

# A Pattern for Protein Identification

Jens Lichtenberg, Lonnie Welch  
Center for Intelligent Distributed and Dependable Systems  
Ohio University  
Athens, Ohio, 45701, USA  
{lichtenj,welch}@ohio.edu

**Abstract.** This paper presents the protein identification problem in form of a design pattern. Not unlike biological patterns, who describe reoccurring structures within biological objects, a design pattern in regard of software engineering is seen as the description of a thing, which is alive and the process, which creates it. The pattern presented in this article addresses the identification of the correct protein based on a given peak list. It provides a process on how to infer the identity and primary structure of such protein based on searches of existing protein mass databases and a process describing how to create these databases. The design pattern for protein identification enables software developers to create their own protein mass fingerprinting solutions, which can be adjusted, modified and extended to an experiment's specific needs. To visualize and enhance the understanding of the pattern a process flowchart of protein identification is presented as well as a local implementation of the pattern.

**Key words:** pattern, protein identification, mzdata, software engineering, HUPO, PSI

## Introduction

Proteomics is the study of proteomes. While each cell has one genome, this genome gets translated into different proteomes, which determine the function of a cell and its biological destiny. Proteomics literature identifies six research areas within the field: Mining, Protein expression profiling, post-translational modification mapping, protein network mapping and protein structure prediction. These areas can be combined to present complete application and their patterns will form the basis of a proteomics pattern language.

Liebler (Liebler 2002) defines Protein expression profiling as a specialized form of mapping that discovers the identity of a protein in a particular sample or proteome. There are multiple sub-steps to the expression profiling approach, that can be seen as major steps in itself but are grouped for structure in the expression profiling. Expression profiling or expression proteomics can be based on either a comparison of healthy and a diseased two-dimensional gels or a protein chip experiment. In order to distinguish proteins from one another a protein separation phase is needed, which maps out the proteins physically based on their molecular weight and isoelectric point.

After the creation of these maps or gels for different time points during the progression of a disease with additional control gels of non-diseased sample it is necessary to digitalize the gel images and analyze them in regard to expression level changes. During the image analysis of the digitalized gel a statistical analysis is conducted in which proteins of interest are determined. A protein of interest is marked as a protein that is significantly changed either through location shifts, or expression differences.

After the identification of proteins of interest, the identity of these proteins has to be determined. In order to do this more information about the proteins has to be generated. While it is possible in general to infer the molecular weight and isoelectric point of a protein through measurements on the 2D gel, these measurements are neither very accurate nor precise. In order to map a protein based on its molecular weight (a specificity that is also applied in the protein separation of the two-dimensional gel analysis) a mass spectrometry of the protein can be conducted

In order to conduct a mass spectrometry experiment the digested peptide mixture has to be mixed with a chemical matrix to create a source. Differentiating between two types of mass spectrometry, the matrix is brought onto a sample plate forming a dried source for a matrix-assisted laser desorption ionization (MALDI) type spectrometry, while it remains aqueous for electrospray ionization (ESI) spectrometry experiments.

During a MALDI experiment a laser is fired onto the source which leads to the absorption of photons by the matrix, that is passed on to the peptide fragments. These fragments leave the matrix structure as both positive and negative ions to be processed by a mass analyzer. In an ESI experiment the aqueous sample is sent through a high voltage needle or stainless steel cone, resulting in a mist of peptide ions and other components. The peptide ions are separated from the other components by a heated capillary or a curtain of nitrogen gas resulting in desolvated ions which are passed on to a mass analyzer.

Various mass analyzers are available and can be combined (eg. Tandem MS or MALDI-TOF-TOF). The time of flight (TOF) analyzer, which is commonly applied to MALDI experiments measures the time needed by the ions to travel from one end of the analyzer to the other. The ion speed, measured by a detector at the end of the TOF analyzer, can be used to determine a mass-to-charge ratio, with the mass being peptide specific and the being instrument specific (in most cases the charge is set to 1). TOF analyzer can be used with ESI sources as well. Another mass analyzer and detector combination is the quadrupole, which captures the ions between four metal rods, set in parallel. The voltage can be varied in order to only let peptides of a specific mass pass on to the detector, where the resulting mass-to-charge ration is measured. An ion trap analyzer present a mass analyzer that get filled with ions and filters them out until only certain ions remain, which are then recorded by a detector. The ion trap contains two electrodes in parallel, used to fill and empty the ion chamber, and a ring shaped electrode surrounding the ion chamber, used to keep the ions in place between loading and filtering processes.

