

Déployer une application

CPack

CPack, génération de paquets plateforme spécifique (.deb, .rpm, dépôts Ubuntu, etc) et outils d'installation (NSIS, nullsoft, Inno Setup). Utilisation autonome ou avec CMake, même syntaxe que CPack. Pour installation ou mise à jour (logiciel dédié ?) Qt Install Framework ?

```
// Dans CPackConfig.cmake
SET (CPACK_PACKAGE_NAME "compute_pi")
SET (CPACK_PACKAGE_VERSION "1.0")
SET (CPACK_PACKAGE_DESCRIPTION_SUMMARY "Parallel computation
of Pi")
SET (CPACK_PACKAGE_VENDOR "Sekou @ free.fr")
SET (CPACK_PACKAGE_DESCRIPTION_FILE
"${CMAKE_CURRENT_SOURCE_DIR}/README")
SET (CPACK_RESSOURCE_FILE_LICENCE
"${CMAKE_CURRENT_SOURCE_DIR}/Copyright.txt")
INSTALL (TARGETS computePi RUNTIME DESTINATION bin)

SET (CPACK_NSIS_HELP_LINK "http://devlper...")
SET (CPACK_DEBIAN_PACKAGE_DEPENDS "libglib-2.0-0 (>= 2.1)")
SET (CPACK_RPM_PACKAGE_REQUIRES "glib >= 2.1")

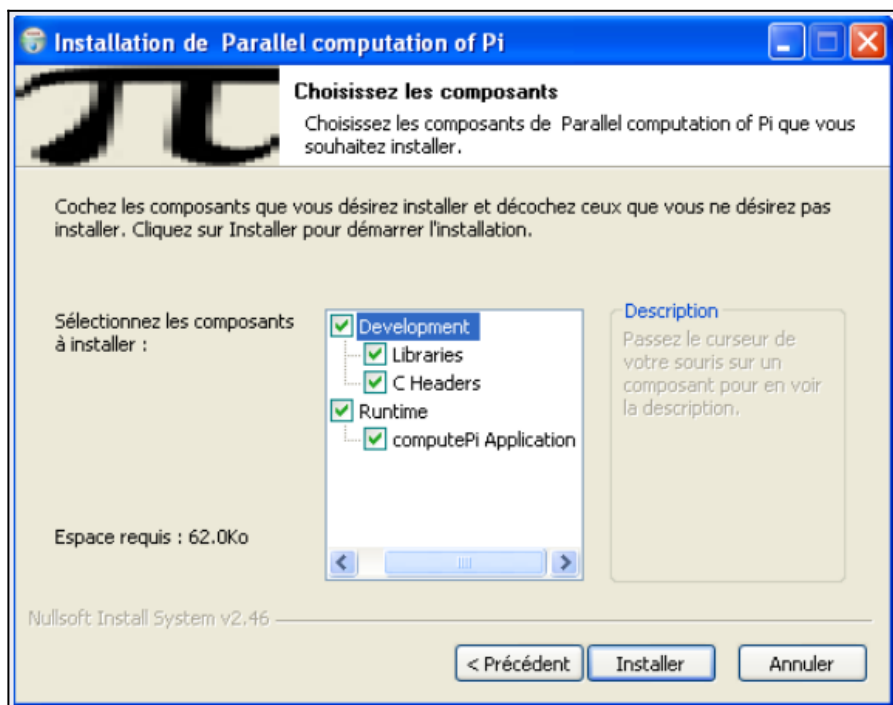
IF (WIN32)
    SET (CPACK_GENERATOR "ZIP; NSIS")
ELSE (WIN32)
    IF (APPLE)
        SET (CPACK_GENERATOR "TGZ")
    ELSE (APPLE)
        SET (CPACK_GENERATOR "TGZ; RPM; DEB")
    ENDIF (APPLE)
ENDIF (WIN32)

// composants
```

```

INSTALL (FILES ./include/compute_pi.h DESTINATION include
COMPONENT headers)
INSTALL (TARGETS compute_pi ARCHIVE DESTINATION lib
COMPONENT libraries)
INSTALL (TARGETS compute_pi DESTINATION bin COMPONENT
applications)
INSTALL (TARGETS computePi RUNTIME DESTINATION bin COMPONENT
applications)
SET (CPACK_COMPONENT_APPLICATIONS_GROUP "Runtime")
SET (CPACK_COMPONENT_LIBRARIES_GROUP "Development")
SET (CPACK_COMPONENT_HEADERS_GROUP "Development")
SET (CPACK_COMPONENTS_ALL applications libraries headers)
INCLUDE (CPack)

```



Installeur généré avec CPack

Plateforme spécifique

- Android
- iOS ?
- Arduino

Utiliser/créer un application store ?

Chapitre précédent	Sommaire principal	Chapitre suivant
Cours, C++		