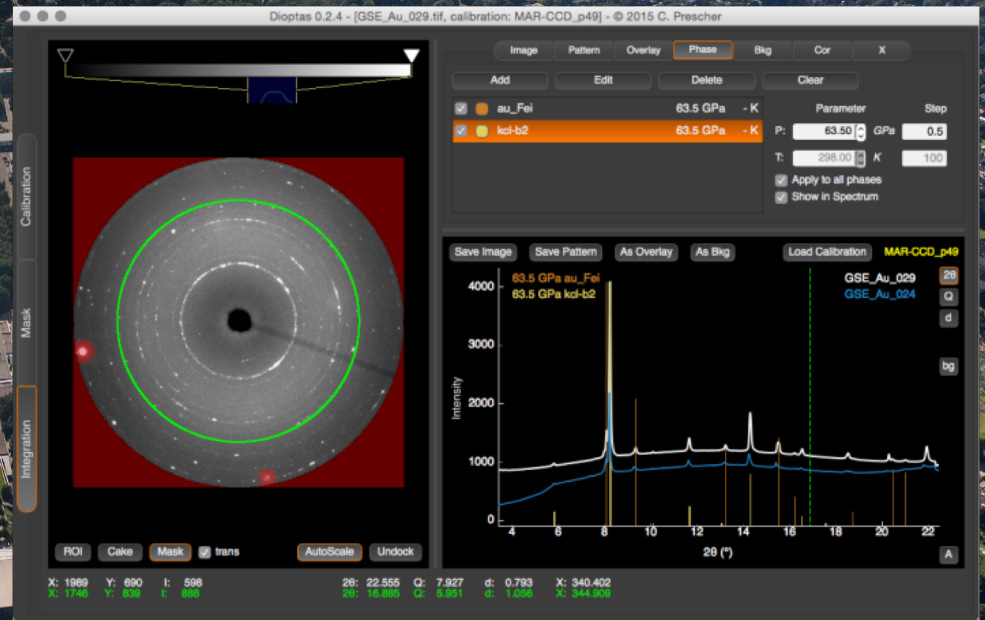


# Dioptras

2D XRD GUI  
software  
utilizing the  
power of pyFAI



Clemens Prescher  
Grenoble, 07.02.2019

# Introduction

## General

### What is Dioptas?

- Python based GUI-software for 2D XRD integration and exploration
- focusses on usability and speed
- focusses on immediate feedback during experiment

### What is Dioptas not?

- no full data analysis suite
- no Rietveld/LeBail refinements
- no SAXS analysis
- no PDF analysis

**Existing software is  
already good in doing  
that!**

# Introduction

## Why another program for 2D XRD

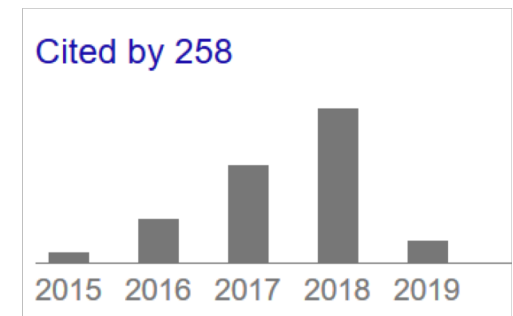
- Almost all XRD beamlines are equipped with 2D XRD detectors
- Readout time of the detectors becomes faster and faster
- Essentially it is faster to detect data than to actually view the integrated patterns
- Previous programs are very slow in integrating 2D XRD images
- Previous programs focus on data analysis and capabilities, but not data exploration

# Introduction

## Statistics

### Where is Dioptas used?

- High pressure X-ray diffraction beamlines:
  - **APS:** HPCAT (Sector 16), CARS (Sectors 13-15), DCS (Sector 35)
  - **ESRF:** ID09, ID27
  - **PETRA 3:** P02.2
  - **ALS:** 12.2.2
  - **LCLS:** MEC
- Many other general purpose diffraction beamlines:
  - e.g. APS, Sector 11; ESRF, ID11
- Many in house laboratories



*Google Scholar*

# Library Infrastructure

## Main libraries used by DIOPTAS

- NumPy
- SciPy
- PyQt4/PyQt5/PySide
- pyqtgraph
- scikit-image
- fabio → ESRF ([github.com/silx-kit/fabio](https://github.com/silx-kit/fabio))
- pyFAI → ESRF ([github.com/silx-kit/pyFAI](https://github.com/silx-kit/pyFAI))
- Silx → ESRF ([github.com/silx-kit/pyFAI](https://github.com/silx-kit/pyFAI))