The mass-to-charge values resulting from the mass spectrometry are called peak lists or mass spectra. They contain the mass-to-charge values detected by the detector of the mass analyzer. The peaks correspond to peptide mass under a certain charge. In order to determine the mass of a peptide, it just has to be divided by the charge. In certain cases a peak intensity is also noted. Based on the peptide masses of a protein it is now possible to search for such a fingerprint among previously known protein sequences – a process called protein identification.

This paper presents an architectural pattern for the process of protein identification in the POSA format. POSA was chosen based on its ability to cover multiple levels of abstraction, from complete architectural patterns down to low level idioms. The necessity of multiple levels of abstraction arises from the upcoming extension of the presented research to include the various processes needed in the protein expression profiling and eventually the description of patterns for the remaining areas of proteomics. A complete proteomics pattern language is the goal of this research.

## Protein Identification

---

---

Protein identification begets the mass spectrometry of a protein sample. While mass spectrometry is a pattern in itself, it can be described here as being the process, which fragments a protein sample and measures the mass to charge ratio of the resulting digested fragments. The mass to charge ratios are stored in a so called peak list. Although the Human Proteome Organization (HUPO) is working on a Proteomics Standard Initiative (PSI) (Kaiser 2002), no generally accepted format for peak lists has been created so far, which leads to a peak list ranging from having a single mass to charge ratio column to having a mass-to-charge, an intensity<sup>1</sup> column and a charge column.

The Protein Identification architectural pattern allows for the identification of proteins represented by Mass Spectrometry peak lists. It infers the identity and primary structure of a protein based on an existing mass database.

---

---

Also known as

Protein Mass Fingerprinting, Mass Database Search

Example

To illustrate and validate the described approach a peak list is created in-silico. In-silico creation of peak lists is a design pattern in itself but can be summarized as the digestion of an amino acid sequence by software under

---

<sup>1</sup> Abundance of a protein fragment within a mass spectrum as detected by the mass analyzer during the mass spectrometry.

assumption of the same extrinsic variables as in a regular mass spectrometry experiment. For a complete in-silico digestion a protein sequence is scanned for the digestion sites of the used digestion enzyme<sup>2</sup>. The amino acids between digestion sites are stored as a sequence fragment. After digestion the molecular weight has to be determined. This is done through the summation of the masses for each of the amino acids belonging to a fragment and the base mass (18.0105 da<sup>3</sup>). The common amino acids in a protein sequence and their respective masses are presented in the chemistry compendium of the International Union of Pure and Applied Chemistry (IUPAC) (Irving, Freiser et al. 1978).

To serve as an example the amino acid sequence of Troponin (Schmidtman, Lindow et al. 2005) is digested in-silico. Based on the described process, the sequence, shown in the FASTA format in Figure 1, is processed into the peak list.

```
>gi|4507615|ref|NP_003271.1| troponin C, slow [Homo sapiens]
MDDIYKAAVEQLTEEQKNEFKAAFDIFVLGAEDGCISTKELGKV
MRMLGQNPTPEELQEMIDEVDEDGSGTVDFDEFLVMMVRCMK
DDSKGKSEEELSDLFRMFDKNADGYIDLDELKIMLQATGETITE
DDIEELMKDG DKNNDGRIDYDEFLEFMKGVE
```

Figure 1: FASTA formatted amino acid sequence of troponin.

The very first sequence fragment MDDIYK would based on the described process and the amino acid masses amount to a monoisotopic molecular weight of 783.3473 (including the base mass of 18.0105 Da):

$$131.04049 + 2 * (115.02695) + 113.08407 + 163.06333 + 128.09497 = 765.33676$$

$$765.33676 + 18.0105 = 783.3473$$

The complete analysis of all the tryptic fragments of troponin under the assumption of a mass range from 800 to 4000, which is common for mass spectrometers, leads to the peak list depicted in Table . This peak list will be used to illustrate the protein identification process and the eventual identification of troponin provides a proof of concept.

<sup>2</sup> In most cases this digestion enzyme is Trypsin with digestion sites at Lysine (K) and Arginine (R).

<sup>3</sup> Da – Dalton is the official mass measurement of amino acids

Table 1: In-silico created peak list.

