mathrice

Pôle CS SCOSI

EasyBuild

installation de logiciels scientifiques - retour d'expérience -

Philippe Marion

27 septembre 2017

Sommaire

mathrice

Pôle CS SCOSI

Sommaire

mathrice

Pôle CS SCOSI

généralités

le Ying et le Yang admin sys. / utilisateurs

mathrice

Pôle CS SCOSI

admin sys

- logiciels matures
- éviter les variantes exotiques
- limiter les mises à jours ...aux indispensables(!)

☞ système et des logiciels stable

utilisateurs

- softs les plus récents
- plusieurs variantes
- les derniers outils
- maximum de convivialité / flexibilité

☞ contrôler toute la pile de logiciels

généralités (suite)

cluster : autres enjeux et contraintes

mathrice

Pôle CS SCOSI

- ☞ reproductibilité, a minima :
  - dans le temps
  - entre collaborateur
- ☞ performances des logiciels
  - compilés à partir des sources
  - optimisés (architecture)
- ☞ convivialité accessiblité
- ☞ par essence, plateforme fortement multi-utilisateurs

conséquences

l'admin sys. HPC fait très rapidement face à une combinatoire de soft/dépendances/environnement inextricable. Etape 1 : chargeur d'environnement. Etape 2 : package manager

notre contexte

petite structure, récente, forte montée en charge

mathrice

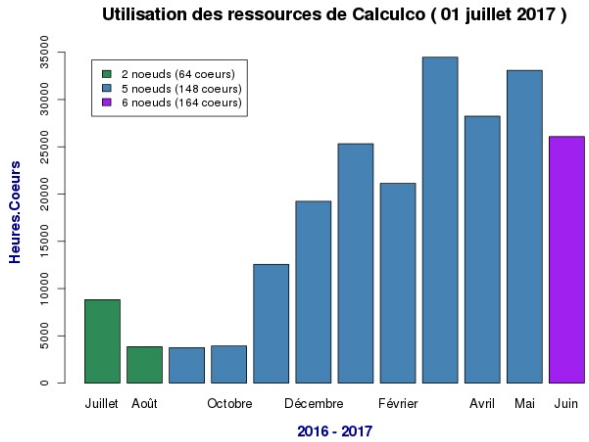
Pôle CS SCOSI

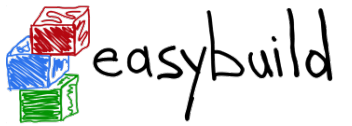
Service commun 2014. Plateforme PCS 2015

démarrage : Université + BQR LMPA - LISIC

montée en charge : élargissement du public, BQR, ANR

«à dimension humaine... » : relations admin.sys/users gérables ;-)





TACC (Lmod)

Sommaire

mathrice

Pôle CS SCOSI

Sommaire

mathrice

Pôle CS SCOSI

lmod (TACC)

gestion de modules

mathrice

Pôle CS SCOSI

- ☞ description :
  - sytème de gestion de modules basé sur le langage lua (version moderne un peu plus conviviale de *module Tools* (Tcl/C ou Tcl)
  - permet de changer dynamiquement d'environnement (modulefiles)
  - lmod, developpé au centre de Calcul de l'université du Texas TACC
- ☞ installation :
  - https://github.com/TACC/Lmod
  - package distribution

Lmod : modules

quelques commandes

charger un environnement :

`module help (ml help)`

`module avail (ml av)` liste les modules disponibles

`module load opencv ( ml opencv)` charger le module

`module spider opencv ( ml spider opencv)` descriptif plus complet

enlever :

`module purge (ml purge)` tous les modules

`module unload OpenCV ( ml -OpenCV))` retirer

Sauvegarder, restaurer :

`module save` (crée ~/.lmod.d/default)

`module save env1` (crée ~/.lmod.d/env1)

