Entorno de módulos (Environment Modules)

TeideHPC es un cluster compartido por cientos de usuarios, por lo que no es lo mismo la forma de proporcionar múltiples aplicaciones, en sus múltiples versiones, a multitud de usuarios, en un entorno de computación distribuida, que en un ordenador personal o en tu propio servidor.

Environment Modules es un sistema de gestión de software que permite a los usuarios de sistemas Unix-like, como Linux y macOS, y particularmente en entornos de computación de alto rendimiento (HPC), **dinámicamente modificar su entorno, acceder a diferentes compiladores, bibliotecas y software que pueden tener múltiples versiones y dependencias.**

Existen diferente implementaciones de Environment Modules pero la usada actualmente en TeideHPC es ***Lmod***.

Lmod

Lmod es un acrónimo de **Lua Module System** es una implementación específica del concepto de Environment Modules para la gestión de entornos de software en sistemas de alto rendimiento (HPC).

Lmod permite a los usuarios:

- cargar y descargar módulos de software dinámicamente.
- cambia las variables de entorno como PATH y LD_LIBRARY_PATH.
- facilitar el uso de diferentes versiones de aplicaciones y bibliotecas sin interferencias entre ellas.

Puedes obtener más información sobre Lmod en https://lmod.readthedocs.io/en/latest/.

¿Cómo se organiza el software en TeideHPC? Nomenclatura plana.

En TeideHPC existen 2 clusters, TeideHPC y AnagaGPU

Por este motivo, el conjunto de módulos disponibles **depende del Cluster que vaya a usar.**

Cada cluster tiene sus propios nodos de login y **no disponen del mismo software instalado.**

El esquema de módulos utiliza una nomenclatura específica para sus módulos, que normalmente sigue el formato:

```
<nombre_del_software>/<versión>-<toolchain>[-<variantes>]
```

- **Nombre_del_software**: Es el nombre del software o la biblioteca que se está instalando.
- **Versión**: Es la versión específica del software.
- **Toolchain**: Es la cadena de herramientas de compilación que incluye el compilador, la biblioteca MPI (si aplica), y otras bibliotecas de optimización de rendimiento que se usan para compilar el software. Por ejemplo, foss es una cadena de herramientas común que incluye GCC, OpenMPI, y otras.
- **Variantes**: Son modificadores adicionales que indican variantes específicas de la instalación, como optimizaciones específicas de hardware o la inclusión de ciertas características.

Por ejemplo, un módulo para la versión 6.0 de Unzip compilada con la cadena de herramientas GCCcore 12.2.0 podría tener una nomenclatura de módulo como

`UnZip/6.0-GCCcore-12.2.0`

Un ejemplo de módulo con distintas variantes y toolchain es el siguiente:

```
module spider 'Python'

----------------------------------------------------------------------------------------------------
  Python:
----------------------------------------------------------------------------------------------------
    Description:
      Python is a programming language that lets you work more quickly and integrate your
systems more
      effectively.

     Versions:
```

```
Python/2.7.18-GCCcore-11.2.0-bare
Python/2.7.18-GCCcore-11.2.0
Python/2.7.18-GCCcore-12.2.0-bare
Python/3.8.6-GCCcore-11.2.0
Python/3.8.6-intel-2021b
Python/3.9.6-GCCcore-11.2.0-bare
Python/3.9.6-GCCcore-11.2.0
Python/3.9.6-intel-2021b
Python/3.10.8-GCCcore-12.2.0-bare
Python/3.10.8-GCCcore-12.2.0
```

¿Qué es un *Toolchain*?