In-Silico Peak List			
2009.962	1744.798	2283.146	2694.262
1244.625	1885.877	831.4636	989.4162
1762.874	1364.646	825.3361	2004.889
2374.151	3625.727	1408.683	1448.653
1855.903	2279.091	1223.567	1733.786

Context

Protein identification is heavily dependent on the output of a mass spectrometry experiment and the mass database of choice. The output of a typical mass spectrometry experiment is presented in Table of the example. A more complex example of a peak list is presented in Figure 2.

The peak list contains additional information about the intensity and charge of a peak. Intensity is defined by the International Union of Pure and Applied Chemistry (IUPAC) as the magnitude of a particular feature in a photochemistry spectrum (Irving, Freiser et al. 1978). While not

842.509948730469,	59514.33984375,	1
856.513671875,	11297.5888671875,	1
870.532470703125,	22048.701171875,	1
881.267211914063,	50740.3515625,	1
899.366394042969,	7076.2587890625,	1
...		

Figure 2: Shortened detailed peak list containing columns for mass to charge ratio, intensity and charge

reporting the actual abundance of an ion, it is still loosely correlated with it. Charge on the other hand is the ionic charge used to create the magnetic or electric field in the mass analyzer.

Protein identification based on mass spectrometry data requires a mass database. While it is possible to use an existing mass database like OWL (Bleasby, Akrigg et al. 1994), it is more common to create a mass database based on an existing sequence database. Numerous sequence databases are available among them NCBIInr (Benson, Karsch-Mizrachi et al. 2006),

MSDB (Perkins 2006) and Swissprot (Wu, Apweiler et al. 2006). In order to create a mass database from the information stored in these databases it is necessary to conduct an in-silico digestion of all proteins contained in them. It is possible to add information about post-translational modifications to these mass database by adding the complete set of in-silico modified fragments for each protein under each incorporated PTM. While this seems to be an easy way to address the possible modifications of peak, it increases the number of fragments in the mass database and influences the identification process through an increased number of false positives.

#### Problem

The resulting peak list from a mass spectrometry experiment has to be analyzed in order to identify the correct protein that was submitted as a sample to the process. To accomplish this existing protein databases have to be searched for the corresponding protein of a peak list. Based on the gathered information the sequence of the protein has to be inferred. To support protein identification following forces need to be resolved:

- Possible protein mixtures have to be accounted for with the complete identification of each included protein or protein fragment.
- Multiple proteins can have the same fragment pattern and masses, which has to be accounted for by the protein identification solution.
- For a correct identification, the biological significance of the predicted protein has to be determined in regard of the proteomic experiment.
- To maximize throughput, the protein identification solution has to be adaptable to the most generic format of peak lists and be able to produce good results with a set of common parameters.

#### Solution

A general overview of the protein identification solution is depicted in Figure 3. To allow for protein identification with peak list data a mass database is needed that can be searched for corresponding masses to the peaks.

To create a mass database one protein at a time has to be read into the database at a time. This protein has to be digested and the molecular weight for each fragment has to be calculated. This process has to be performed for every protein in the sequence database. A minimal mass database has to contain the protein accession number as a key for the original sequence database, the fragment sequence and the molecular weight of that fragment. The most detailed mass database would contain

all the protein information for each detected fragment in addition to the minimal information stored.

Once a mass database is created protein identification can be conducted. In order to detect the correct protein in the mass database a peak list describing the protein is needed. A peak list can be either provided directly as an array in the parameters of a protein identification solution or through a peak list file. If a peak list file is provided, this file needs to be read and the peak list extracted and once again submitted to the actual identification process as an array.

The protein identification process, highlighted through the box in Figure 3 reads one peak at a time from the peak list array and tries to detect it in the database. During the detection it is important that the identification process allows for a previously specified error. This error has to be taken into account since currently available mass analyzers, while being very accurate, still allow for slight errors. The stringency of this parameter has a great influence in the detection of false positives. It is regarded as a good approach to allow for an error of 5 parts per million (ppm). Based on various indexing techniques to the database it is not necessary to compare each peak to every entry in the mass database. After the comparison of the peak to a chosen set of database record, the matched entries are stored in another array together with the analyzed peak.

After the analysis of the complete peak list array all peaks and their matched database records are stored in an array. Based on this match information a scoring scheme can be applied. A scoring function will analyze the matched peaks for each protein that contains one or more matched peaks. Applied scoring schemes can rank from just a summation of the number of matches for a protein (Pappin, Hojrup et al. 1993) up to complex Bayesian probability models (Zhang and Chait 2000). The differences in scoring schemes determine the quality of one protein identification approach over another and their implementation is often times an unpublished, valuable secret in commercial protein identification solutions. Some of the most common scoring schemes will be presented in the “Known Uses” of this pattern.

