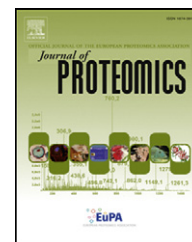


Available online at www.sciencedirect.com

SciVerse ScienceDirect

www.elsevier.com/locate/jprot

Technical note

ProteoCloud: A full-featured open source proteomics cloud computing pipeline☆

Thilo Muth^{a,b,c}, Julian Peters^d, Jonathan Blackburn^d, Erdmann Rapp^c, Lennart Martens^{a,b,*}

^aDepartment of Medical Protein Research, VIB, Ghent, Belgium

^bDepartment of Biochemistry, Ghent University, Ghent, Belgium

^cMax Planck Institute for Dynamics of Complex Technical Systems, Magdeburg, Germany

^dInstitute of Infectious Disease and Molecular Medicine and Division of Medical Biochemistry, University of Cape Town, Cape Town, South Africa

ARTICLE INFO

Article history:

Received 19 October 2012

Accepted 21 December 2012

Keywords:

Proteomics

Bioinformatics

Mass spectrometry

Cloud computing

Search algorithms

ABSTRACT

We here present the ProteoCloud pipeline, a freely available, full-featured cloud-based platform to perform computationally intensive, exhaustive searches in a cloud environment using five different peptide identification algorithms. ProteoCloud is entirely open source, and is built around an easy to use and cross-platform software client with a rich graphical user interface. This client allows full control of the number of cloud instances to initiate and of the spectra to assign for identification. It also enables the user to track progress, and to visualize and interpret the results in detail. Source code, binaries and documentation are all available at <http://proteocloud.googlecode.com>. This article is part of a Special Issue entitled: EUPA 2012: NEW HORIZONS.

© 2013 Elsevier B.V. All rights reserved.

Mass spectrometry (MS) based proteomics provides an efficient high-throughput method for the analysis of peptides and proteins [1]. In a typical proteomics tandem mass spectrometry (MS/MS) experiment, as stored in the ms_lims system in Ghent [2], on average 80% of the recorded fragmentation spectra remain unidentified, however. This number has not shifted dramatically throughout the past 10 years [3]. A proportion of these derives from contaminant and poor quality spectra, yet the remaining fraction of spectra is likely to contain unexpected peptides. To successfully assign high-quality unidentified spectra to peptides more permissive identification strategies are required. Such methods, including tag-based and *de novo* searches, typically come with a high computational cost however, and the processing of a large quantity of experimental spectra necessi-

tates servers with a large storage capacity. While such exhaustive identification attempts can thus be beneficial to most proteomics labs, they are rarely carried out due to a lack of computational power, and limited access to bioinformatics support to run the various identification algorithms. We therefore here propose ProteoCloud, a free and open software pipeline that is aimed at allowing end users to easily identify very large sets of spectra using five distinct algorithms through cloud computing technology [4,5]. This allows the use of a virtually unlimited number of servers with high performance to execute the computationally intensive tasks in parallel, dramatically reducing the wall time it takes to process the data [6,7]. The key innovations in our ProteoCloud software pipeline compared to previous efforts by Halligan et al. [6] and Mohammed et al. [7] lie in the use of

Abbreviations: MS, mass spectrometry; MS/MS, tandem mass spectrometry.

☆ This article is part of a Special Issue entitled: EUPA 2012: NEW HORIZONS.

* Corresponding author at: Department of Medical Protein Research and Biochemistry, VIB and Faculty of Medicine and Health Sciences, Ghent University, A. Baertsoenkaai 3, B-9000 Ghent, Belgium. Tel.: +32 92649458; fax: +32 92649484.

E-mail address: lennart.martens@ugent.be (L. Martens).

1874-3919/\$ – see front matter © 2013 Elsevier B.V. All rights reserved.

