BIOGNOSYS
NEXT GENERATION PROTEOMICS

TECHNICAL NOTE

Large scale DIA data analysis and scalability using Spectronaut™ software

In this technical note you will learn about:

- **Configuring Spectronaut for maximal performance**
- **Scalability of Spectronaut**
- **Properly selecting your computer/workstation for Spectronaut**

You can also send us an inquiry to support@biognosys.ch for a completely configured workstation according to your needs.

Hyper Reaction Monitoring (HRM)
NEXT-GENERATION PROTEOMICS

Introduction

Since the introduction of targeted DIA/SWATH data analysis the technique has moved from a proof of concept towards an established method for high-throughput proteomics data quantification. With this came the shift from small technical benchmarks containing a few runs and small sample-specific libraries, towards large-scale biological experiments spanning over hundrets of runs and libraries covering thousands of proteins. In order to tackle these experiments of ever increasing size, data analysis algorithms and software solutions must adapt and improve their scalability and efficiency when it comes to utilizing the available processing hardware. Here we present the latest version of Spectronaut and it's capabilities of processing large MS experiments on well affordable computer hardware.

Experiment Design

A large, non-sample-specific spectral library covering ~100,000 peptide precursor and 5607 proteins for targed data analysis using Spectronaut was used. The total size of the raw data was 1.5 TB. Two different experiments were selected in order to benchmark the scalability of Spectronaut. The first experiment consists of 259 HeLa runs recoreded on 3 different Thermo instruments (Q Exactive™, Q Exactive™ HF, Orbitrap Fusion™ Lumos™). The set containd several different acquisition methods and chromatoraphy gradients ranging from 1 to 8 hours. The second experiment consisted of 240 human serum runs recorded on a Sciex TripleTOF 5600 with a total size of 500 GB [2]. A library containing ~60,000 peptide precursors and 1702 proteins was used. Both sets where converted to HTRMS to minimize the vendor API overhead during the benchmark. Ideally, the HTRMS conversion would be performed while the data-set is being acquired (Figure 1).

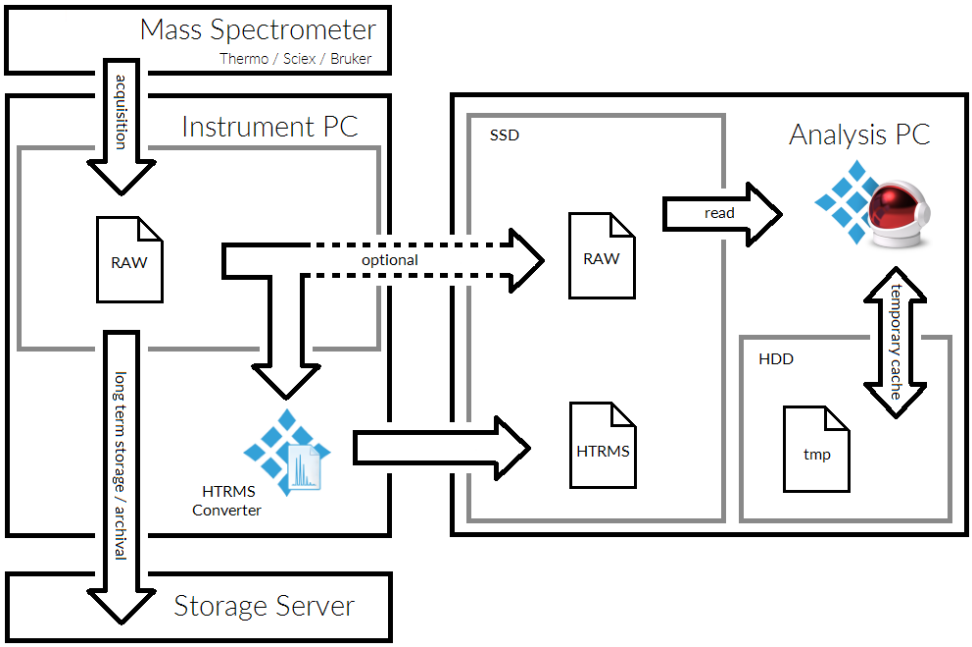

Figure 1. *Recommended system setup for running Spectronaut. Raw files can be either processed directly or converted to HTRMS. Data analysis from HTRMS will be on average > 50% faster as compared to the raw data. Conversion to HTRMS can be done on the instrument PC during experiment acquisition using the HTRMS converter. Raw or HTRMS data is then typically copied to an analysis PC. Ideally the analysis PC features two storage volumes, one to store the raw data on, the second volume for temporary caching of data by Spectronaut. The caching directory can be set in the global settings of Spectronaut (Temporary Directory). Having two volumes will speed up the processing on average by 5 to 10% because the disc IO is distributed. The performance difference is especially noticeable when loading directly from raw files. SSDs are preferred over spinning discs as they have a higher performance.*