# Features

## Dioplas is a GUI for:

- fast image interaction and intensity scaling (thanks to *pyqtgraph*)
- Detector Geometry Calibration
- Mask creation
- Image integration
- Very fast image integration (thanks to *pyFAI*)
- Batch processing and auto-processing of files
- Comparing different pattern using overlays
- Adding lines of different phases and adjusting their pressure and temperature
- Automatic pattern background subtraction
- Different sample and detector absorption corrections

# Distribution

## Code

- via [github.com/dioptas/dioptas](https://github.com/dioptas/dioptas)
- As conda package: `conda install -dioptas -c cprescher`
- current other contributors:
  - Eran Greenberg (GSECARS/University of Chicago)
  - Maxim Bykov (Petra III, Hamburg)
  - Valentin Valls (ESRF, Grenoble)

## Executables

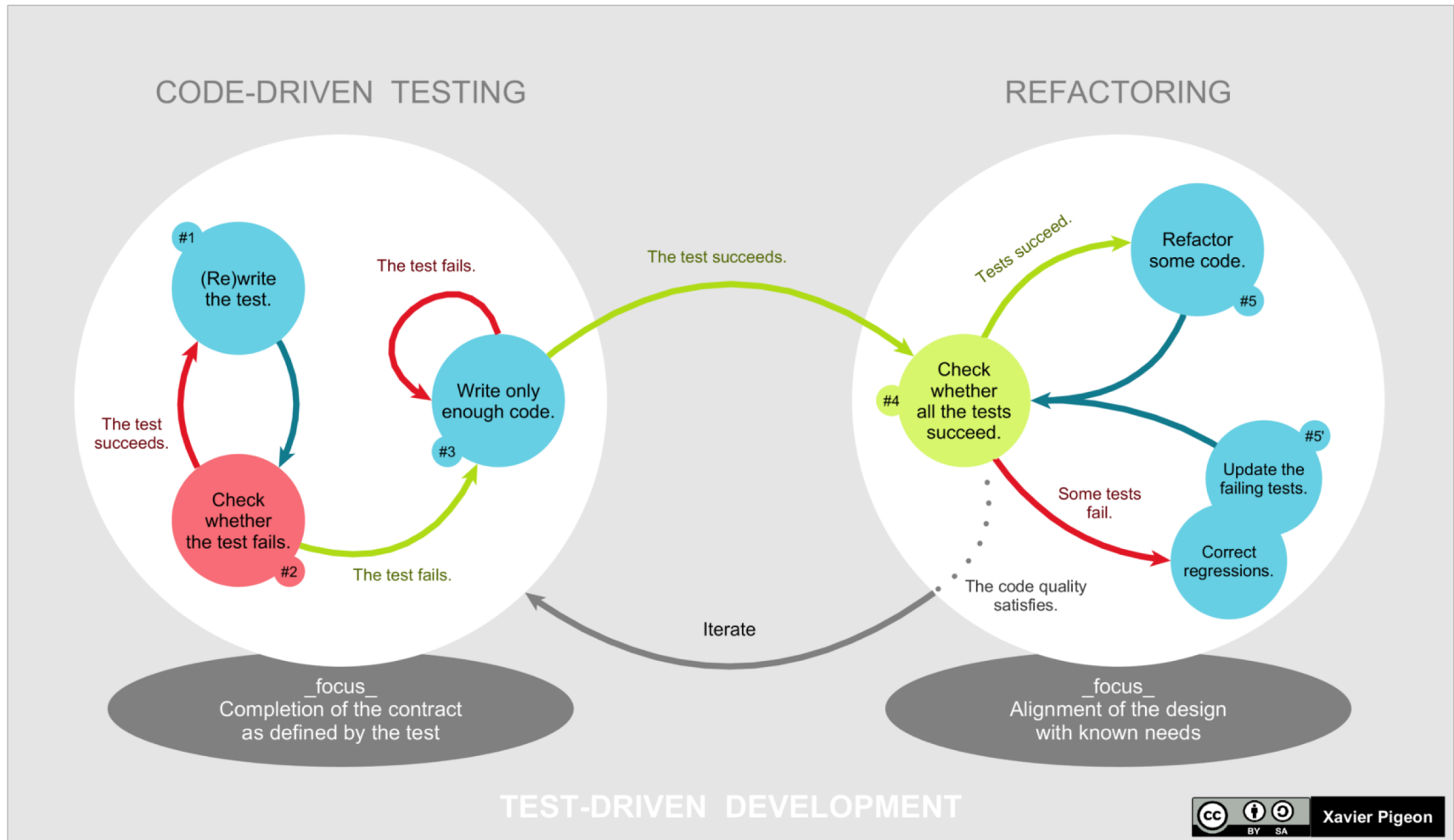
- Available for:
  - Windows 7,8, 10
  - Mac OS X
  - Linux Debian
- via:
  - [clemensprescher.com/programs/dioptas](http://clemensprescher.com/programs/dioptas)

