

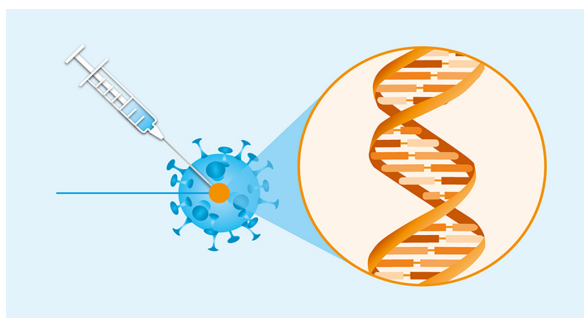


Enhancing Developer Experience in Near Real-Time Scientific Data Processing: The **AsapoWorker** Library

Diana Rueda, Tim Schoof, Mikhail Karnevskiy, Aleksandra Tolstikova, Parthasarathy Tirumalai,
Marc-Olivier Andrez, Anton Barty
Deutsches Elektronen-Synchrotron DESY,
Hamburg, Germany

Experiments at Synchrotrons

Synchrotron light sources generate intense, high-energy X-rays that enable molecular-level analysis of materials, **revolutionizing research** fields from drug discovery to materials science. By revealing the structure and properties of complex biological systems, synchrotrons provide **critical insights that drive breakthroughs** in pharmaceutical development and medical treatments.



Medical research as one the applications *

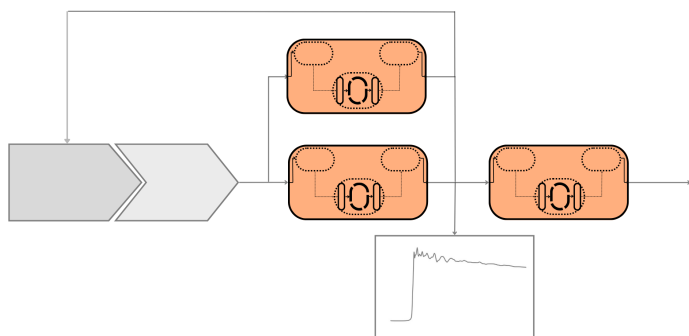
Experiments at these facilities generate **high-volume data** at speeds up to 2kHz with **large payloads**, requiring **near-real-time processing for experiment steering decisions**. A streaming technology is needed to efficiently transfer these large messages to the HPC cluster for parallel processing and **return results to users with minimal latency**.

High-Performance Processing Pipelines

To address these challenges, DESY has developed **Asapo**, a **distributed streaming framework** designed to handle very large data payloads. While this framework provides fundamental building blocks, their complexity necessitated the creation of **AsapoWorker**, a higher-level abstraction that **simplifies the construction** of real-time data processing pipelines.

We encourage you to visit the poster:

#39. Framework for Distributed Near Real-Time Data Processing Pipelines.

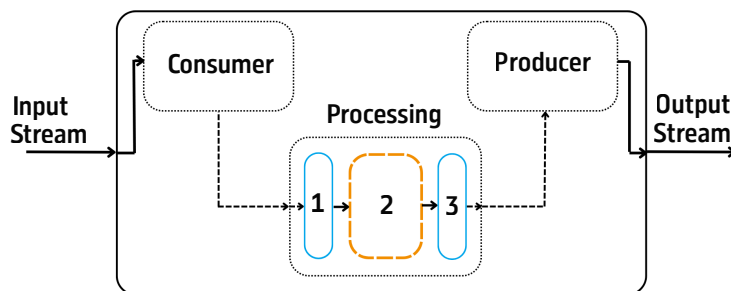


Conceptual diagram of an analysis pipeline including AsapoWorkers

AsapoWorker aims to **encapsulate diverse analytical processes** as standardized worker components that can be seamlessly integrated into processing pipelines. These workers can wrap various types of analyses, including **data preprocessing**, **scientific algorithms**, and **machine learning inference** tasks.

The AsapoWorker Library

The AsapoWorker library originated from a collection of scripts that were repeatedly written to run processing pipelines on the Asapo streaming platform. Distributed data analysis presents unique challenges that must be addressed by workers to **ensure reliable data processing**.



Conceptual diagram of an AsapoWorker

The conceptual diagram above illustrates the **main logical components** of the AsapoWorker. Components 1 and 3 are data handlers, with 1 performing **deserialization** and 3 performing **serialization**. Component 2 represents the actual **analysis method in Python**, which can be adapted from conventional analysis approaches that process one message (image or scan) at a time.

Main Features

Some of the main benefits of using the AsapoWorker library include:

- **Error Handling:** Automatic detection and handling of processing failures with fail-fast behavior, preventing corrupted results.
- **Stream Management:** Automatically process new streams as they become available. Intelligent logic to detect stream completion and shutdown workers, preventing infinite blocking on receiver failures.
- **Configurable input parameters:** Easily customize parameters to run multiple workers with different configurations for result comparison.
- **Python code integration:** Seamlessly integrate existing Python analysis code into the streaming pipeline system.

Planned improvements

To promote adoption of the AsapoWorker library, we are considering the following improvements:

- + **Enhanced Documentation:** Simplify onboarding with comprehensive guides and practical examples that users can build upon.
- + **Configuration File Support:** Replace lengthy command-line arguments with structured configuration files for easier parameter management.
- + **Multiple output streams:** Providing flexibility for specialized use cases.

Question or suggestions? Let's discuss!

* Illustration: DESY

AsapoWorker <https://gitlab.desy.de/fs-sc/asapoworker>

Asapo <https://asapo.pages.desy.de/asapo/>

Xdi Pipeline https://gitlab.desy.de/fs-sc/analysis-pipelines/xdi_pipeline

Questions/comments diana.rueda@desy.de, tim.schoof@desy.de, marc-olivier.andrez@desy.de

We gratefully acknowledge funding from the Helmholtz Association HGF for the ROCK-IT project.

