pybind11 & CMake

Combining Python and C++ using pybind11
Introduction to CMake

02
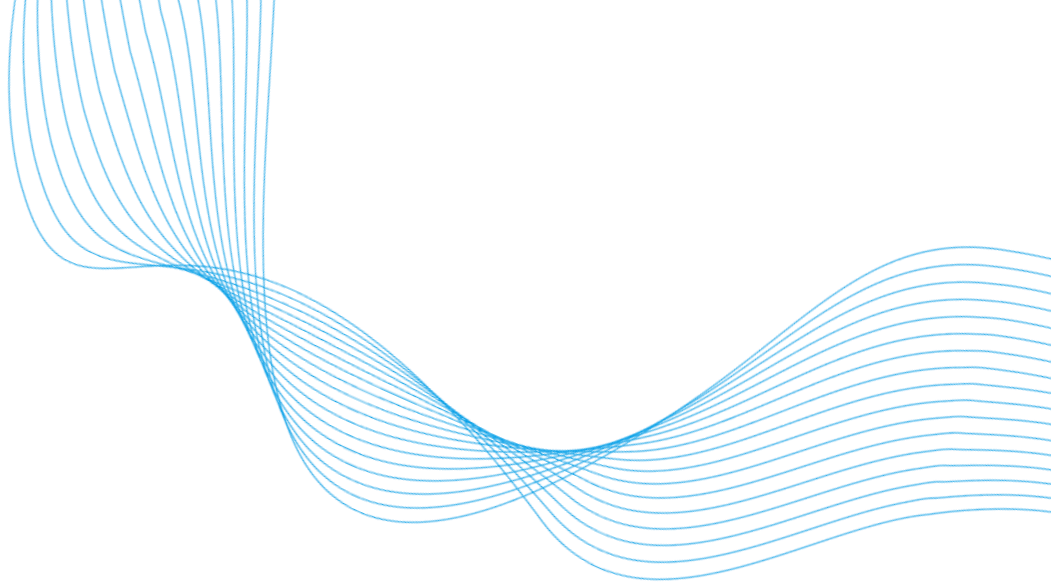
Agenda

Combining Python and C++ using pybind11

Why and how?

Case study: MurTree project

Tutorial

    Building a Python wrapper for C++ code

Introduction to CMake

What is CMake and how is it used?

Tutorial

    Using CMake to build an application

Combining Python and C++

Motivation:

- Make C++ libraries available in Python (wrappers)

- Increase the performance of Python

Combining Python and C++

Motivation:

- Make C++ libraries available in Python (wrappers)

- Increase the performance of Python code

fast but unfriendly

Example copied from the CppCon 2017 talk
"Pybind11 - Python on Steroids" by Jan Plhak

Combining Python and C++

Motivation:

- Make C++ libraries available in Python (wrappers)

- Increase the performance of Python code

fast but unfriendly

friendly but slow

CODING SESSION | 14th APRIL 2023

Example copied from the CppCon 2017 talk
"Pybind11 - Python on Steroids" by Jan Plhak

Combining Python and C++

Motivation:

- Make C++ libraries available in Python (wrappers)
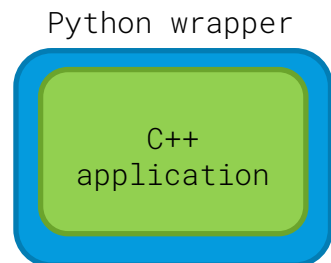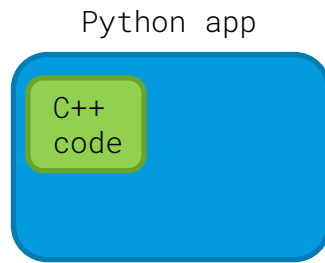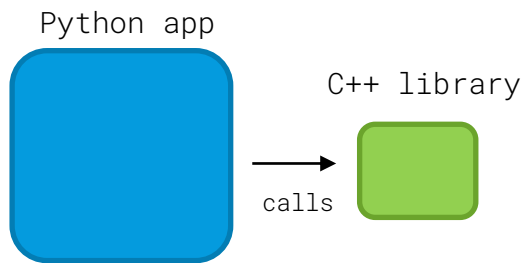
- Increase the performance of Python

