Scientific Software Development with Python

Part 1 — Recap

CHALMERS
UNIVERSITY OF TECHNOLOGY

Simon Pfreundschuh
Department of Space, Earth and Environment

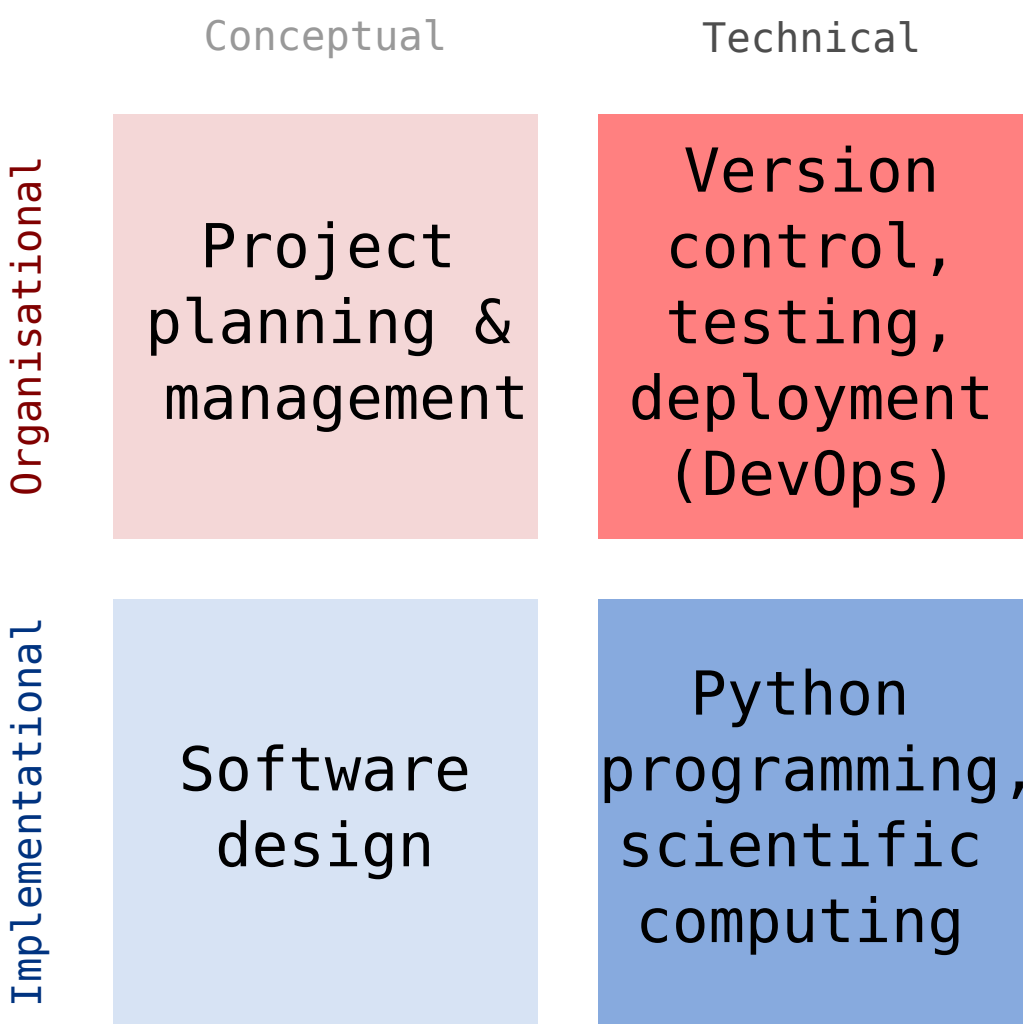
Agenda

CHALMERS
UNIVERSITY OF TECHNOLOGY

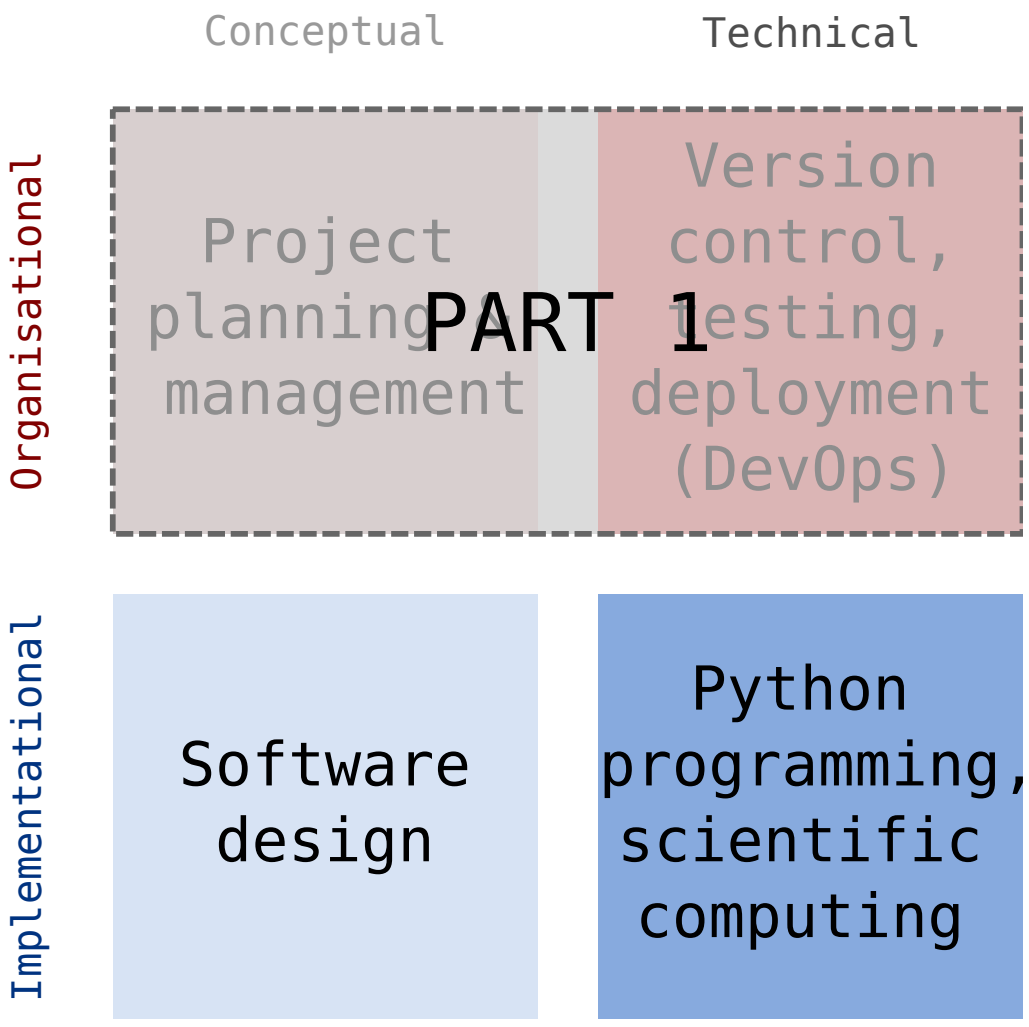
1. Recap

2. Preview

What we have learned so far



What we have learned so far



Agile project management

1. Embrace and acknowledge uncertainty:
   - We can't plan everything ahead, requirements will likely change
2. Iterative development
   - Release early, release often
   - Sprint-based process model

We should make use of Agile practices and tools (DevOps) to make Agile work.

