

A Platform for Comparative Mass Spectrometry Based Proteomics

Extending CPAS to support quantitation and ProteinProphet

Mark Igra¹, Brendan MacLean², Joshua Eckels², Adam Rauch², Peter Hussey², Vitor Faca¹, Samir Hanash¹, Martin McIntosh¹

¹Fred Hutchinson Cancer Research Center, Seattle, WA USA

²LabKey Software, LLC, Seattle, WA USA

Overview

The CPAS [1] open source proteomics analysis system has been extended to provide a more robust and extensible platform for mass spectrometry based proteomics. These extensions provide the ability to manage, load and analyze data from new experiment types, differing hardware and software platforms, and additional analysis procedures including ProteinProphet [2] analysis and MS1-based quantitation strategies. This is achieved via an architecture that supports pluggable providers at a variety of levels including extensibility in the pipeline for information processing, experiment and data file loading, and reporting of the results of these new data sets.

We have used to use this architecture to process, load, display and analyze relative quantitation data produced by measuring labeled samples on an LTQ-FT from Thermo-Finnigan. The MS data was processed by the XPRESS [3] and Q3 [4] algorithms for relative quantitation, as well as ProteinProphet to provide robust protein identification. All of these results were loaded into the CPAS system and analyzed.

Introduction

Mass Spectrometry-based proteomics experiments are performed using a variety of environments. The CPAS platform is a system for storing and analyzing datasets acquired via LCMS. The system has been widely used (over 12,000 MS2 runs and 120,000,000 spectra in our main installation) and has been downloaded by over 100 institutions.

Though CPAS was useful for a subset of proteomics experiments it did not support quantitation or robust protein identification with ProteinProphet. CPAS users had to resort to additional tools to perform data analysis. Our objective was to fully support these experiment types, and to provide an architecture that could be easily extended to support new experiment types as the field evolves.

Methods

We designed and implemented a modular Java architecture for processing, loading and analyzing experimental data. The architecture comprises a general purpose data processing pipeline, an extensible experiment loading module and online analysis tools.

Data Processing Pipeline

A general purpose pipeline module allows extensions of the Java PipelineProvider class to provide data processing procedures without changing the core architecture. Each PipelineProvider is responsible for processing an arbitrary set of file types, supplying actions to perform on those files. For mass spectrometry based proteomics these files include raw and mzXML format mass spec files, peptide sequencing or quantitation files. Each pipeline can supply a custom user interface for starting jobs and performing work asynchronously, communicating status to the web server. Different pipelines can provide actions for the same files, for example, both quantitation and peptide sequencing pipelines might offer services for mzXML files

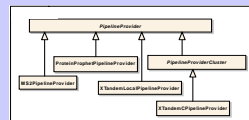


Figure 1
This UML diagram shows a subset of PipelineProvider classes for processing experimental data types. The ProteinProphetPipelineProvider class is responsible for running ProteinProphet analyses. Also shown are specialized PipelineProvider subclasses for running data on a cluster.

Results

We used the expanded experiment pipeline to process, load and analyze data from a series of mass spec runs from a Thermo-Finnigan LTQ-FT. Each run consisted of two samples of human sera, one of which was treated with D0 acrylamide and the other of which had been treated with either D3 or ¹³C isotopes of acrylamide to provide quantitation.

We were able to process this data with XPRESS, TANDEM [7], PeptideProphet and ProteinProphet using the CPAS pipeline. The extensible data loaders loaded quantitation and ProteinProphet data into the CPAS system.

We were able to use new display features in CPAS to show and modify quantitation data, as well as examine protein groups from ProteinProphet.



Figure 2
This experiment process graph shows the sample and data processing used in our experiment. The data was searched using TANDEM, quantitated with XPRESS. In addition ProteinProphet scores were computed to identify proteins.