The results of the scoring have to be presented in a final step to the biologist with enough information to be able to make a founded choice of protein. This information usually contains a score and a confidence value describing the probability that the protein is a false positive.

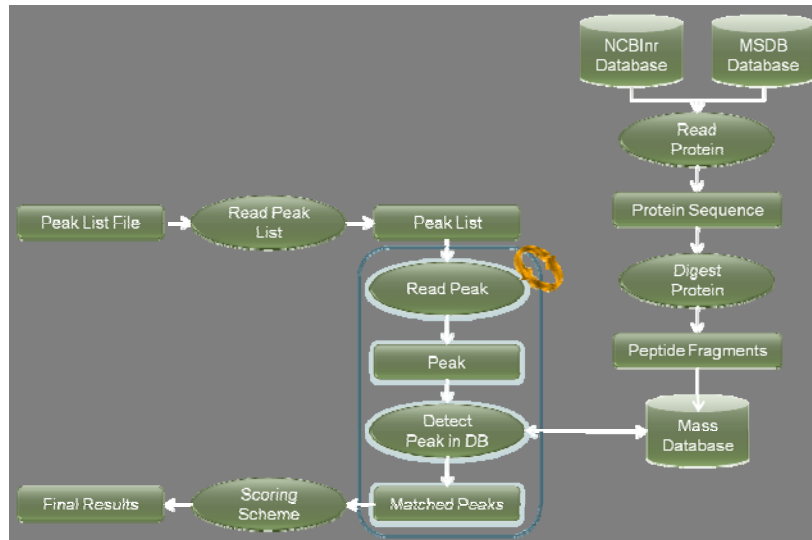


Figure 3: Protein identification pattern solution.

Structure

The protein identification pattern can be structured into three classes. A protein class is needed to describe a database record. A unique *accession* number, a protein *name* and an amino acid *sequence* would describe a Protein object. While these main attributes are required, additional information can be gathered through a *lookup()* method that returns the information stored in the specified database field. In addition to *lookup()* Protein needs methods to return the variable content for the sequence and the accession number (*get\_sequence()*, *get\_accession()*). Since not all sequence databases store molecular weight explicitly it is necessary to incorporate a method that calculates and returns the molecular weight for the protein based on its amino acid sequence (*get\_mol\_weight()*). To create a protein mass fingerprint - a complete digested protein - a digestion method is needed called *digest()*, which create Peptide objects by digesting the amino acid sequence stored in *sequence*. Upon completion of the actual matching process the scoring function has to be called, which assigns a score to each Protein object using the *scoring()* method.

<p><b>Class</b></p> <p>Protein</p>	<p><b>Collaborator</b></p> <ul style="list-style-type: none"> <li>• Peptide</li> </ul>
<p><b>Responsibility</b></p> <ul style="list-style-type: none"> <li>• A protein identifies a sequence database record</li> <li>• Used to digest an amino acid sequence and create Peptide objects</li> </ul>	

The Peptide objects, created during the *digest()* of *sequence*, contain the respective fragment sequence (*fragment\_sequence*) and the accession number (*accession*) of the digested protein as attributes. Once more methods are needed to retrieve the private attributes (*get\_fragment\_sequence()*, *get\_accession()*). In addition a molecular weight calculation method (*get\_mol\_weight()*) is needed, which determines the mass of the peptide.

<p><b>Class</b></p> <p>Peptide</p>	<p><b>Collaborator</b></p> <ul style="list-style-type: none"> <li>• Peptide</li> <li>• Peak List</li> </ul>
<p><b>Responsibility</b></p> <ul style="list-style-type: none"> <li>• A peptide identifies a sequence fragment</li> </ul>	

Based upon the input from the mass spectrometry experiment, a Peak List object is invoked. The location of a peak list file is set to *file* using the *set\_file()* method. A peak list in its simplest form consists of a mass-to-charge value identified as *peak*. To accommodate for more complex peak list formats, *intensity* and *charge* attributes are provided. The peaks are read into an array *peak[]* using the *read\_peak\_list()* method. Besides the usual private attribute retrieval methods (*get\_peak()*, *get\_intensity()*, *get\_charge()*) a *matching()* method is provided. The *matching()* method compares each of the entries in the *peak[]* array to a set of Peptide objects and stores the results in *matching\_results[]*.

