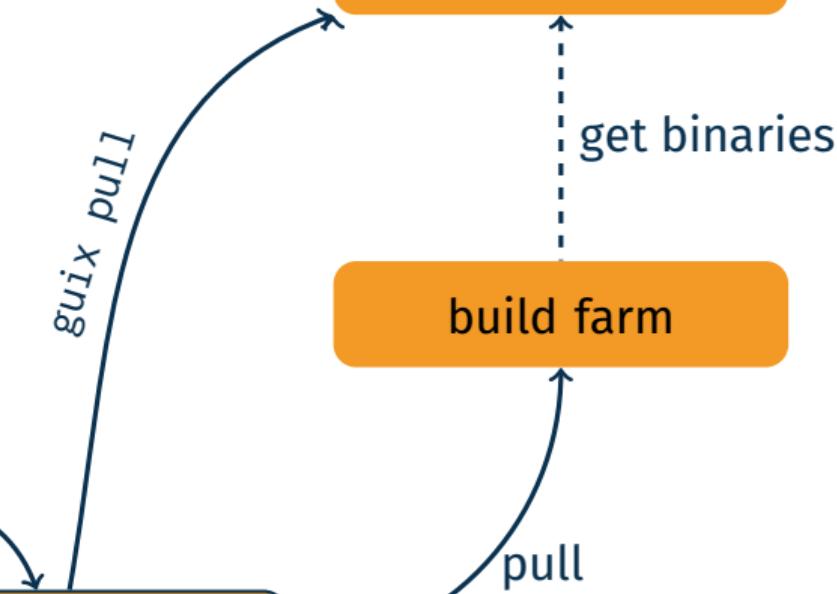
guix pull

build farm

user

get binaries

pull



Channel

Commit

<https://gitlab.inria.fr/guix-hpc/guix-hpc.git>

[f002aad7b66cd215b6a1720c48f08ed73f2661d5](#)

↑ <https://git.savannah.gnu.org/git/guix.git>

[7b0f145802f0c2c785014293d748721678fef824](#)

Builds

All 36

Scheduled 0

Succeeded 32

Failed 4

ID	Completion time	Job	Name	System
✓ ● 714970	1 Jun 18:19 +0200	rochpl.x86_64-linux	rochpl-5.7.1	x86_64-linux
✓ ● 714967	1 Jun 18:17 +0200	rocblas.x86_64-linux	rocblas-5.7.1	x86_64-linux
✓ ● 714971	27 May 13:48 +0200	rochpl.x86_64-linux	rochpl-5.6.1	x86_64-linux
✓ ● 714964	27 May 13:44 +0200	rocblas.x86_64-linux	rocblas-5.6.1	x86_64-linux
✓ ● 714973	27 May 11:30 +0200	rochpl.x86_64-linux	rochpl-5.4.4	x86_64-linux
✓ ● 714972	27 May 11:06 +0200	rochpl.x86_64-linux	rochpl-5.5.1	x86_64-linux
✓ ● 714966	27 May 11:04 +0200	rocblas.x86_64-linux	rocblas-5.5.1	x86_64-linux
✓ ↑ 714948	27 May 09:50 +0200	py-melissa-core.x86_64-linux	py-melissa-core-1.0.0-2.d8de4d5	x86_64-linux

<https://hpc.guix.info/blog/2023/03/contiguous-integration-and-continuous-delivery-for-hpc/>

Build details

[Action ▾](#)