<http://dx.doi.org/10.1016/j.jprot.2012.12.026>

Please cite this article as: Muth T, et al, ProteoCloud: A full-featured open source proteomics cloud computing pipeline, J Prot (2013), <http://dx.doi.org/10.1016/j.jprot.2012.12.026>

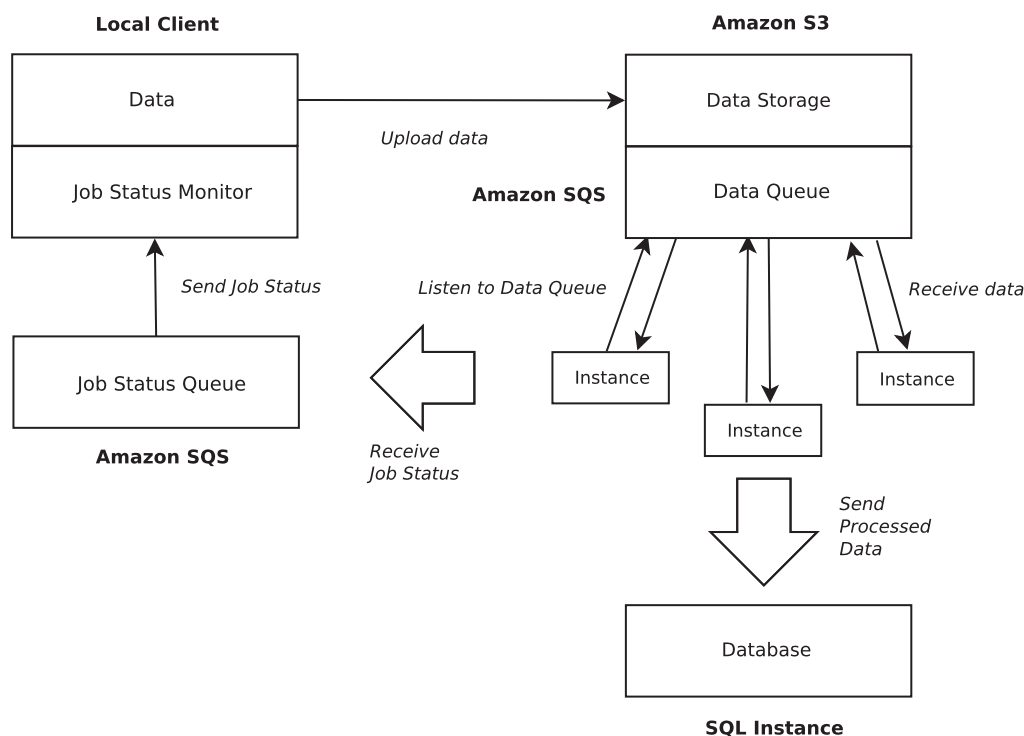


Fig. 1 – Cloud computing workflow diagram. The local client runs the controller unit, while all other tasks are located in the Amazon compute cloud. Data queues and the shared Amazon S3 storage allow the controller, workers, and database unit to communicate data, progress and tasks. See main text for details.