Version control

- Using git and GitHub

Testing

- Unit testing with Pytest

Packaging

- Defining packages
- Publishing packages

Documentation

- Writing documentation with Sphinx

Continuous integration

- Automating DevOps with GitHub

Other useful tools

- Code formatter: black
- Linter: pylint
- Test coverage: Coverage.py

A typical development workflow

- Central repository contains **working** development version
- Developers implement features in personal forks
- When a feature is ready, it developer makes pull request to central repository
- Good practice: Request code review from other developer to ensure that code satisfies quality requirements

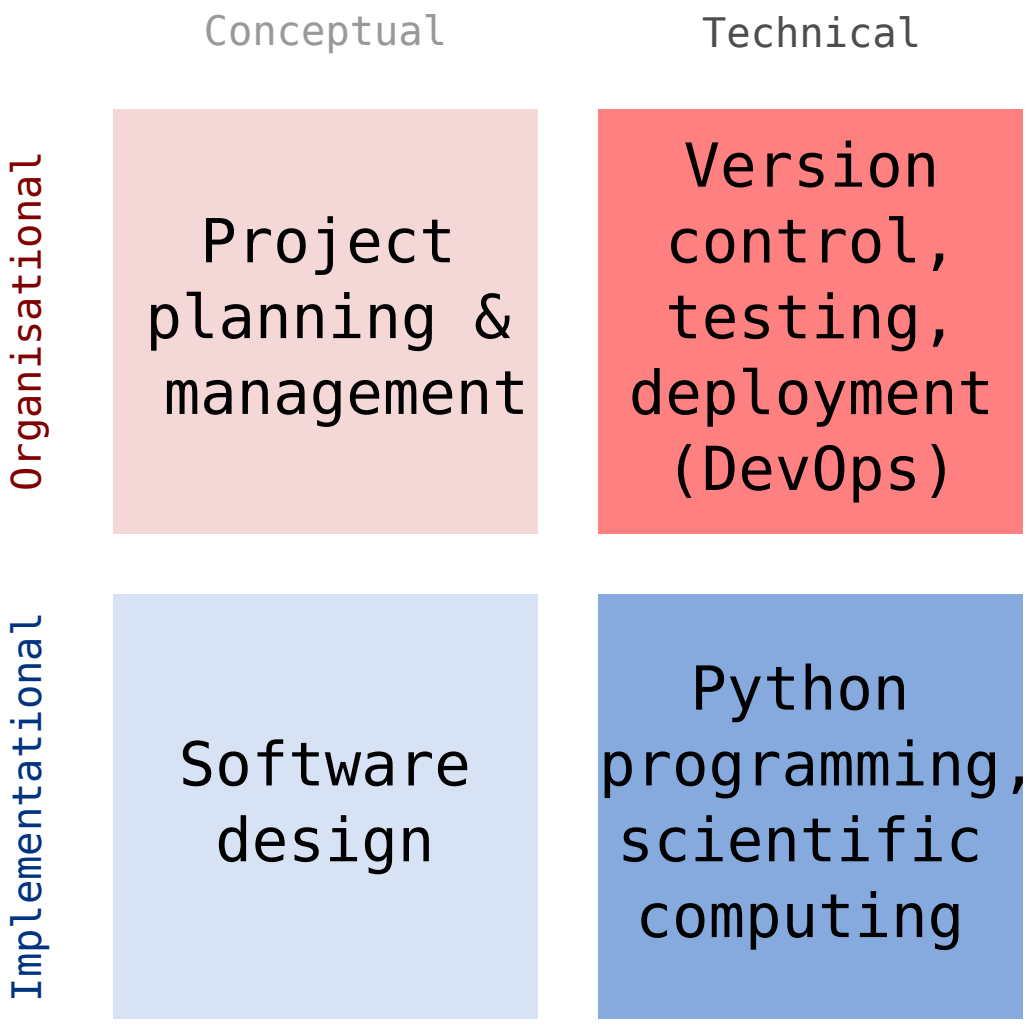
Definition of done

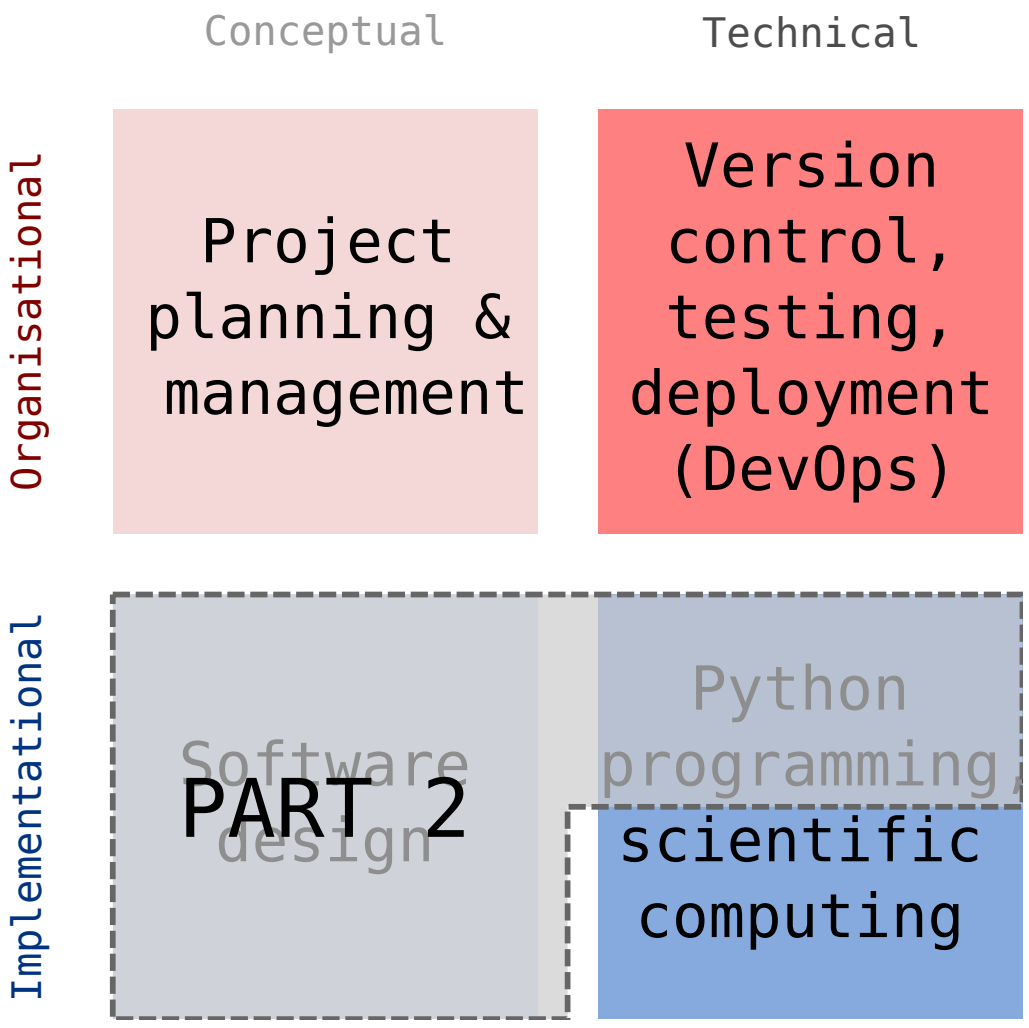
- What is required for a feature to be considered done?
- Should be agreed upon by the team
- Typical requirements:
  - Unit tests
  - Acceptance test (example) for user story
  - Documentation
  - No decrease in test coverage

Agenda

1. Recap

2. Preview


Conceptual
Technical
Organisational
Implementational
Project planning & management
Version control, testing, deployment (DevOps)
Software design
Python programming, scientific computing


Conceptual
Technical
Organisational
Implementational
Project planning & management
Version control, testing, deployment (DevOps)
Software design
Python programming
scientific computing
PART 2

Topics

- Object orient programming with Python (Lecture 1 and 2)
- Python standard library (Lecture 3)
- Python recipes (Lecture 4)