<p><b>Class</b></p> <p>Peak List</p>	<p><b>Collaborator</b></p> <ul style="list-style-type: none"> <li>• Peptide</li> </ul>
<p><b>Responsibility</b></p> <ul style="list-style-type: none"> <li>• A peak list identifies the mass spectrometry input into the protein identification process</li> <li>• Used to conduct a matching between peaks and peptides</li> </ul>	

The structure of the participants in the Protein Identification pattern is illustrated in Figure 4.

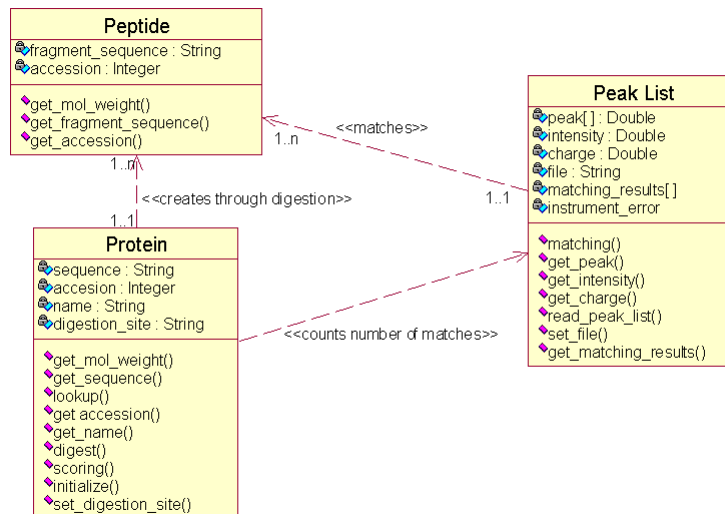


Figure 4: Class diagram of the Protein Identification pattern

## Dynamics

Collaborations within the Protein Identification pattern occur in two phases. The primary phase is the creation of the required mass database while the second phase describes the actual identification process.

**Phase I:** This phase describes the collaborations between protein and peptide participants in order to prepare the required mass database:

- An application that is empowered in the database setup initializes the protein by setting the amino acid sequence, accession number and name of the protein.
- The protein digests the provided sequence returns peptide objects for the resulting fragments.

**Phase II:** This phase describes the actual protein identification process:

- The protein identification application initialize a Peak List by setting its file.
- After reading data from the file the peak list conducts a matching against the existing peptides.
- In preparation for the scoring the peak list retrieves additional information from the proteins the matched peptides originated from.

- Based on existing information scores the peak list assigns scores to the proteins.
- The controlling protein identification application retrieves the results of the matching and subsequent scoring.

The collaboration diagram in Figure 5 visualizes the interactions of the two phases and links them together chronologically.

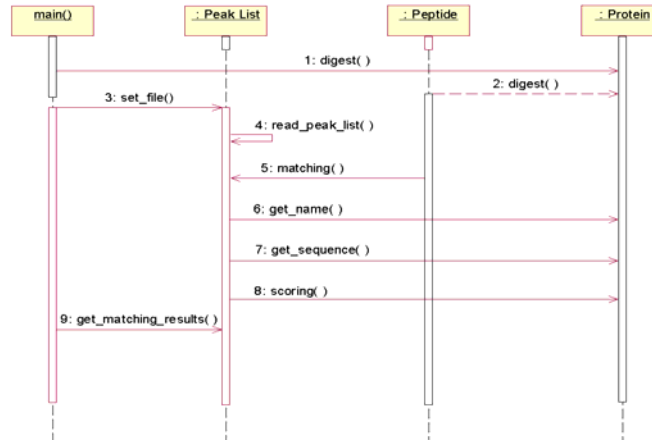


Figure 5: Protein identification collaboration diagram.

Implementation

The participants of the protein identification pattern can be described in two layers:

- *Database Setup:* This layer performs the tasks necessary to create a mass database, which is required for the protein identification process.
- *Protein Identification:* This layer performs the necessary tasks to determine the identity of a protein abstracted through a peak list.

Layer 1

*Database Setup.* The coverage of the database setup begins with re-capture of the FASTA sequence format. As presented in Figure 1 a FASTA entry consists of a header line preceding the sequence information. This header line is indicated by a ‘>’. This identified is followed by the sequence identifier and depending on the database of origin additional information. In case of the Genbank sequence database a FASTA header is set up the following way:

>gi|gi-number|gb|accession|locus

While “gi” and “gb” used for all Genbank entries, the other three fields allow the unique identification of an entry. The NCBI FASTA format

description suggests that each line of text in a FASTA file should be no longer than 80 characters and that all sequences are expected to contain the IUB/IUPAC (Irving, Freiser et al. 1978) amino acid codes.