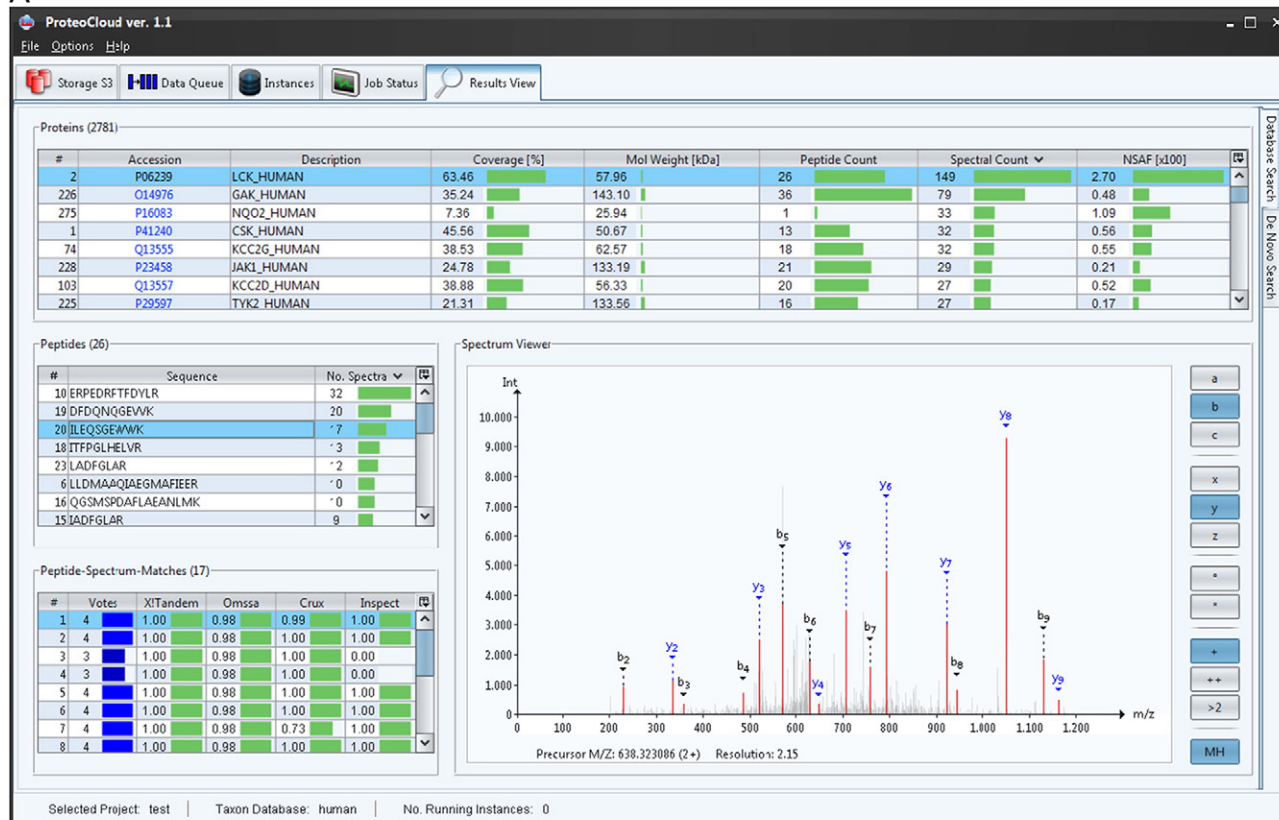
a combination of different identification methods including database searching, tag-based and *de novo* sequencing algorithms, and in the production-grade client and server software that dramatically lower the adoption threshold of cloud computing.

The ProteoCloud software is written in the Java programming language and can be downloaded from <http://proteocloud.googlecode.com>. The application makes use of JetS3t (<http://www.jets3t.org>) as the library to access the Amazon S3 storage system and typica (<http://code.google.com/p/typica>) to access the Amazon EC2 compute cloud and SQS queuing service. The virtual server instances use a 64-bit Ubuntu (10.04 LTS Lucid) public Amazon Machine Image (AMI) as basic template. Additionally, the database search software X!Tandem [8], OMSSA [9], Crux [10], Inspect [11] and the *de novo* peptide sequencing algorithm PepNovo+ [12] were installed on the image. The QVality [13] software was added to the image in order to obtain false discovery rates for the database search results. In order to store peptide and protein identifications to the MySQL relational database of the system, OMSSA Parser [14] and X!Tandem Parser [15] were used for extracting the results from the search engine output files.