`module restore env1` (restaure l'environnement de module env1)

www-calculco.univ-littoral.fr/utilisation/environnement-logiciel

Lmod
illustration

mathrice

Pôle CS SCOSI

Terminal - pmarion@calculco: ~

Fichier Édition Affichage Terminal Onglets Aide

```
pmarion@calculco:~$ ml av

------------------------ /nfs/opt/apps/modulefiles/Core -------------------------
   OpenCV/3.2 (D)    julia/0.6    matlab/R2016b    matlab/R2017a (D)

------------------- /nfs/opt/apps/lmod/lmod/modulefiles/Core -------------------
   lmod/7.4.7    settarg/7.4.7

------------------ /nfs/opt/easybuild/x86-64/modules/compiler ------------------
   GCC/4.9.3-2.25    GCC/6.3.0-2.27 (D)    GCCcore/4.9.3    GCCcore/6.3.0 (D)

------------------- /nfs/opt/easybuild/x86-64/modules/devel --------------------
   Autoconf/2.69-foss-2016a
   Autoconf/2.69                                    (D)
   Automake/1.15-foss-2016a
   Automake/1.15                                    (D)
   Autotools/20150215-foss-2016a
   Autotools/20150215                               (D)
   CMake/3.5.2-foss-2016a
   GObject-Introspection/1.48.0-foss-2016a
   JUnit/4.12-Java-1.8.0_92
   M4/1.4.17-foss-2016a
   M4/1.4.17-GCCcore-4.9.3
```

Terminal - pmarion@calculco: ~

Fichier Édition Affichage Terminal Onglets Aide

```
No modules loaded
pmarion@calculco:~$ ml matlab
pmarion@calculco:~$ ml

Currently Loaded Modules:
  1) matlab/R2017a



pmarion@calculco:~$
```

Sommaire

mathrice

Pôle CS SCOSI

EasyBuild

gestion de modules

mathrice Pôle CS SCOSI

- ☞ description :
  - package manager de logiciels scientifique
  - framework en Python

  http://easybuild.readthedocs.io, projet initié à l'université de Ghent (Belgique)
- ☞ installation :
  - prérequis : un chargeur de modules, par ex. *Lmod* ;-)
  - pip
  - méthode recommandée : bootstrap ...
  - un utilisateur (std), un répertoire logiciels
  - ... et quelques variables d'environnements EASYBUILD_PREFIX EASYBUILD_SOURCEPATH EASYBUILD_OPTARCH

EasyBuild (suite)

...et la magie opère ! :

séquence basique :

`eb --help`

`eb -S eigen` recherche librairie Eigen

`eb Eigen-3.2.7-foss-2016a.eb -D` (–dry-run)

`eb Eigen-3.2.7-foss-2016a.eb -r` (–robot)

écosystème - lexique

easyblock : plugin d'Easybuild qui définit pour chaque soft une procédure de compilation/installation

fichier easyconfig *.eb : le fichier d'installation.

toolchain : chaîne compilateurs - librairies

foss : gcc g+++ binutil OpenMPI OpenBLAS ScaLAPACK FTFW

intel : icc ifort MKL intelMPI

dummy : compilateurs et librairies du système

EasyBuild (suite)

exemple de fichiers *.eb

mathrice

Pôle CS SCOSI

```
name = 'Eigen'
version = '3.2.7'

homepage = 'http://eigen.tuxfamily.org/index.php?title=

description = """Eigen is a C++ template library for li

matrices, vectors, numerical solvers, and related algor
toolchain = {'name': 'foss', 'version': '2016a'}
source_urls = [BITBUCKET_SOURCE]
sources = ['%(version)s.tar.bz2']
moduleclass = 'math'
```

