

# 1. Introduction

Pieter P

## 1.1. Goals and non-goals

---

The goal of this guide is to get to an ergonomic workflow for developing scientific software using C++. It is not designed to be a tutorial of the C++ language itself, but an overview of useful tools (package managers, build systems, linters, static analysis, run-time sanitizers, debuggers, testing frameworks), recommendations on how to structure a (scientific) C++ project, guidelines for interoperability with other popular languages, and so on.

## 1.2. Related resources

---

- [cppreference](#): the recommended language and standard library reference
  - [learncpp.com](#): a solid, up-to-date tutorial for the C++ language itself
  - [C++ Core Guidelines](#): an extensive set of guidelines for using C++ well
  - [C++ 23 Standard \(draft\)](#): draft of the official C++ standard, as a big PDF
-