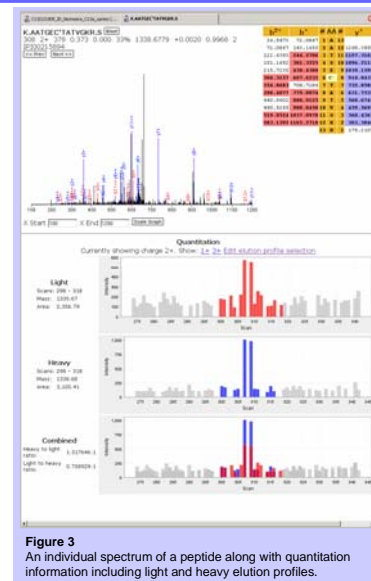


Figure 3
An individual spectrum of a peptide along with quantitation information including light and heavy elution profiles.

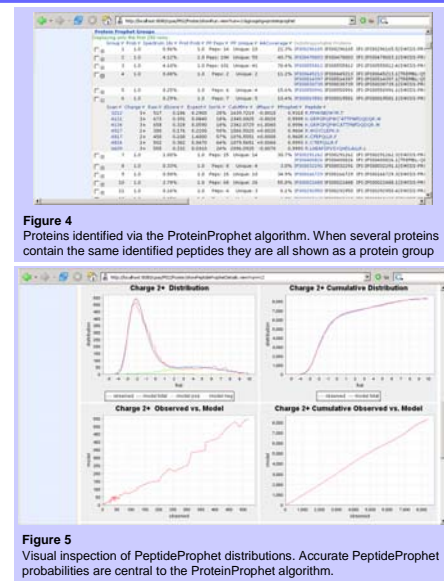


Figure 4
Proteins identified via the ProteinProphet algorithm. When several proteins contain the same identified peptides they are all shown as a protein group

Figure 5
Visual inspection of PeptideProphet distributions. Accurate PeptideProphet probabilities are central to the ProteinProphet algorithm.

Data Loading Architecture

The CPAS experiment management service is responsible for loading FuGe [5] based experiment descriptions into the CPAS database. The experiment loading service supports a modular architecture for loading new data types and for providing custom code for describing experiments. The CPAS XarReader class calls an ExperimentDataHandler class to load each file type within a XAR. We implemented a new data handler to load ProteinProphet files.

Peptide sequencing and quantitation results are loaded from a pepXML [6] file. pepXML files can contain a variety of results. Once again a modular system was implemented to support loading peptide searching and quantitation data. Several different quantitation summary loaders were built including support for both XPRESS and Q3 quantitation.

Online Analysis Tools

We built an elution profile viewing and editing tool to allow experimenters to visually evaluate the spectra determined to be differentially labeled versions of the same peptide. Experimenters can adjust the machine-generated quantitation ratio by manually changing the scan range attributed to the light and heavy labeled versions of the peptide.

We also built a custom view of ProteinProphet output. This view supports protein groups, which are a set of proteins that are indistinguishable based on the set of identified peptides.

Conclusions

We present an extensible experimental pipeline for processing and loading experimental data. We built modules to process a variety of experimental processes including

- A clustered pipeline supporting TANDEM for MS based proteomics
- Support for loading data from multiple peptide sequencing programs
- Support for quantitation
- Support for reliable protein identification via ProteinProphet

Future directions might include

- Incorporating ProteinProphet protein groups into comparison of multiple runs
- Implementing a full workflow processing system for improved asynchrony
- Improved association of sample properties with quantitated samples

References

- 1) Rauch A et al. *J Proteome Res*, 2006, 5(1), 112-121.
- 2) Nesvizhskii A et al. *Anal Chem*, 2003 Sep 17;75(17), 4646-951.
- 3) Han, D.K. et al. *Nat Biotechnol*, 2001 Oct;19(10), 946-951.
- 4) Faca, V. et al. *J. Proteome Research*, 2006, in press.
- 5) Jones, A. et al. *Bioinformatics* 2004, 20 (10), 1585-1590.
- 6) Keller et AL. *Mol Syst Biol*, 2005, *EPub*, (August).
- 7) Craig, R.; Beavis, R. C., *Bioinformatics* 2004, 20 (9), 1466-1467.