# Live Demonstration of Dioptras



# Programming Scheme

## Test-driven development



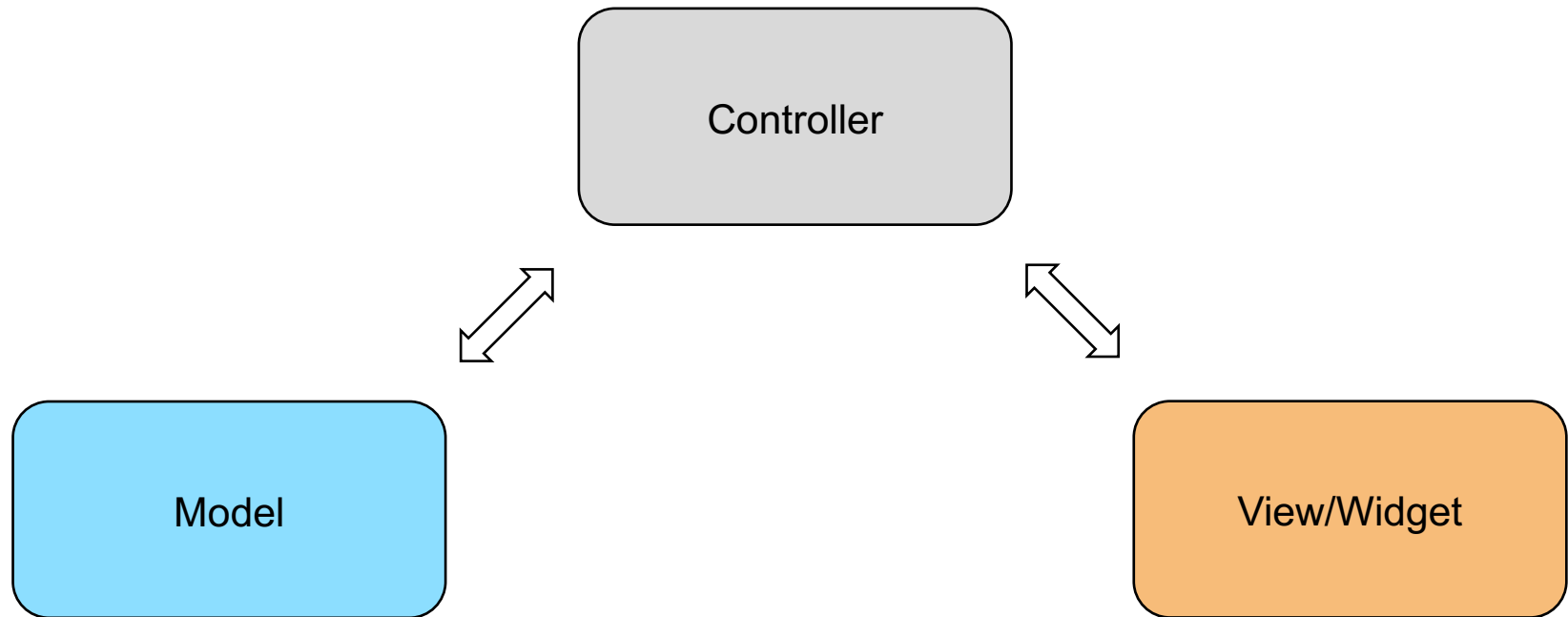
# Programming Scheme

## Test-driven development (TDD)

- TDD provides a way of thinking what you actually really need
- TDD facilitates thinking about nice interfaces
- For Dioptas the process is usually:
  - Write tests for new capability, implement low-level in the model
  - Implement GUI for this capabilities
  - Write tests for integration and connection of controller to GUI and model
- I mock the the user interaction by using the capabilities of pyqt5 and the mock library
  - Thus, the GUI is tested with actual user interaction