Data analysis

Data analysis was run from converted HTRMS files on a Lenovo P700 workstation. The workstation has been configured with 2 Intel Xeon E5-2630 v3 processors with 8x 2.4 GHz and 128 GB of RAM. The data was stored on an internal 2 GB HDD. The temporary data storage of Spectronaut was set on an internal 2x 6TB HDD RAID-0 setup. Spectronaut was set to High-Performance mode in the general settings which allows the software to allocate more memory for a higher throughput. All other parameter have been left at default settings. A pre-release version of Spectronaut 10 was used for data analyis. The analysis was run including data normalization, interference correction as well as testing for differential abundance. These are computationally challenging processes as they require information for all runs within an experiment and cannot be performed run-wise.

Results

Data analysis for the first experiment comprising 259 Thermo LC-MS runs took 18 hours and 36 minutes. When analysing the data from converted HTRMS files the biggest contributors to the overall time consumption were the two calibration processes that are performed for each run (mass and retention time calibration). In case of a re-analysis with the same library, these steps would be skipped since the HTRMS file will save the calibration information. This would result in an overall decrease in analysis time of 6 hours. Only 42 MB of RAM per run accumulated over the course of the first experiment. The peak memory consuption for the first experiment was 42 GB of 128 GB available on the system.
Since the second experiment comprising 240 Sciex LC-MS runs was much smaller in terms of library and raw data size, the data analyis could be completed in only 9 hours and 45 minutes. For both experiments, the total average processing time per run was below 5 minutes. Memory accumulation per run was measured at 36 MB while the peak memory consumption spiked at 30 GB. HTMRS conversion took between 3 to 5 minutes per run. Data analysis done directly from raw or wiff would therefore roughly double the total analysis time.

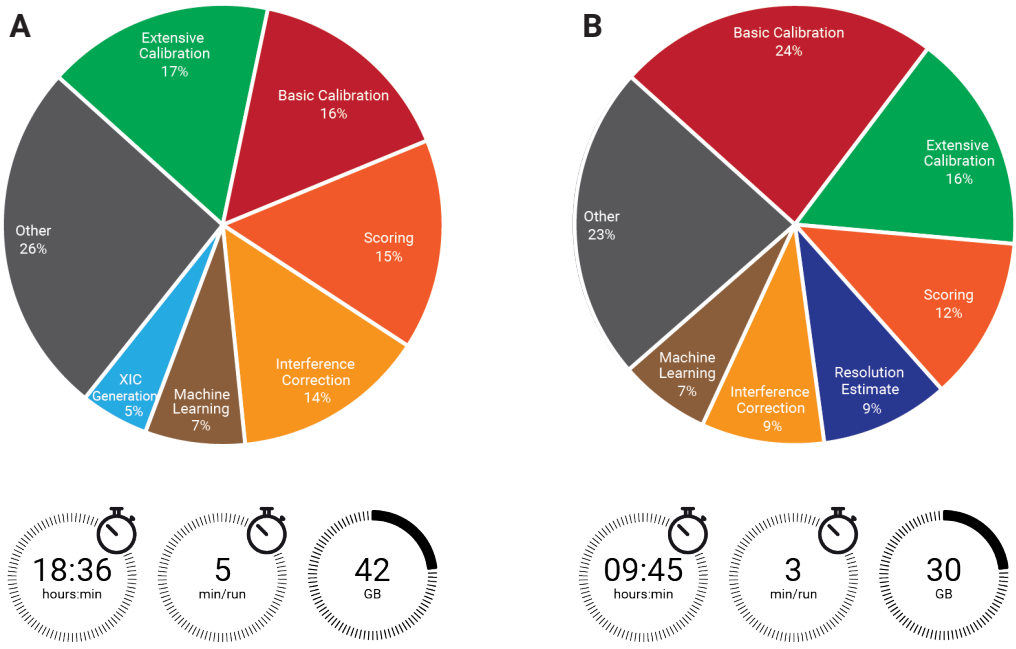

Figure 2. *Performance summary for the two experiments. A - Pie charts showing the relative time consumption per process for the two different experiments. Processes that contributed less then 5% to the total analysis time were summarized as "Other". This includes post analysis processes like normalization, clustering and regulation analysis. B - Performance counters for the two experiments. Peak working set memory as reported by the windows task manager has been used as measure for total memory consumption. This includes memory allocated by .NET that was not used during the process.*