Build ID	938839
Evaluation	8086586 (guix-hpc)
Status	Succeeded
System	x86_64-linux
Name	rocblas-5.6.1
Duration	8302 seconds
Finished	15 Sep 01:23 +0200
Weather	Still succeeding
Log file	pretty , raw
Derivation	/gnu/store/01rp7gjfgv5m9dfm1579hrk812aqqc0y-rocblas-5.6.1.drv
Dependencies	cmake-minimal-3.24.2 tar-1.34 gzip-1.13 bzip2-1.0.8 file-5.45 diffutils-3.10 patch-2.7.6

instance at <https://guix.bordeaux.inria.fr>

Build details

Action ▾

Build ID	952555
Evaluation	8092988 (guix-hpc)
Status	✖ Failed
System	x86_64-linux
Name	maphys-1.0.0
Duration	211 seconds
Finished	16 Sep 13:25 +0200
Weather	⬇ New failure Channel changes compared to the previous (successful) build :
	guix c0d4bd5 → 034eb1b
Log file	pretty , raw
Derivation	/gnu/store/kh77cdzsfxmh7h9bbr1nk2vzqw7dqb6b-maphys-1.0.0.drv
Dependencies	✓ hwloc-2.11.1

Wrapping up.

Robust deployment
is a precondition
for **faithful continuous integration.**

Faithful continuous integration
is a requirement
for **reliable continuous delivery.**



ludovic.courtes@inria.fr | @civodul@toot.aquilenet.fr

<https://hpc.guix.info>

Bonus slides!

2. Flexibility.

- ▶ <https://github.com/guix-science/guix-science>
 - ▶ RStudio Server, JupyterLab, Grid Engine, ...
- ▶ <https://github.com/guix-science/guix-science-nonfree>
 - ▶ CUDA, PyTorch with CUDA, oneAPI MKL, ...
- ▶ <https://gitlab.inria.fr/guix-hpc/guix-hpc>
 - ▶ ROCm/HIP, StarPU, Chameleon, Maphys, Simgrid, ...
- ▶ ...

<https://hpc.guix.info/channels>

```
guix install hwloc \  
  --with-source=./hwloc-2.1rc1.tar.gz
```

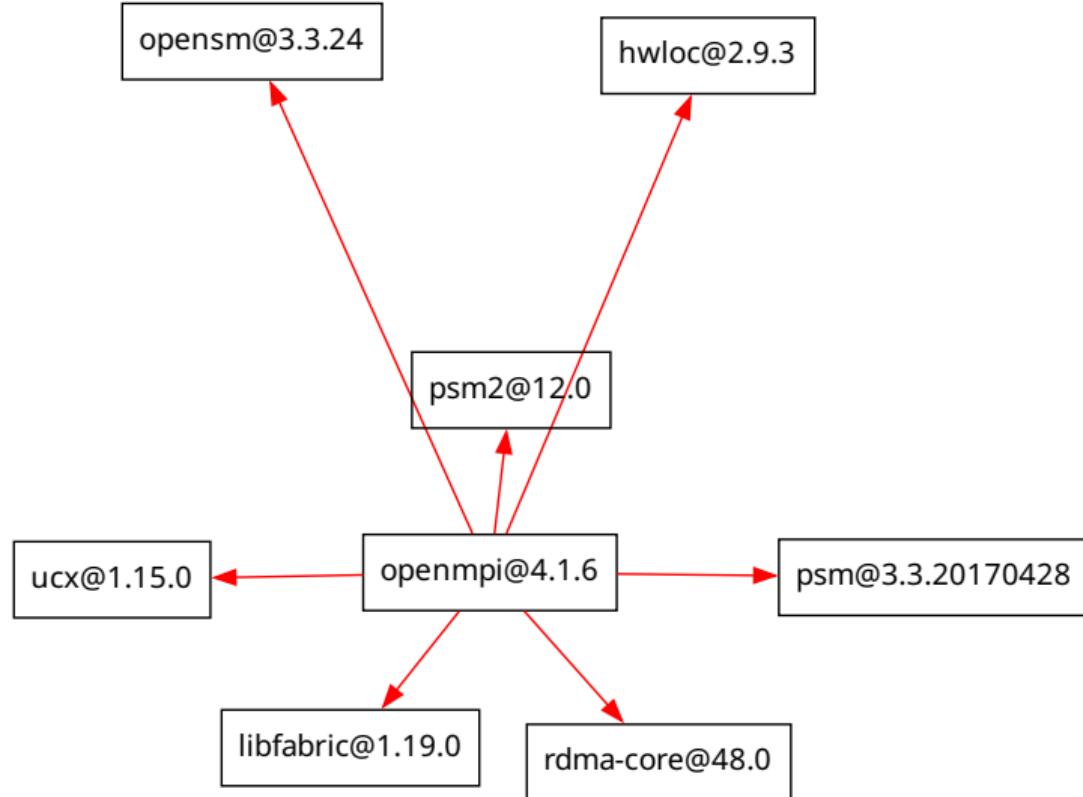
```
guix shell intel-mpi-benchmarks \  
  --with-input=openmpi=mpich
```