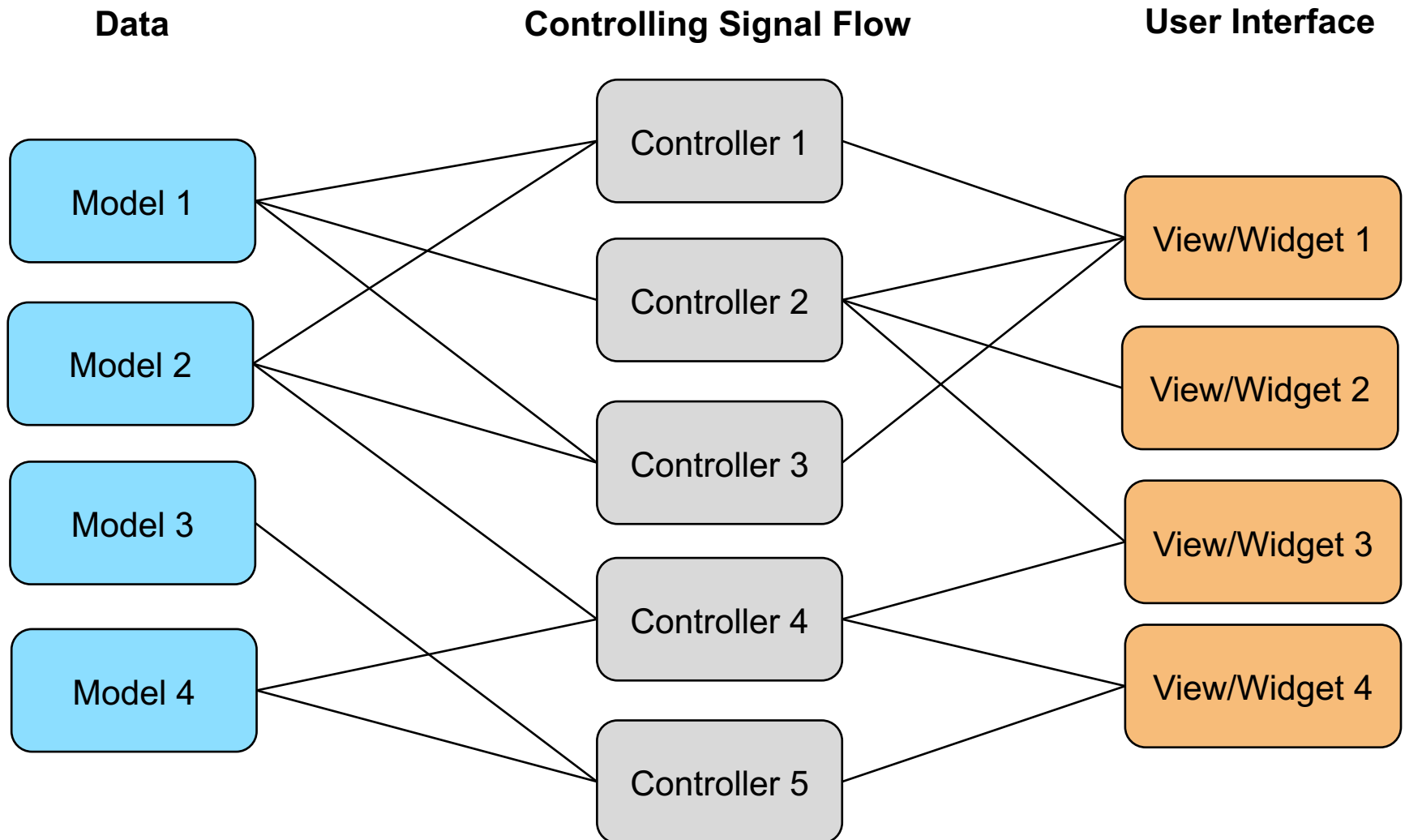
# Program architecture

Everything is organized in a Model-View-Controller Scheme



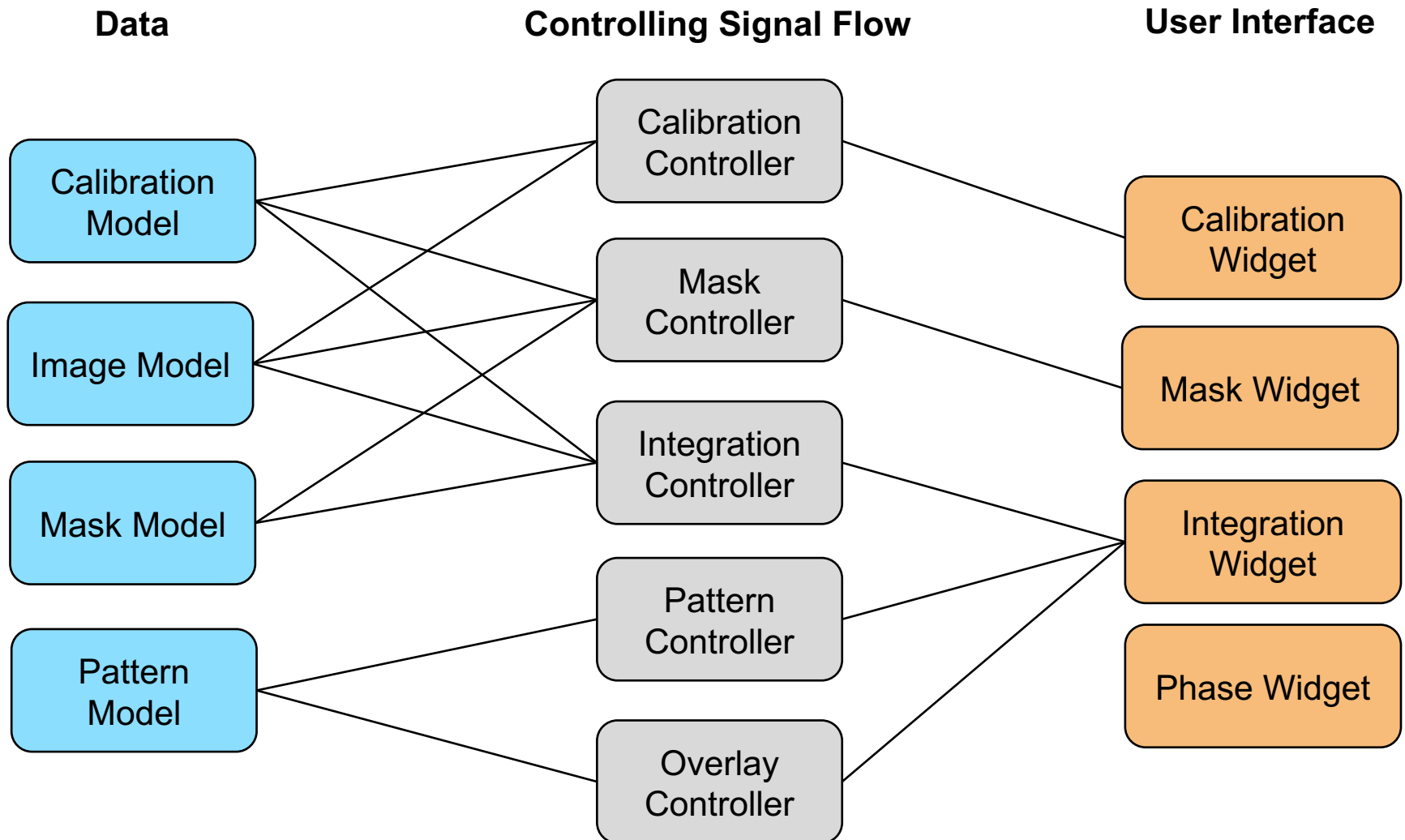
# Program architecture

Too Complex to organize everything in one triad:



# Program architecture

## How MVC is implemented in Dioplas



**Thank you for the  
attention**

## Contact

**DESY.** Deutsches  
Elektronen-Synchrotron

[www.desy.de](http://www.desy.de)

Clemens Prescher

FS-PS

[clemens.prescher@desy.de](mailto:clemens.prescher@desy.de)

+49 (0)40 8998 5302