

Introducción a la herramienta cmake



Parte I: bash, g++, cmake, gnuplot, gms, paraview



Practique los ejemplos en Gitpod



Open in Gitpod

Disponible en



¡Únete al grupo en Telegram!



CMake

Es un *generador de sistemas de compilación* de código abierto e independiente del compilador y/o plataforma, es decir, produce instrucciones para otros sistemas de compilación como Makefile, Ninja, Visual Studio, Qt Creator, Android Studio y Xcode. También incluye características que permiten la instalación, empaquetamiento y soporte nativo de pruebas de software.

Resultados del aprendizaje:

- ▶ Construir diversos ejemplos de proyectos de CMake en los lenguajes C/C++/Fortran/Python que construyan ejecutables u objetos compartidos/estáticos/interfaces.
- ▶ Correr pruebas con ctest, catch2, gtest, pytest.
- ▶ Usar dependencias de terceros en un proyecto CMake.

Está orientado a cualquier estudiante, profesor, investigador, programador que quiera aprender de Arch Linux y quiera aprender a usar efectivamente CMake en un proyecto basado en Dune.

No se asume que tenga experiencia en GNU/Linux o C++, pero sí familiarizado con algún lenguaje de programación.

Tres herramientas de línea de comando

- ▶ `/usr/bin/cmake`
- ▶ `/usr/bin/cpack`
- ▶ `/usr/bin/ctest`

Tres herramientas interactivas

- ▶ `/usr/bin/ccmake`
- ▶ `/usr/bin/cmake-gui`

Instalación

Esta tecnología imprescindible en el lenguaje C++ se encuentra disponible en mayoría de repositorios de distribuciones GNU/Linux importantes. Si desea tener una versión actual, puede instalar desde el repositorio [extra]

```
[user@host somedir]$ sudo pacman -Syu cmake graphviz plantuml gcovr cppcheck python-cpplint co
```

El proceso de construcción CMake

Tipos de construcción CMake

- ▶ Debug
- ▶ Release
- ▶ RelWithDebInfo
- ▶ MinSizeRel
- ▶ None

Generación automática de documentación con CMake

La documentación es una parte esencial de cualquier proyecto exitoso.

Veamos cómo integrar Doxygen con CMake para generar automáticamente la documentación para proyectos con CMake.

CMake generará un archivo `Doxyfile`.

Esperamos que Doxygen genere la documentación de la interfaz de programación de aplicaciones (API) para cada clase y sus diagramas de herencia con `dot` de `graphviz`.

Pruebas con ctest

Referencias

► Libros



Oliver Sander. *DUNE — The Distributed and Unified Numerics Environment*. First. Lecture Notes in Computational Science and Engineering 140. Springer International Publishing, 2020. ISBN: 978-3-030-59701-6. DOI: 10.1007/978-3-319-03038-8.

► Artículos



Peter Bastian et al. "The Dune framework: Basic concepts and recent developments". En: *Computers & Mathematics with Applications* 81.1 (1 de ene. de 2021). Development and Application of Open-source Software for Problems with Numerical PDEs, págs. 75-112. ISSN: 0898-1221. DOI: 10.1016/j.camwa.2020.06.007. URL: <https://www.sciencedirect.com/science/article/pii/S089812212030256X>.

► Sitios web



Santiago Torres Arias, Jesús Castro y Andrea Gómez. *Taller de contribución a Arch Linux – Cumbre de Contribuidores de Open Source Software (CCOSS)*. 22 de oct. de 2020. URL: <https://sg.com.mx/buzz/ponencias/ccoss-2020/taller-de-contribucion-arch-linux> (visitado 10-03-2021).



ArchWiki. *Arch Linux*. 16 de mayo de 2023. URL: https://wiki.archlinux.org/title/Arch_Linux (visitado 16-05-2023).