fast but unfriendly

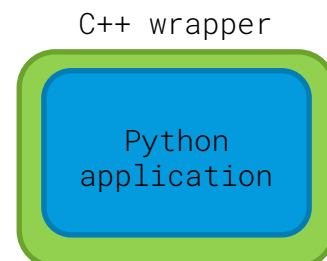friendly but slow

fast and friendly

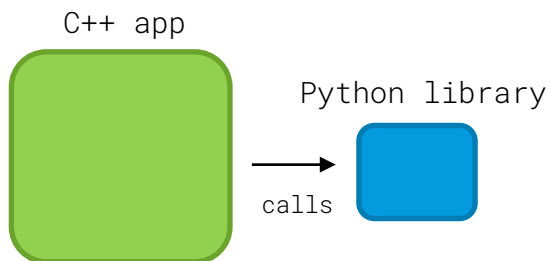Example copied from the CppCon 2017 talk
"Pybind11 - Python on Steroids" by Jan Plhak

Combining Python and C++

Multiple ways to do it

Python app → C++ library

calls
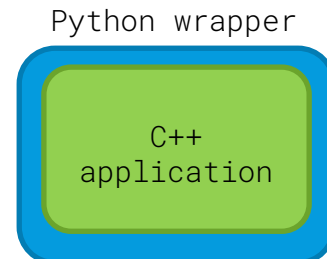
Python app
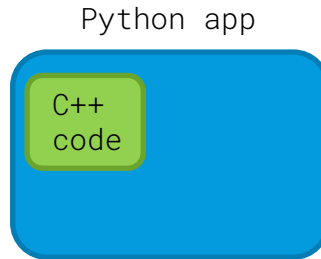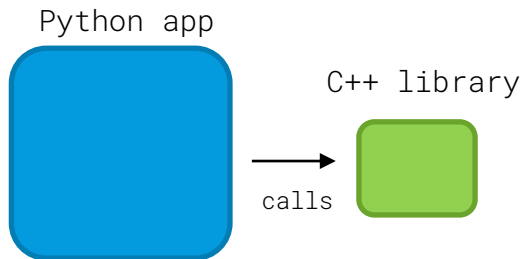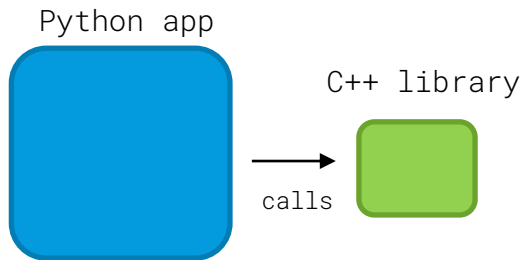
C++ code

Python wrapper

C++ application

Combining Python and C++

Multiple ways to do it

Combining Python and C++
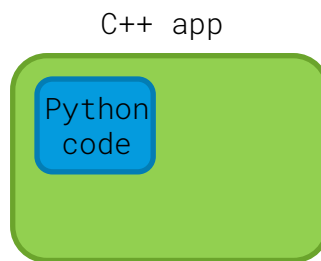
Multiple ways to do it

