ADOPTING PYQT FOR BEAM INSTRUMENTATION GUI DEVELOPMENT AT CERN

S. Zanzottera (CERN), S. Jensen (CERN), S. Jackson (CERN)

ICALEPCS 2021

Abstract

As Java GUI toolkits become deprecated, the Beam Instrumentation (BI) group at CERN has investigated alternatives and selected PyQt as one of the suitable technologies for future GUIs, in accordance with the paper presented at ICALEPCS19.

This paper presents tools created, or adapted, to seamlessly integrate future PyQt GUI development alongside current Java oriented workflows and the controls environment. This includes (a) creating a project template and a GUI management tool to ease and standardize our development process, (b) rewriting our previously Java-centric Expert GUI Launcher to be language-agnostic and (c) porting a selection of operational GUIs from Java to PyQt, to test the feasibility of the development process and identify bottlenecks.

To conclude, the challenges we anticipate for the BI GUI developer community in adopting this new technology are also discussed.

bipy-gui-manager

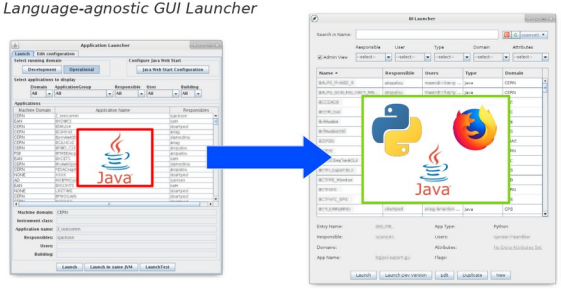
- Asks for:
  - project name
  - author name
  - author email
  - etc…
- Creates:
  - A template project in the desired location
  - A standardized, precompiled README
  - A GitLab repository
  - A dedicated virtual environment
  - A template for Sphinx-based documentation
  - A Continuous Integration pipeline
  - A Continuous Deployment pipeline
  - Continuous Documentation pipeline

Developer effort to setup a standard project following all recommended best practices is now close to zero.

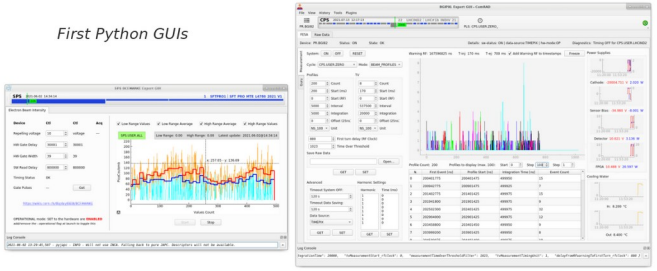
ComRAD

- Zero-code GUI tool for Rapid Application Development (RAD) using PyQt as underlying GUI framework.
- Does not limit the developers from using the full capabilities of PyQt when necessary.
- Designed to allow custom widgets.
- Generated apps are regular PyQt apps (after an export step).

A Language-agnostic GUI Launcher



First Python GUIs



Abstract

As Java GUI toolkits become deprecated, the Beam Instrumentation (BI) group at CERN has investigated alternatives and selected PyQt as one of the suitable technologies for future GUIs, in accordance with the paper presented at ICALEPCS19.

This paper presents tools created, or adapted, to seamlessly integrate future PyQt GUI development alongside current Java oriented workflows and the controls environment. This includes (a) creating a project template and a GUI management tool to ease and standardize our development process, (b) rewriting our previously Java-centric Expert GUI Launcher to be language-agnostic and (c) porting a selection of operational GUIs from Java to PyQt, to test the feasibility of the development process and identify bottlenecks.

