Autonomie, flexibilité et reproductibilité : le pari de Guix-HPC

Ludovic Courtès

Atelier Guix-HPC

7 novembre 2024

Inria

Guix, un logiciel libre pour la reproductibilité des sciences en HPC

Date: 05 sep. 2017

Accueil > Actualités et évènements > Guix, un logiciel libre pour la reproductibilité des sciences en HPC

Mis à jour le 19/10/2020

Guix est un logiciel libre, développé sous les auspices du projet GNU* par une communauté enthousiaste d'organisations grandissantes : aujourd'hui entre 40 et 50 personnes y contribuent chaque mois. Il permet de reproduire des environnements logiciels. Récemment, les centres de recherche Inria Bordeaux – Sud-Ouest, Max Delbrück Center for Molecular Medicine (Berlin, Allemagne) et Utrecht Bioinformatics Center (Pays-Bas) ont décidé de collaborer autour de ce logiciel. Le point commun de ces trois établissements ? Tous sont ou ont des utilisateurs de logiciel de calcul haute performance (HPC), et dans ces structures, et bien d'autres, se pose la question de la reproductibilité des

https://www.inria.fr/fr/guix-un-logiciel-libre-pour-la-reproductibilite-des-sciences-en-hpc

Reproducible and User-Controlled Software Environments in HPC with Guix

Ludovic Courtès[1] and Ricardo Wurmus[2]

[1] Inria, Bordeaux, France
[2] Max Delbrück Center for Molecular Medicine, Berlin, Germany

Abstract. Support teams of high-performance computing (HPC) systems often find themselves between a rock and a hard place: on one hand, they understandably administrate these large systems in a conservative way, but on the other hand, they try to satisfy their users by deploying up-to-date tool chains as well as libraries and scientific software. HPC system users often have no guarantee that they will be able to reproduce results at a later point in time, even on the same system—software may have been upgraded, removed, or recompiled under their feet, and they have little hope of being able to reproduce the same software environment elsewhere. We present GNU Guix and the functional package management paradigm and show how it can improve reproducibility and sharing among researchers with representative use cases.

https://hal.science/hal-01161771/ (2015)

https://hpc.guix.info/

2022–2023 ACTIVITY REPORT

https://hpc.guix.info/blog/tag/activity-report/

“Guix-HPC aims to tackle the following high-level objectives :

- Reproducible **scientific workflows**. (…)
- **Cluster usage**. (…)
- **Outreach** & user support. (…)”

By **total** number of packaged projects

1. nix (nixpkgs unstable) - 99675
2. AUR - 75638
3. Debian+derivs (Raspbian Testing) - 41570
4. FreeBSD Ports - 31260
5. GNU Guix - 28533
6. Fedora (Fedora 38) - 24051
7. Gentoo (LiGurOS stable) - 19591
8. ALT Sisyphus - 19313
9. MacPorts - 19128
10. Rosa (Rosa 2016.1) - 18928

By **total** number of packaged projects

1. nix (nixpkgs unstable) - 99675
2. AUR - 75638
3. Debian+derivs (Raspbian Testing) - 41570
4. FreeBSD Ports - 31260
5. GNU Guix - 28533
6. Fedora (Fedora 38) - 24051
7. Gentoo (LiGurOS stable) - 19591
8. ALT Sisyphus - 19313
9. MacPorts - 19128
10. Rosa (Rosa 2016.1) - 18928

+ `https://hpc.guix.info/channels` → **50k+**

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

5 centres Inria

9 logiciels CEA

CNRS, univ., ...

LLNL, Stanford, ...

https://numpex.org/fr/exa-di-developpement-et-integration/

CINES

5 centres Inria

TGCC

9 logiciels CEA

CNRS, univ., ...

IDRIS

LLNL, Stanford, ...

mésocentres

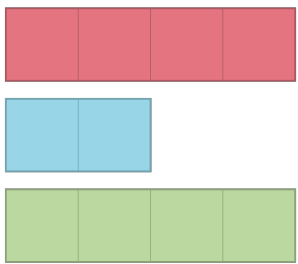
https://numpex.org/fr/exa-di-developpement-et-integration/


https://hpc.guix.info/events/2023/workshop/


[bob@supercomp ~]$ module av mpi
-------- /cm/shared/modules/generic/modulefiles ---
mpi/intel/2019.4.243
mpi/openmpi/2.0.4
mpi/openmpi/3.1.4
mpi/openmpi/4.0.1
mpi/openmpi/4.0.1-intel
mpi/openmpi/4.0.2
mpi/openmpi/4.0.2-testing
mpi/openmpi/4.0.3
mpi/openmpi/4.0.3-mlx
mpi/openmpi/4.0.7
mpi/openmpi/4.1.1
mpi/openmpi/4.1.5
mpi/openmpi/5.0.1
[bob@supercomp ~]$ module load openmpi/2.0.4
A-t-on progressé ?

