PaSER powered data streams for real-time processing and feedback

ASMS 2021, ThP 153

Tharan Srikumar[1], Sven Brehmer[2], Vijayaraja Gnanasambandan[3], Marc-Antoine Beauvais[4], Christopher Adams [4], Dennis Trede[2], Robin Park[4]

1) Bruker Ltd., Milton, ON
2) Bruker Daltonics GmbH & Co.KG, Bremen, Germany
3) Bruker Scientific LLC, Billerica, MA
4) Bruker Scientific LLC, San Jose, CA

Introduction

Parallel search engine in real-time or PaSER was developed together with the Yates lab to take advantage of GPU-powered database search. The GPU-powered ProLuCID-4D algorithm can process a large number of MSMS spectra generated by the PASEF process on the timsTOF platform, while utilizing all four dimensions – retention time, CCS value, m/z and fragment spectra – to increase the confidence in each identification. PaSER has now been extended into a platform that can integrate 3rd party tools enabling these tools to perform real-time analysis with generally minor adaption of their existing code. To achieve this, PaSER utilizes the concept of streams and stream-processors to realize fully customizable real-time processing workflows including on-the fly decision making based on the data being generated.

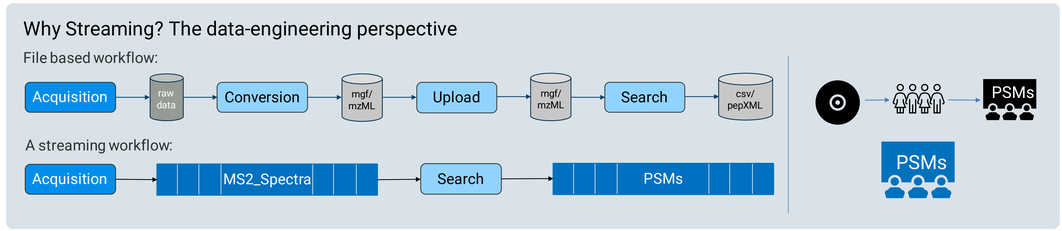

Fig.1 Why Streaming? Let's look at a use case of acquiring a LC-MS/MS data and needing to identify the compound/peptides in this file. A) In most current workflows one would acquire a file, the file may need to be converted before uploading to a processing computer\server. The file would then be processed (search results generated) and results files would be available. This process effectively results in data processing being 'chunked' by different files. B) In a streaming workflow, the data to be processed (MS2 spectra) is streamed as its generated. A search process would detect data in the stream and process it and output PSMs to the stream. Additional process engines could be added to extend the processing workflow. C) The difference between the two can be illustrated via an analogy of the 'old' Netflix model of DVD rentals to see the movie you want, vs the current Netflix streaming model. The result is an enjoyable movie with family or friends, but the streaming model affords greater flexibility and ease of use. Similarly, streaming workflows for LC-MS/MS data processing enable: i) less data conversions with transparent and open access to data using schemas, ii) makes real-time analysis possible and iii) provides better connection and communication between acquisition and analysis software!

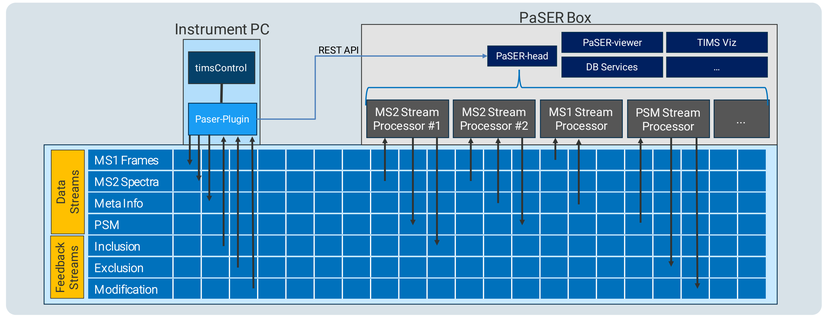

Fig. 2. Stream in PaSER. A plugin on the instrument PC directly access data from timsControl (the acquisition software). The plugin outputs data into appropriate streams, while monitoring the feedback streams for any instructions. It also communicates via REST APIs with PaSER head. PaSER head controls the user defined workflows and activates the requested stream processors. Each stream processor can interact with one or more streams. Feedback calls, such as inclusion\exclusion of a given precursor or other method modifications, can supplied via the feedback streams. This allows for highly asynchronous flexible orchestration of the processing pipeline, while safeguarding data acquisition on the instrument.

Results

Streaming based data processing with PaSER

During acquisition on the timsTOF platform, the MS1 frames and MS/MS spectra are streamed to PaSER via a dedicated private network connection. MS1 frames and MS/MS spectra are published to separate streams, allowing each stream of data to be processed independently. Related information, for example, the m/z, mobility, charge state and retention time for a given precursor, are also published to the MS/MS stream. In PaSER, various stream processors can be invoked depending on the user defined workflow being utilized. Each stream processor can read data from a stream and output data to another stream. For example, ProLuCID-4D reads the MS/MS stream for spectra, assigns putative peptide identifications, and outputs these identifications and scores (XCorr, deltaCN and TIMScore) to the PSM stream. Multiple stream processors can interact with the one stream and a stream processor can interact with multiple streams. A user definable workflow dictates which stream processor will be utilized, while boundary conditions are managed by PaSER. Feedback to the timsTOF can also be realized via a dedicated feedback streams. Here we show the utility of stream processing supported via micro-services for proteomics data processing.

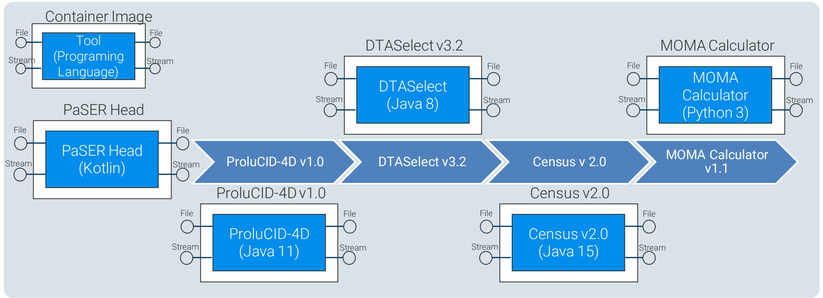

Fig. 3. Containers for faster development, flexible deployment and reproducible workflows. PaSER utilizes a microservices architecture. Each service is containerized, meaning that each stream processor is also containerized. Individual containers allow each stream processor to be developed, tested and deployed independently. This allows 3rd party developers (and collaborators) the freedom to choose their development environment. For routine users, containers ensure reproducible workflows. Every request to use a workflow will initiate the same containers. If there is an update to an algorithm, it would be in a new container (to which the workflow could be updated to). This gives users the flexibility to update tools and take advantage of new features as they become available, without sacrificing the reproducibility of longer-term projects. It also allows for great transparency and shareability between PaSER users and labs.

Conclusions

- Stream processing allows for a more transparent and immediately accessible pipeline of data for processing.
- Containerized stream engines, allow algorithms developers to choose the programming language that is optimized for the task and allows flexible deployment strategies.
- Containerized micro-services allows users to realize consistently reproducible workflows without sacrificing access to new features.

timsTOF Pro