UL HPC School 2017
PS3b: Software Build and Customization using Easybuild on the UL HPC Platform

UL High Performance Computing (HPC) Team
Sarah Peter

University of Luxembourg (UL), Luxembourg
http://hpc.uni.lu
Latest versions available on Github:

UL HPC tutorials:  
https://github.com/ULHPC/tutorials

UL HPC School:  
http://hpc.uni.lu/hpc-school/

PS3b tutorial sources:  
https://github.com/ULHPC/tutorials/tree/devel/advanced/easybuild
Introduction

Summary

1. Introduction

2. Software/Modules Management
Main Objectives of this Session

- Understand LMod / Environment module
- Discover Easybuild
- Build your own software on top of the provided software set
Summary

1 Introduction

2 Software/Modules Management
Based on Environment Modules / LMod

- convenient way to dynamically change the users’ environment $PATH$
- permits to easily load software through module command

Currently on UL HPC:

- 133 software packages, in multiple versions, within 18 categories
- reworked software set for iris cluster and soon deployed everywhere
  - ✓ RESIF v2.0, allowing [real] semantic versionning of released builds
- hierarchical organization
  
  Ex: toolchain/{foss,intel}

$>$ module avail
   # List available modules

$>$ module load <category>/<software>[/<version>]

https://hpc.uni.lu/users/software/
Easybuild: open-source framework to (automatically) build scientific software

Why?: "Could you please install this software on the cluster?"

→ Scientific software are often painful to build
  ✓ non-standard build tools / incomplete build procedure
  ✓ hardcoded parameters and/or poor/outdated documentation

→ EasyBuild helps to facilitate this task
  ✓ consistent software build and installation framework
  ✓ automatically generates LMod modulefiles

$> \text{module use /path/to/easybuild}
$> \text{module load tools/EasyBuild toolchain/intel}
$> \text{eb -S HPL} \quad \# \text{Search for recipes for HPL software}
$> \text{eb HPL-2.2-intel-2017a.eb} \quad \# \text{Install HPC 2.2 w. Intel toolchain}
**RESIF**: Revolutionary EasyBuild-based Software Installation Framework

- Automatic Management of **software sets**
- Fully automates software builds and supports all available toolchains
- Clean (hierarchical) modules layout to facilitate its usage
- **Versioning** of software set builds
- Easyconfig files from multiple sources
- Define options and software in easy to read **yaml** files
- Targeted at ULHPC sysadmins use case to build many softwares in one go
Available software sets

- **Gaia**
  - **core**: available by default
  - **lcsb**: load with
    
    ```
    module use $RESIF_ROOTINSTALL/lcsb/modules/all
    ```

- **Iris**
  - **default**: available by default
  - **bioinfo**: load with
    
    ```
    module use /opt/apps/resif/data/stable/bioinfo/modules/all
    ```
We provide software that is used by many users on the cluster.

What the users should install themselves:

- Python packages
- R packages
- Perl modules
- Software only used by 1 or 2 persons
We provide several how-tos and tutorials for installing software as a user on the ULHPC website and in our HPC school tutorials:

- Installation with **EasyBuild**
- Installation from source with `configure` and `make`
- Installation of Python packages and usage of virtual environments
- Installation of Perl modules and how to set up a local library

Or just download precompiled binaries (see Bioinformatics tutorial) ;-)
Thank you for your attention...

Questions?

http://hpc.uni.lu

The UL High Performance Computing (HPC) Team
University of Luxembourg, Belval Campus:
Maison du Nombre, 4th floor
2, avenue de l’Université
L-4365 Esch-sur-Alzette
mail: hpc@uni.lu

1 Introduction

2 Software/Modules Management