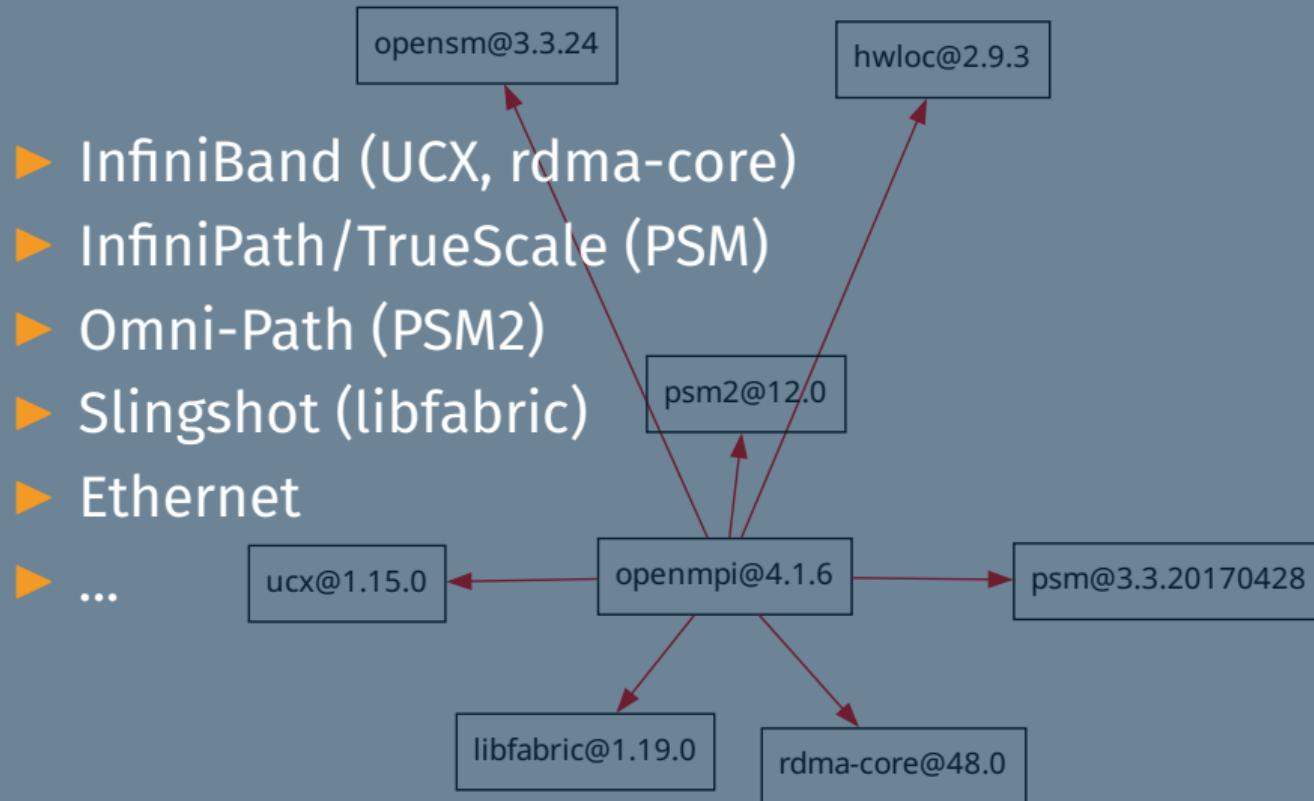
```
guix build dealii \  
  --with-branch=kokkos=master
```