Spack

CONDA



EASYBUILD.io

building software with ease

APPTAINER

Quota d'inodes *: Le quota d'inode est super restrictif si on fait des installations locales multiples (Spack ou miniConda) [...] ce qui nous oblige à archiver certaines installations pour tester d'autres versions*

– Comité des utilisateurices de l'IDRIS du 26 septembre 2024

Quota d'inodes *: Le quota d'inode est super restrictif si on fait des installations locales multiples (Spack ou miniConda) [...] ce qui nous oblige à archiver certaines installations pour tester d'autres versions*

Réponse IDRIS *: (...) En cas de besoin contacter l'assistance.*

– Comité des utilisateurices de l'IDRIS du 26 septembre 2024

A-t-on progressé...

- en autonomie ?

A-t-on progressé…

- en autonomie ?
- en reproductibilité ?

A-t-on progressé…

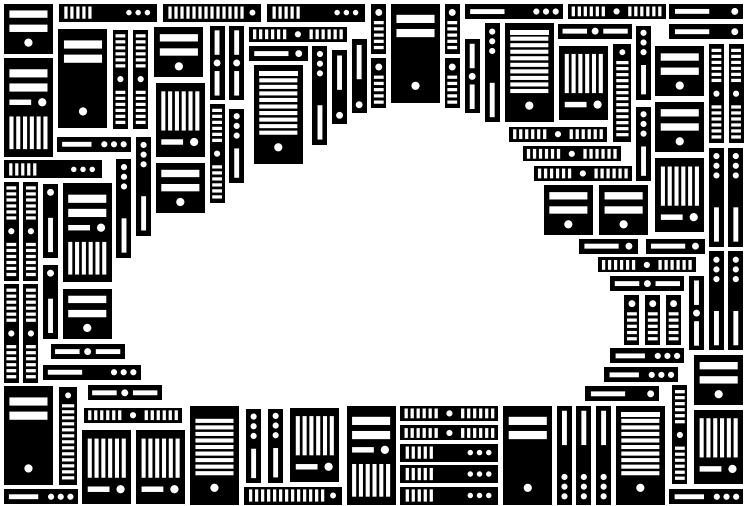
- en autonomie ?
- en reproductibilité ?
- en coopération ?

A-t-on progressé…

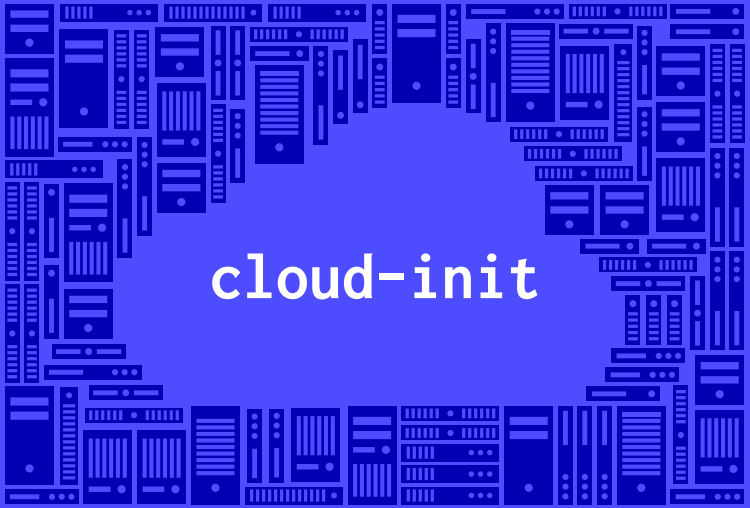
- en autonomie ?
- en reproductibilité ?
- en coopération ?
- en portabilité des performances ?

La suite ?

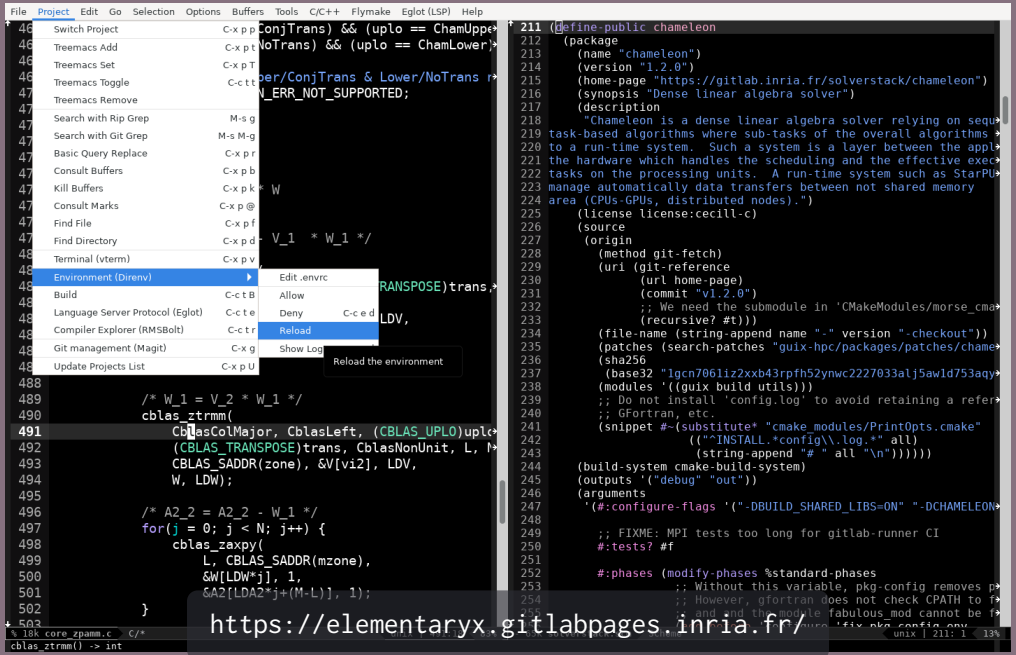
- **faire entendre la voix**
  reproductibilité + performance
- **montrer nos résultats** (cf. Emmanuel, Romain)
- **acculturer les admins sys**