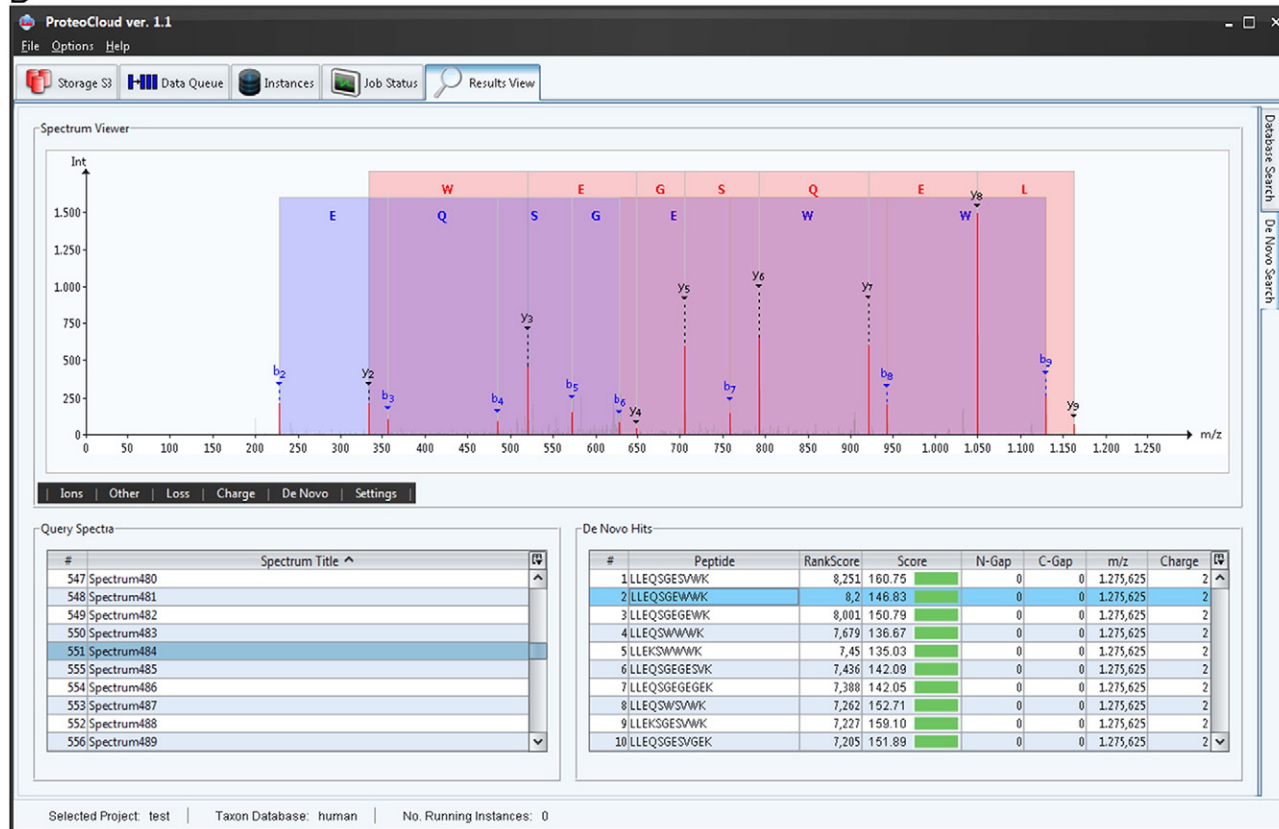
The ProteoCloud pipeline consists of three types of units: a single client-side controller unit, that constitutes the master control for the pipeline; any number of worker units that will perform the actual identifications; and a single database unit running the MySQL database instance where all results will be stored. These three types of units are tied together through the Amazon S3 based shared data storage system on the one hand, and three message queues on the other hand: the data queue allows the controller unit to inform worker units about input data to process; the status queue continuously provides the controller unit with updates on the workers' status; and the database queue is used by the workers to deposit their results so the database unit can store them. A complete workflow diagram of ProteoCloud is shown in Fig. 1, starting with the loading of data into the controller unit. These data are then partitioned into files of configurable size before they are sent to the data storage. At the same time, the controller unit also puts out messages on the data queue that will be picked up by idling worker units, informing them of the location of a collection of spectra, and the relevant parameters required to process them. Upon receiving such a data message, the worker unit will flag it as 'in progress' so other