- ▶ --with-commit
- ▶ --with-patch
- ▶ --with-input
- ▶ --with-c-toolchain
- ▶ --with-graft
- ▶ ...

[https://guix.gnu.org/manual/en/html_node/
Package-Transformation-Options.html](https://guix.gnu.org/manual/en/html_node/Package-Transformation-Options.html)

3. Performance portability.





```
$ guix shell eigen-benchmarks -- \
  benchBlasGemm 240 240 240
240 x 240 x 240
cblas: 0.20367 (16.289 GFlops/s)
eigen : 0.285149 (11.635 GFlops/s)
```

```
$ guix shell eigen-benchmarks -- \
  benchBlasGemm 240 240 240
240 x 240 x 240
cblas: 0.20367 (16.289 GFlops/s)
eigen : 0.285149 (11.635 GFlops/s)
```

**Package
multi-versioning**

```
$ guix shell --tune eigen-benchmarks -- \
  benchBlasGemm 240 240 240
guix shell: tuning for CPU micro-architecture skylake
240 x 240 x 240
cblas: 0.203131 (16.333 GFlops/s)
eigen : 0.0929638 (35.688 GFlops/s)
```



<https://hpc.guix.info/blog/2024/01/hip-and-rocm-come-to-guix/>

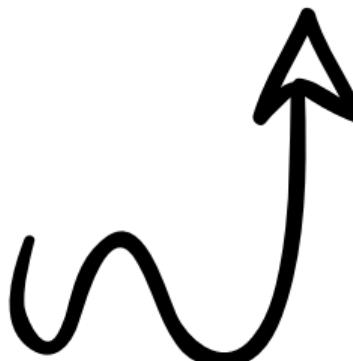
5. Open science & reproducible research.



Software Heritage



Guix



<https://hal.science/hal-04586520v1>

- ▶ P. Swartvagher, *On the Interactions between HPC Task-based Runtime Systems and Communication Libraries*, PhD thesis, Dec. 2022
- ▶ M. Felšöci, *Fast Solvers for High-Frequency Aeroacoustics*, PhD thesis, Feb. 2023
- ▶ N. Vallet *et al.*, *Toward practical transparent verifiable and long-term reproducible research using Guix*, Nature Scientific Data, Oct. 2022

Copyright © 2010, 2012–2024 Ludovic Courtès ludo@gnu.org.

GNU Guix logo, CC-BY-SA 4.0, <https://gnu.org/s/guix/graphics>.

Hand-drawn arrows by Freepik from flaticon.com.

DeLorean time machine picture © 2014 Oto Godfrey and Justin Morton, CC-BY-SA 4.0,
https://commons.wikimedia.org/wiki/File:TeamTimeCar.com-BTTF_DeLorean_Time_Machine-OtoGodfrey.com-JMortonPhoto.com-07.jpg.

Copyright of other images included in this document is held by their respective owners.

This work is licensed under the **Creative Commons Attribution-Share Alike 3.0** License. To view a copy of this license, visit <https://creativecommons.org/licenses/by-sa/3.0/> or send a letter to Creative Commons, 171 Second Street, Suite 300, San Francisco, California, 94105, USA.

At your option, you may instead copy, distribute and/or modify this document under the terms of the **GNU Free Documentation License, Version 1.3 or any later version** published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is available at <https://www.gnu.org/licenses/gfdl.html>.

The source of this document is available from <https://git.sv.gnu.org/cgit/guix/maintenance.git>.