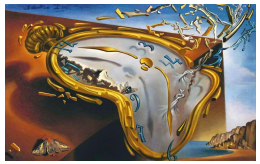
Sommaire

mathrice

Pôle CS SCOSI

conclusions

...sur le retour d'expérience

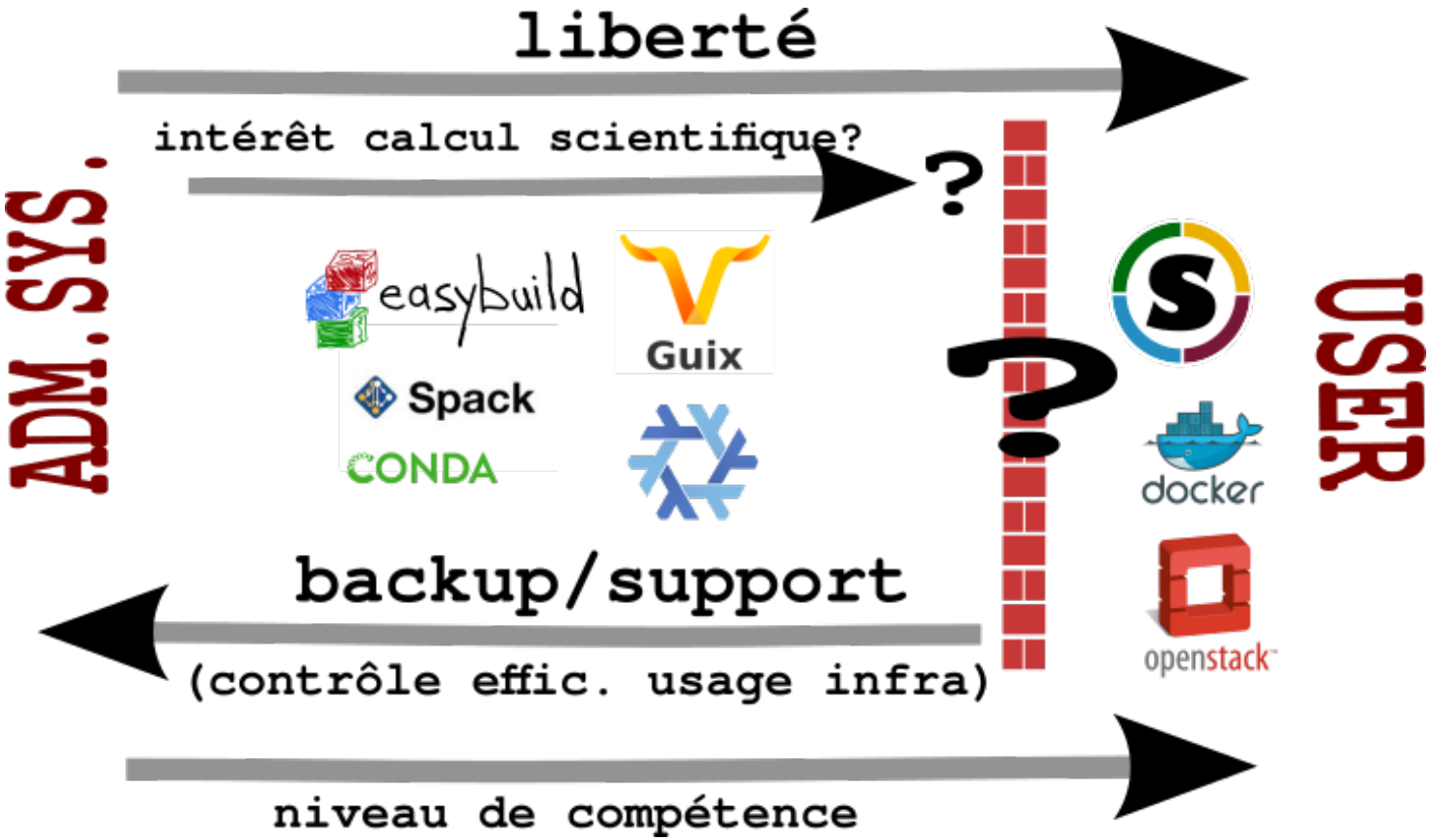


- ☞ les plus (**fait le job !**) :
  - bon rapport offre logiciels récents / système stable (Debian)
  - modules files (Lmod) : transparence pour les utilisateurs (proc. unique)
  - optimisation matérielle (commande unique)
  - reproductibilité, portabilité (limitées à la plateforme ?)
  - fiabilité, pérennité : importante communauté, bonne référence, nombreux *.eb
- ☞ les moins :
  - documentation, prise en main.

Questions ?

...ou plutôt interrogations ! ?

- ☞ Où en êtes-vous ?
- ☞ Un «débordement» par les développeurs NIX Guix est-il possible ?



mathrice

Pôle CS
SCOSI

Questions ?

quelques liens

autour de...ou, pour aller plus loin ?

https://github.com/easybuilders/easybuild/wiki/Step-by-step-guide

https://guix-hpc.bordeaux.inria.fr/blog/

https://easybuilders.github.io/easybuild/

https://calcul.math.cnrs.fr/spip.php?article285

https://lmod.readthedocs.io/

https://conda.io/docs/