There is NO CLOUD, just other people's computers



There is NO CLOUD, just other people's computers


File Project Edit Go Selection Options Buffers Tools C/C++ Flymake Eglot (LSP) Help
Switch Project C-x p p
Treemacs Add C-x p t
Treemacs Set C-x p T
Treemacs Toggle C-c t
Treemacs Remove
Search with Rip Grep M-s g
Search with Git Grep M-s M-g
Basic Query Replace C-x p r
Consult Buffers C-x p b
Kill Buffers C-x p k
Consult Marks C-x p @
Find File C-x p f
Find Directory C-x p d
Terminal (vterm) C-x p v
Environment (Direnv)
Build C-c t B
Language Server Protocol (Eglot) C-c t e
Compiler Explorer (RMSBolt) C-c t r
Git management (Magit) C-x g
Update Projects List C-x p U
Edit .envrc
Allow
Deny C-c e d
Reload
Show Log
Reload the environment
define-public chameleon
(package
(name "chameleon")
(version "1.2.0")
(home-page "https://gitlab.inria.fr/solverstack/chameleon")
(synopsis "Dense linear algebra solver")
(description
"Chameleon is a dense linear algebra solver relying on sequ
task-based algorithms where sub-tasks of the overall algorithms
to a run-time system. Such a system is a layer between the appl
the hardware which handles the scheduling and the effective exec
tasks on the processing units. A run-time system such as StarPU
manage automatically data transfers between not shared memory
area (CPUs-GPUs, distributed nodes).")
(license license:cecill-c)
(source
(origin
(method git-fetch)
(uri (git-reference
(url home-page)
(commit "v1.2.0")
;; We need the submodule in 'CMakeModules/morse_cma
(recursive? #t)))
(file-name (string-append name "-" version "-checkout"))
(patches (search-patches "guix-hpc/packages/patches/chame
(sha256
(base32 "1gcn7061iz2xxb43rpfh52ynwc2227033alj5aw1d753aqy
(modules '((guix build utils)))
;; Do not install 'config.log' to avoid retaining a refer
;; GFortran, etc.
(snippet #~(substitute* "cmake_modules/PrintOpts.cmake"
((\"^INSTALL.*config\\.log.*\" all)
(string-append \"# \" all \"\n\"))))))
(build-system cmake-build-system)
(outputs '("debug" "out"))
(arguments
'(#:configure-flags '("-DBUILD_SHARED_LIBS=ON" "-DCHAMELEON
#:tests? #f
#:phases (modify-phases %standard-phases
;; Without this variable, pkg-config removes p
;; However, gfortran does not check CPATH to f
;; and so the module fabulous_mod cannot be f
https://elementaryx.gitlabpages.inria.fr/
cblas_ztrmm() -> int

```
(define (make-osvariant-cluster base-os)
  (operating-system
    (inherit base-os)
    (services
     (modify-services (operating-system-user-services base-os)
       (sysctl-service-type
        config
        => (sysctl-configuration
            (settings (append '(("net.ipv4.conf.all.arp_ignore" . "1")
                                ("net.ipv4.conf.all.arp_announce" . "2"))
                              %default-sysctl-settings)))))))))

(define base-os
  (make-osvariant-cluster
   (make-osvariant-nssldap
    (make-osvariant-swap %glicid-one-disk-vm-os))))
```

https://gitlab.univ-nantes.fr/glicid-public/guix-glicid

```
(define (make-osvariant-cluster base-os)
  (operating-system
    (inherit base-os)
    (services
     (modify-services (operating-system-user-services base-os)
       (sysctl-service-type
        config
        => (sysctl-configuration
            (settings (append '(("net.ipv4.conf.all.arp_ignore" . "1")
                                ("net.ipv4.conf.all.arp_announce" . "2"))
                              %default-sysctl-settings)))))))))

(define base-os
  (make-osvariant-cluster
   (make-osvariant-nssldap
    (make-osvariant-swap %glicid-one-disk-vm-os))))
```

GLiCID <3

https://gitlab.univ-nantes.fr/glicid-public/guix-glicid

Approprions-nous les moyens de calcul.

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

https://hpc.guix.info