



Guix, le transfert comme partage

Ludovic Courtès

Journée SED, 30 mai 2024

Inria

Environnements reproductibles : 2 fichiers, 2 commandes

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

DEPARTMENT: SCIENTIFIC PROGRAMMING

Reproducibility and Performance: Why Choose?

Ludovic Courtès  Inria, 75012, Paris, France

Code Staging in GNU Guix

Ludovic Courtès
Inria
Bordeaux, France

Abstract

GNU Guix is a “functional” package manager that builds upon earlier work on Nix. Guix implements high-level abstractions such as packages and operating system services as domain-specific languages (DSLs) embedded in Scheme. It also implements build actions and operating system orchestration in Scheme. This leads to a multi-tier programming environment where embedded code snippets are staged for eventual execution.

means that software build processes are considered as pure functions: given a set of inputs (compiler, libraries, build scripts, and so on), a package’s build function is assumed to always produce the same result. Build results are stored in an immutable persistent data structure, the *store*, implemented as a single directory, `/gnu/store`. Each entry in `/gnu/store` has a file name composed of the hash of all the build inputs used to produce it, followed by a symbolic name. For example, `/gnu/store/yr9rk90jf...-gcc-7.1.0.identi`

Building a Secure Software Supply Chain with GNU Guix

Ludovic Courtès^a
^a Inria, France

Abstract The *software supply chain* is becoming a widespread analogy to designate the series of steps taken to go from source code published by developers to executables running on the users’ computers. A security vulnerability in any of these steps puts users at risk, and evidence shows that attacks on the supply chain are becoming more common. The consequences of an attack on the software supply chain can be tragic in a society that relies on many interconnected software systems, and this has led research interest as well as governmental incentives for supply chain security to rise.

Source Code Archiving to the Rescue of Reproducible Deployment

Ludovic Courtès
ludovic.courtes@inria.fr
Inria
Bordeaux, France

Simon Tournier
simon.tournier@inserm.fr
Université Paris Cité
Paris, France

Timothy Sample
samplet@ngyro.com
Saskatoon, Canada

Stefano Zacchiroli
stefano.zacchiroli@telecom-paris.fr
LTCI, Télécom Paris, Institut Polytechnique de Paris
Palaiseau, France

1. Outil pour la recherche.

<https://hpc.guix.info/channel/guix-hpc>

- ▶ AeroSol
- ▶ BitPit
- ▶ Chameleon
- ▶ Hawen
- ▶ Maphys
- ▶ NewMadeleine
- ▶ PaSTiX
- ▶ StarPU

- ▶ ...



<https://hpc.guix.info/channel/guix-hpc>

- ▶ AeroSol
- ▶ BitPit
- ▶ Chameleon
- ▶ Hawen
- ▶ Maphys
- ▶ NewMadeleine
- ▶ PaSTiX
- ▶ StarPU
- ▶ medInria, dtk
- ▶ ...



- ▶ 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

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



NumPEX

CNRS, univ., ...

IDRIS

LLNL, Stanford, ...

mésocentres

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

2. Vecteur de transfert.

Accueil ▀ Formations ▀ Guix, optimiser le cycle de vie du logiciel, du développement au déploiement

GÉNIE LOGICIEL

Guix, optimiser le cycle de vie du logiciel, du développement au déploiement



<https://www.inria-academy.fr/formation/guix/>



CENTRE EUROPÉEN DE RECHERCHE ET DE FORMATION AVANCÉE EN **CALCUL SCIENTIFIQUE**

<https://cerfacs.fr/avbp7x/>



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

3. Un collectif.





- ▶ **Guix depuis 2012**

- ▶ **Guix-HPC depuis 2017** (Inria, MDC, UBC, UTHCS)

- ▶ **50,000+ paquets**

- ▶ **100 contributeurices** chaque mois

- ▶ conférences, ateliers, formations, Café Guix, ...

- ▶ du **long terme**
- ▶ travail **entre pairs**
- ▶ construction d'**un commun**



<https://hpc.guix.info>

ludovic.courtes@inria.fr

Bonus !

```
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 rocm-opencl-runtime git
```

```
guix shell -D rocm-ocl-runtime git --container
```



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

```
(specifications->manifest  
  '("gcc-toolchain" "coreutils" "grep" "sed"  
    "cmake" "rocm-cmake" "rocm-ocl-runtime"  
    "rocm-device-libs"))
```

```
$ guix shell -D rocm-opencl-runtime git \  
--export-manifest
```

```
$ guix shell -D rocm-ocl-runtime 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 "rocm-ocl-runtime"))))
```

```
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.sv.gnu.org/git/guix.git
```

```
commit: cabba9e15900d20927c1f69c6c87d7d2a62040fe
```

```
alice@supercomp$ guix pull --commit=cabba9e
```

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

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

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

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 4.0](https://creativecommons.org/licenses/by-sa/4.0/) License. To view a copy of this license, visit <https://creativecommons.org/licenses/by-sa/4.0/> 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](https://www.gnu.org/licenses/gfdl.html) 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>.