PYTHONDAQ – A PYTHON BASED MEASUREMENT DATA ACQUISITION AND PROCESSING SOFTWARE

Daniel Jäger
Technical University of Munich

ABSTRACT

This paper introduces PythonDAQ, an open-source Python package for measurement data acquisition, visualization, storage, and post-processing. The code acquires measurement data from any sensor with digital data output, performs online calculations, and stores the measured and computed data. A client for live data visualization and tools for post-processing are also contained in the software package. After the introduction of the code, comparisons between the PythonDAQ and the data acquisition software of a major aero-engine OEM and other commercial DAQ solutions are made.

INTRODUCTION

Measurement setups for turbomachinery test rigs generally count a high number of channels, whereas, in modern setups, most of them come from sensors with digital data output. The measurement data is then transferred to computers for further processing. As raw sensor data is often not directly human-interpretable, characteristic turbomachinery quantities are computed live from the sensor readings. The data is stored and displayed to the user as the last step.

To fulfill these tasks, the designer of a test rig has the following two options: Either use a specially designed commercial data acquisition software, which is generally expensive, or implement its own code. The latter is not a trivial task resulting in a large personal workload.

With this paper, a shortcut to the development process is provided. PythonDAQ is an open-source software package that intends to cover the whole process from data acquisition and storage until post-processing. The software is designed in a modular way, such that different measurement setups can be built from an existing catalog of classes, avoiding starting from scratch. Sensors that are not part of the package or online calculations for a specific purpose can be implemented from existing base classes.

RESULTS AND DISCUSSION

In this paper, the software architecture of PythonDAQ and its capabilities are discussed. The software is designed as a server-client architecture where the server handles the data acquisition, online computations, and data storage. The data is then provided to the local network via OPC UA. The client is responsible for visualizing the live data to the user connecting to the OPC port of the server. Figure 1 illustrates the software layout of PythonDAQ.

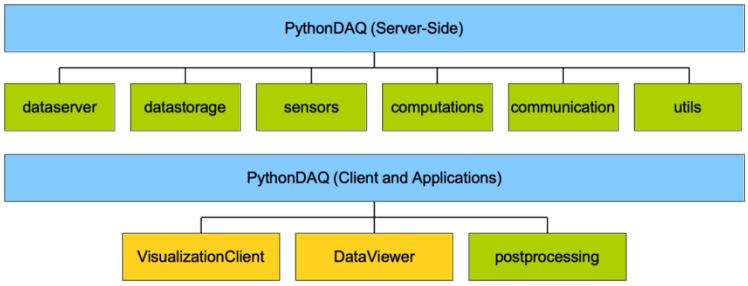

Figure 1. Software layout of the PythonDAQ package

The capabilities and limitations of the current implementation are compared against the professional industry solution at their compressor test rigs. It is found that PythonDAQ covers the basic needs for being used in a turbomachinery test rig. However, various features of the commercially used code are still missing, thus being in an early stage of development.

Furthermore, the software is tested in real operation at the compressor test rig FRANCC of the chair of turbomachinery and flight propulsion of the Technical University of Munich. The test shows good coverage for the needs of this low-speed test rig and reliable and stable operation.