To conclude, the challenges we anticipate for the BI GUI developer community in adopting this new technology are also discussed.

bipy-gui-manager

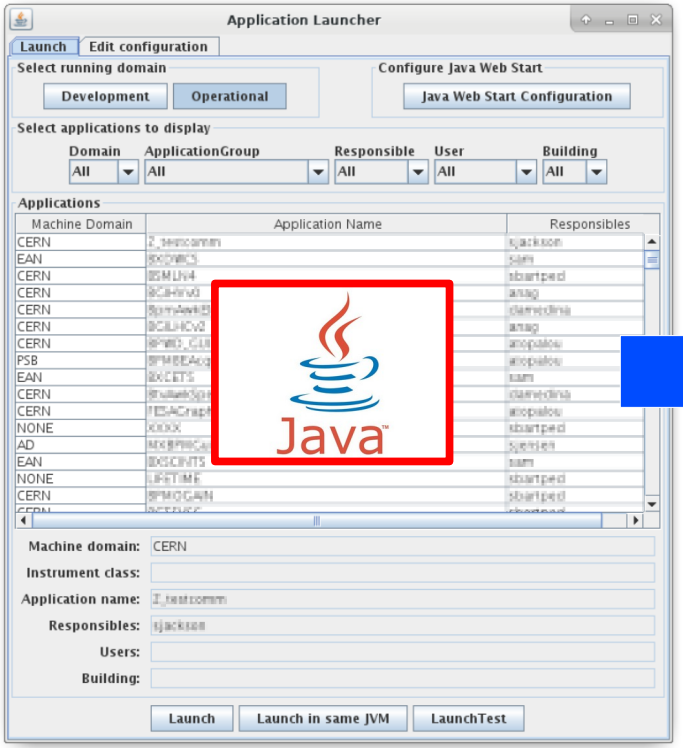
- Asks for:
  - project name
  - author name
  - author email
  - etc…
- Creates:
  - A template project in the desired location
  - A standardized, precompiled README
  - A GitLab repository
  - A dedicated virtual environment
  - A template for Sphinx-based documentation
  - A Continuous Integration pipeline
  - A Continuous Deployment pipeline
  - Continuous Documentation pipeline

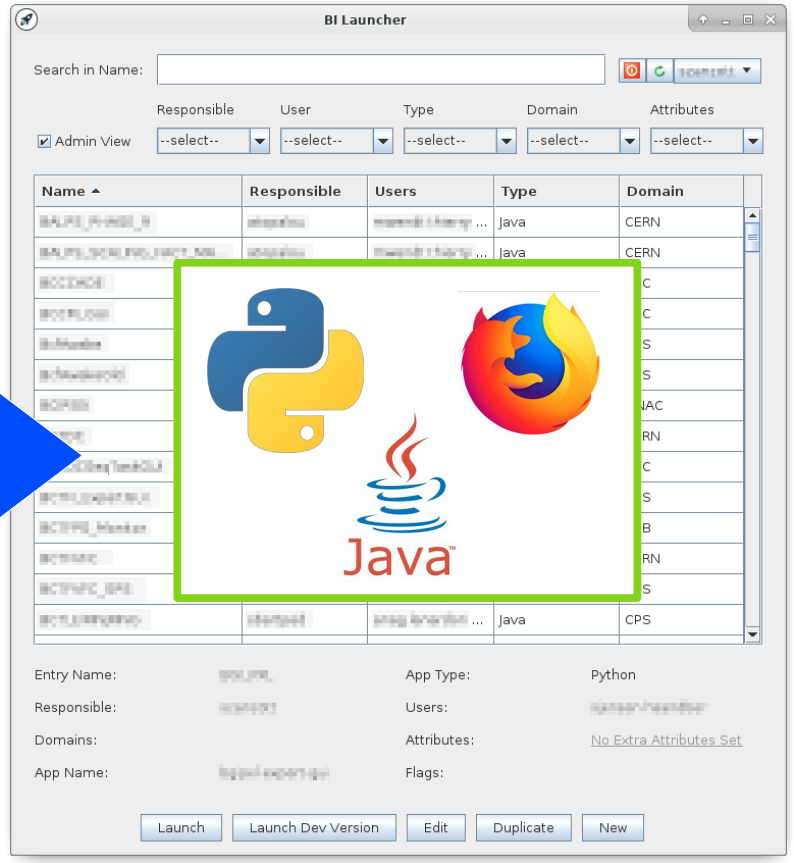
Developer effort to setup a standard project following all recommended best practices is now close to zero.

ComRAD

- Zero-code GUI tool for Rapid Application Development (RAD) using PyQt as underlying GUI framework.
- Does not limit the developers from using the full capabilities of PyQt when necessary.
- Designed to allow custom widgets.
- Generated apps are regular PyQt apps (after an export step).

A Language-agnostic GUI Launcher





First Python GUIs