Fig. 2 – Screenshots of the graphical user interface of the ProteoCloud client application. (A) The client application shows details on proteins (top), including sequence coverage, spectral counts, and NSAF abundance calculations, peptides (middle left) and peptide-to-spectrum matches (lower left) including the consensus and individual algorithm q-values. The selected spectrum is shown on the right, including user-configurable annotations corresponding to the highlighted peptide match. (B) For *de novo* searches, the different hits and their scores are shown on the lower right for each selected spectrum (shown on lower left). When selected, the *de novo* sequence is highlighted on the spectrum panel (shown on top).

A



B



workers will ignore it, and will download the corresponding file of spectra from the data storage to the local hard drive. The worker unit then automatically starts processing the spectra, informing the controller through messages on the status queue of progress in every individual processing step. The controller unit is thus able to identify precisely which process runs on each of the various worker instances at any time, and reports this in real time to the user. When the worker unit completes the processing of its assigned spectra, it transfers the results to the database server, and moves the original data message from the data queue to the database queue. This allows the forwarded message to transfer the information about the processed data to the database unit. When the database unit receives such a message, it parses the output data produced by the worker and stores the content and context in the relational database.

Because ProteoCloud relies on five distinct algorithms to identify each spectrum, the different results need to be integrated. We therefore implemented a voting system where each algorithm puts in a vote when it confidently identifies a spectrum according to the *q*-value provided by QValue. If no peptide was identified, there is no vote from this algorithm for the spectrum. Since each algorithm produces somewhat different identifications overall [16], and different algorithms may produce conflicting identifications for a spectrum, the voting mechanism simply aggregates the vote of each algorithm. A confident identification by any one algorithm is then considered correct, unless the other algorithms disagree on the identification, providing at least one more confident suggestion. Since the underlying metric is the *q*-value produced by QValue rather than the original search engine score, this method thus allows the straightforward detection of strong consensus (agreement among the majority of algorithms), specific identification expertise (a highly confident identification proposed by a single algorithm), or strongly conflicting opinions (several votes for one confident identification, against several other votes for a different, but roughly equally confident identification). The client software allows the detailed inspection of the final output in a user-oriented graphical user interface, by showing the proteins, peptides, and peptide-to-spectrum matches for each spectrum, including the consensus achieved and the detailed spectrum annotation (see Fig. 2). Protein sequence coverage, peptide count, spectral count and NSAF abundance estimation are also provided in the same interface. A separate, specialized interface is reserved for *de novo* identifications, where many rivaling options may be provided for a single spectrum.

ProteoCloud thus makes it easy to wield the considerable power of five state-of-the-art algorithms in the field, with a few clicks of a button, while delegating all the computational workload to on-demand cloud computing instances. As an example, five different datasets (each containing 1000 Orbitrap MS/MS spectra) were searched against the human UniProt FASTA database. On average, the upload time of 1000 MS/MS spectra was less than 2 min and the running time for the five algorithms was about 48 min, of which the *de novo* algorithm took about three quarters of the total running time. This allows end users to quickly and easily perform exhaustive identification on hundreds of thousands to millions of spectra at very little overhead cost. Furthermore, our cloud platform performs

search engine integration with full provenance of information, and archives all search results in a cloud-based relational database that can be downloaded locally for further offline use. Our ProteoCloud pipeline is also unique across the available cloud-based proteomics platforms in that it offers *de novo* sequencing of spectra, the most computationally intensive task in the pipeline, and the one that most benefits from the on-demand computational power of the cloud. The rich client allows exquisite and detailed control over the cloud instances and task allocation, provides the user with constant real-time progress updates, and comes complete with a detailed and intuitive interface for (offline) result inspection.