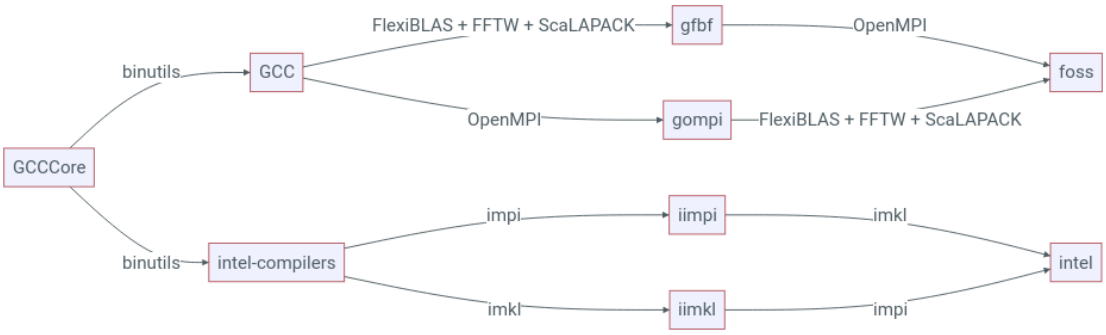
El término "toolchain" se refiere al conjunto de herramientas de software que se utilizan para compilar y enlazar programas.

Un toolchain típicamente **incluye compiladores para diferentes lenguajes de programación, bibliotecas de software matemáticas, y herramientas de enlazado y depuración**.

Los toolchains están definidos por un conjunto de componentes que incluyen:

- **Compiladores**: GCC (GNU Compiler Collection), Intel Compiler, etc.
- **Bibliotecas de comunicaciones MPI**: Por ejemplo, OpenMPI, MPICH, Intel MPI, etc.
- **Bibliotecas de matemáticas**: BLAS (Basic Linear Algebra Subprograms) y LAPACK (Linear Algebra Package).
- **Otras bibliotecas y herramientas**: Como GMP (GNU Multiple Precision Arithmetic Library), FFTW (Fastest Fourier Transform in the West), etc.

El siguiente diagrama puede ver cómo se divide el software en función del Toolchain elegido. **Básicamente se identifican 2 ramas en función del compilador principal: GNU GCC o Intel GCC**



Asimismo, **existen diferentes generaciones de estos toolchains que son incompatibles entre sí.**

Toolchain *foss*

El toolchain *foss* consta enteramente de software de código abierto (de ahí el nombre, derivado del término común 'FOSS', que es la abreviatura de "Free and Open Source Software").

En TeideHPC podemos identificar las siguientes generaciones para *foss*

foss	**binutils**	**GCC**	**Open MPI**	**FlexiBLAS**	**OpenBLAS**	**LAPACK**	**ScaLA**
2021b	2.37	11.2.0	4.1.1	3.0.4	0.3.18	(incl. with OpenBLAS)	2.1.0
2022a	2.38	11.3.0	4.1.4	3.2.0	0.3.20	(incl. with OpenBLAS)	2.2.0
2022b	2.39	12.2.0	4.1.4	3.2.1	0.3.21	(incl. with OpenBLAS)	2.2.0

Toolchain *intel*

El Toolchain *Intel* consta de compiladores y bibliotecas de Intel.

En TeideHPC podemos identificar las siguientes generaciones para *intel*

intel	**binutils**	**GCC**	**Intel compilers**	**Intel MPI**	**Intel MKL**
2021b	2.37	11.2.0	2021.4.0	2021.4.0	2021.4.0
2022a	2.38	11.3.0	2022.1.0	2021.6.0	2022.1.0
2022b	2.39	12.2.0	2022.2.1	2021.7.1	2022.2.1

En la siguiente sección puede ver cómo usar modules para ver y cargar el software disponible en TeideHPC.

`module spider foss`

```
----------------------------------------------------------
  foss:
----------------------------------------------------------
```

```
    Description:
      GNU Compiler Collection (GCC) based compiler toolchain, including OpenMPI for MPI
support, OpenBLAS (BLAS
      and LAPACK support), FFTW and ScaLAPACK.

     Versions:
       foss/2021b
       foss/2022b
----------------------------------------------------------
  To find other possible module matches execute:

      $ module -r spider '.*GCC.*'
```

```
module spider intel
```

```
----------------------------------------------------------
  intel:
----------------------------------------------------------
    Description:
      Compiler toolchain including Intel compilers, Intel MPI and Intel Math Kernel Library
(MKL).

     Versions:
       intel/2021b
       intel/2022b
     Other possible modules matches:
       intel-compilers

----------------------------------------------------------
  To find other possible module matches execute:

      $ module -r spider '.*intel.*'
```