Python app

C++ library

calls

Extending Python with C++

Python app

C++ code

Python wrapper

C++ application

C++ app

Python library

calls

C++ app

Python code

C++ wrapper

Python application
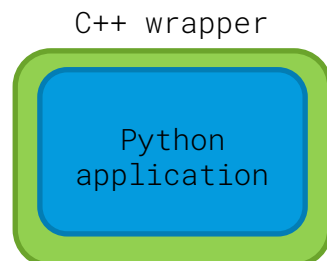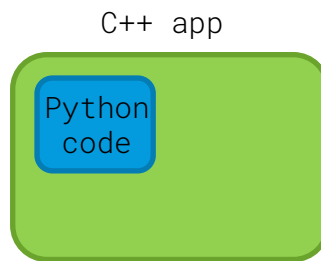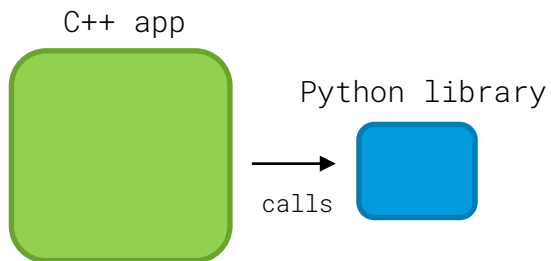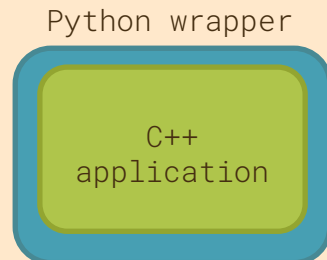
Extending Python with C++

Various tools to do the job

https://stromberg.dnsalias.org/~strombrg/speeding-python/

ctypes

Cython

SWIG

CFFI

Boost.
Python

pybind11

https://pybind11.readthedocs.io/
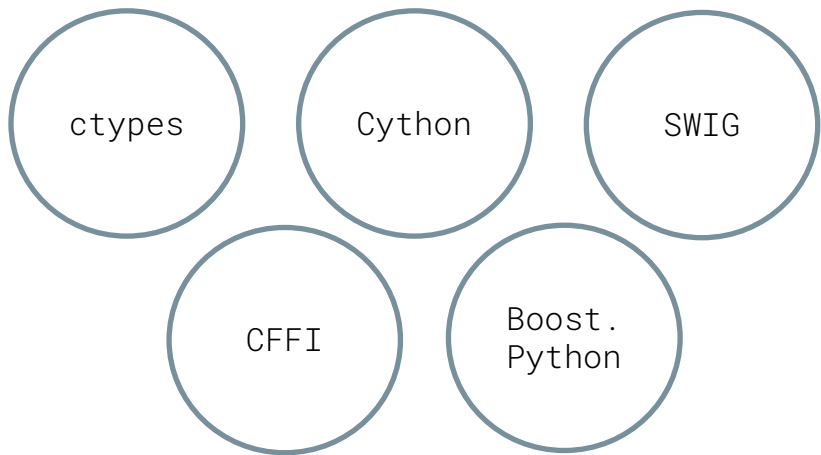
Extending Python with C++

pybind11

- Lightweight library that creates Python bindings for C++ code

- Exposes C++ types in Python and viceversa

- Compatible with modern C++ language features

- Project example: KpLib
  https://gitlab.com/muellergroup/kplib

https://tech.blueyonder.com/python-calling-c++/

Case Study: MurTree

Project description

- Researcher developed a C++ application. He wants his application to be available for Python users. The DCC builds a Python wrapper for MurTree using pybind11.

- Things in our favour:

  - codebase is small

  - simple architecture

  - no dependencies

https://github.com/MurTree/murtree

Case Study: MurTree

Overview

MurTree GitHub organization repositories
- murtree — original c++ code
- pymurtree — bindings only (uses murtree repo)
- murtree-data — input datasets

pymurtree

- pyproject.toml — registers project dependency on pybind11, setuptools and gitpython
- setup.py — clones murtree repo using gitpython and adds bindings using pybind11 function
- src/main.cpp — declares python bindings for cpp code

Let's take a look at the code 😃

Pybind11 Mini Tutorial

Creating a Python module from existing C++ code

Goal: run the following python code where the add function is implemented in C++

```
>>import pymymath
>>pymymath.add(1,3)
>>4.0
```


https://github.com/yiquintero/dcc-codingsession-pybind11

Pybind11 Mini Tutorial

Adding a new feature to mymath and pymymath

Goal: Add a new function to the mymath library and expose it to Python users by adding it to the pymymath module

👉 Don't forget to uninstall the previous pymymath module using "pip uninstall pymymath"

```
>>import pymymath
>>pymymath.subtract(3,1)
>>2.0
```


https://github.com/yiquintero/dcc-codingsession-pybind11

CODING SESSION | 14th APRIL 2023

CMake

One tool to build them all

Builds, tests and packages software.