Scalability

The memory usage in Spectronaut can be separated into two main categories. The dynamic part, which is data that will remain in RAM only for a brief amount of time. This includes raw data loading, the building of XICs and peak detection and scoring. And the static part which includes all object that stay in RAM till the end of the experiment. The first group will be neglectable for medium to large experiments and can be lowered by disabling the “High Performance Mode” in the settings of Spectronaut. The second group mostly correlates wiht the total number of targeted precursors per run. It is important to keep in mind that, if available, Spectronaut will allocate more memory from the system then directly necessary in order to minimize memory management overhead which will increase processing speed. The barplot therefore does not directly tell you how much memory Spectronaut will use but rather how much memory your system should have at least in order to process a certain experiment. Figure 4 shows the theoretical limit for two different spectral library sizes and 6 different system setups ranging from 8 GB to 256 GB of RAM.

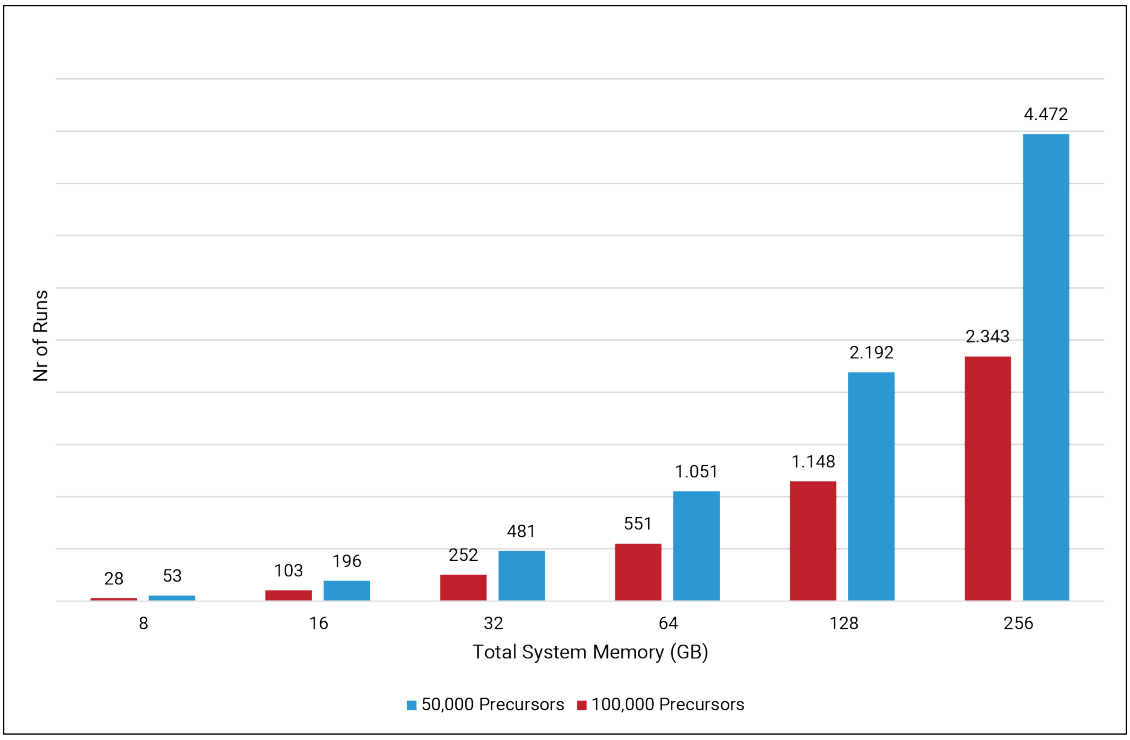

Figure 3. *Theoretical experiment size limits for different library sizes and system setups. The prediction was tested by running a 24 run experiment with a spectral library containing roughly 100,000 precursors on a Lenovo X220 laptop contain 8GB of RAM. The real peak memory consumption of this experiment was 5.4 GB.*

References

1. Bruderer R, Bernhardt OM, Gandhi T, Miladinovic SM, Cheng LY, Messner S, et al. Extending the limits of quantitative proteome profiling with data-independent acquisition and application to acetaminophen treated 3D liver microtissues. Mol Cell Proteomics. 2015.

2. Liu, Y., Buil, A., Collins, B. C., Gillet, L. C., Blum, L. C., Cheng, L. Y., ... & Dermitzakis, E. T. (2015). Quantitative variability of 342 plasma proteins in a human twin population. Molecular systems biology 2015, 11(2), 786.

3. Reiter L, Rinner O, Picotti P, Hüttenhain R, Beck M, Brusniak MY, Hengartner MO, Aebersold R, mProphet: automated data processing and statistical validation for large-scale SRM experiments, Nat. Methods 2011, 8: 430-435