The authors would like to thank Sven Degroove, Niklaas Colaert, Kenny Helsens and An Staes for their support. J.P. thanks the Carnegie Corporation for a studentship. J.B. thanks the National Research Foundation, South Africa, for a research chair. L.M. acknowledges the support of Ghent University (Multidisciplinary Research Partnership “Bioinformatics: from nucleotides to networks”), and the PRIME-XS project, grant agreement number 262067 funded by the European Union 7th Framework Program.

Conflict of interest

The authors declare no conflict of interest.

REFERENCES

- [1] Gevaert K, Van Damme P, Ghesquière B, Impens F, Martens L, Helsens K, et al. A la carte proteomics with an emphasis on gel-free techniques. *Proteomics* 2007;7:2698–718.
- [2] Helsens K, Colaert N, Barsnes H, Muth T, Flikka K, Staes A, et al. *ms_lims*, a simple yet powerful open source laboratory information management system for MS-driven proteomics. *Proteomics* 2010;10:1261–4.
- [3] Flikka K, Martens L, Vandekerckhove J, Gevaert K, Eidhammer I. Improving the reliability and throughput of mass spectrometry-based proteomics by spectrum quality filtering. *Proteomics* 2006;6:2086–94.
- [4] Wall DP, Kudtarkar P, Fusaro VA, Pivovarov R, Patil P, Tonellato PJ. Cloud computing for comparative genomics. *BMC Bioinformatics* 2010;11:259.
- [5] Fusaro VA, Patil P, Gafni E, Wall DP, Tonellato PJ. Biomedical cloud computing with Amazon Web Services. *PLoS Comput Biol* 2011;7:e1002147.
- [6] Halligan BD, Geiger JF, Vallejos AK, Greene AS, Twigger SN. Low cost, scalable proteomics data analysis using Amazon's cloud computing services and open source search algorithms. *J Proteome Res* 2009;8:3148–53.
- [7] Mohammed Y, Mostovenko E, Henneman AA, Marissen RJ, Deelder AM, Palmblad M. Cloud parallel processing of tandem mass spectrometry based proteomics data. *J Proteome Res* 2012;11:5101–8.
- [8] Craig R, Beavis RC. TANDEM: matching proteins with tandem mass spectra. *Bioinformatics* 2004;20:1466–7.
- [9] Geer LY, Markey SP, Kowalak JA, Wagner L, Xu M, Maynard DM, et al. Open mass spectrometry search algorithm. *J Proteome Res* 2004;3:958–64.
- [10] Park CY, Klammer AA, Käll L, MacCoss MJ, Noble WS. Rapid and accurate peptide identification from tandem mass spectra. *J Proteome Res* 2008;7:3022–7.

- [11] Tanner S, Shu H, Frank A, Wang L, Zandi E, Mumby M, et al. InsPecT: identification of posttranslationally modified peptides from tandem mass spectra. *Anal Chem* 2005;77: 4626–39.
- [12] Frank A, Pevzner P. PepNovo: *de novo* peptide sequencing via probabilistic network modeling. *Anal Chem* 2005;77:964–73.
- [13] Käll L, Storey JD, Noble WS. QVALITY: non-parametric estimation of q-values and posterior error probabilities. *Bioinformatics* 2009;25:964–6.
- [14] Barsnes H, Huber S, Sickmann A, Eidhammer I, Martens L. OMSSA parser: an open-source library to parse and extract data from OMSSA MS/MS search results. *Proteomics* 2009;9: 3772–4.
- [15] Muth T, Vaudel M, Barsnes H, Martens L, Sickmann A. XTandem parser: an open-source library to parse and analyse X!Tandem MS/MS search results. *Proteomics* 2010;10:1522–4.
- [16] Stephan C, Reidegeld KA, Hamacher M, van Hall A, Marcus K, Taylor C, et al. Automated reprocessing pipeline for searching heterogeneous mass spectrometric data of the HUPO Brain Proteome Project pilot phase. *Proteomics* 2006;6:5015–29.