Features:

- Compatibility with many programming languages and compilers

- Cross-platform and open source

- Multiple build configurations possible with a single CMakeLists file

- Well established tool

  ○ Many libraries come with a CMakeLists file
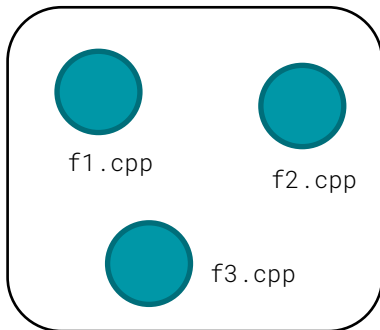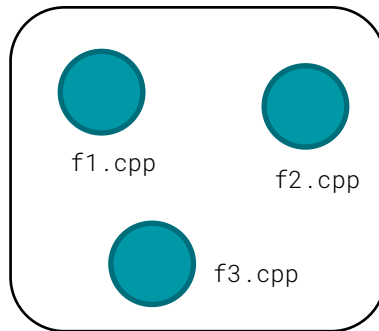
  ○ Supported by many IDEs and workflows

https://cmake.org/

CMake

Cmake is not a build system itself;
                              it generates another system's build files.
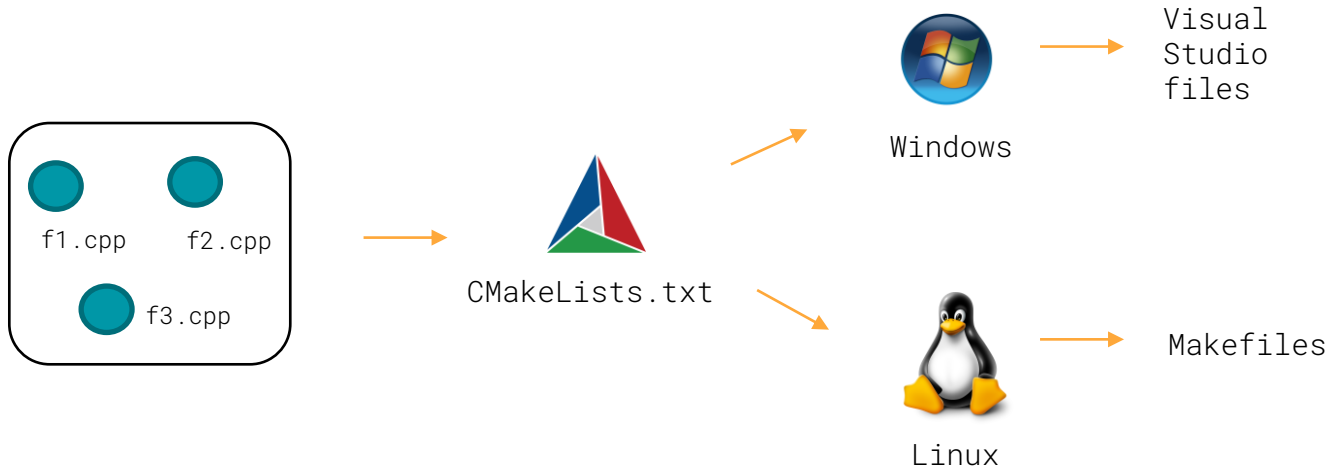
Windows – Visual Studio files

Linux - Makefiles

f1.cpp

f2.cpp

f3.cpp

f1.cpp

f2.cpp

f3.cpp

CMake

Cmake is not a build system itself;
it generates another system's build files.

f1.cpp   f2.cpp
f3.cpp

CMakeLists.txt

Windows → Visual Studio files

Linux → Makefiles

CMake Mini Tutorial

Build a C++ application using CMake

https://github.com/yiquintero/dcc-codingsession-pybind11