Due to emerging bioinformatics programming packages (Bioperl (Mangalam 2002; Stajich, Block et al. 2002), BioJava (Mangalam 2002)), it is not necessary to analyze the parsing of a FASTA sequence entry. It can be assumed that the sequence itself is available as a variable with the header information stored in different variables. Under this simplifying assumption attention can be shifted to the detection of digestion sites. This should be done with a regular expression, by formatting of the digestion enzyme site string into the appropriate regular expression format of the programming language of choice. Upon detection of a digestive site, the preceding un-digestive string should be used to initialize the fragment sequence of a peptide object. At the time of peptide initialization the attributes for the peptide's accession number should also be set to the accession number of the protein, taken from the FASTA entry header (*accession* in case of the Genbank format).

After the completion of the digestion of all database entries, a protein identification can be conducted. It is advisable to store the digestion result and automate the digestion process and makes its renewed execution dependable from apparent updates to the sequence database. The storage of the peptides and the start of an optional update demon concludes the database setup process for single CPU usage.

## Layer 2

*Protein Identification.* The mechanisms needed for the protein identification layer consists of all three structural components. Since every protein identification is depending on a peak list derived through a mass spectrometry experiment, it is necessary to read it in as one of the first steps.

The main protein identification application initializes a peak list by setting a peak list file. The new peak list object then reads the information from the file and parses it's mass-to-charge ratios ( $m/z$ ) into an array.

For each of the  $m/z$  values in an array element a matching the existing peptide masses has to be conducted. The matching function is provided one  $m/z$  value at a time and retrieves the mass for each peptide at a time as well. Upon request through the peak list object the peptide object executes the mass calculation, described in the motivating example and returns said mass. The peptide mass and peak list  $m/z$  value are compared under

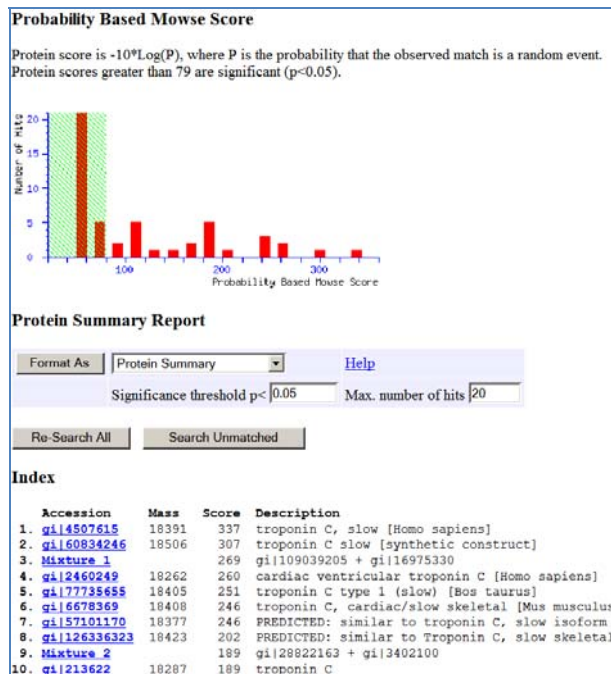
respect of an instrument error, which can be set for the peak list object through the protein identification application. If the two masses are equal within the instrument error range, the peptide is regarded as a match and is stored in the *matching\_results[]* array. The *matching\_results[]* contain the accession number of the protein that formed the origin of the peptide, the peak and peptide mass and their difference, as well as the peptide sequence.

The protein identification application retrieves and summarizes these matching results for each protein containing matched peptides. It then scores these results by applying different metrics, some of which will be presented in the “Known Uses”. For most of these metrics and also the final result presentation it is necessary to retrieve information about the complete protein, including its molecular weight, sequence and name.

#### Example Resolved

The pattern’s validity and performance is illustrated through the resolution of the motivating example. The example provides a proof of concept that said pattern can identify proteins.

The input peak list, generated by in-silico digestion from the protein troponin, is shown in Table 1. A submission of this peak list to the popular protein identification tool Mascot (Perkins, Pappin et al. 1999) returned the results presented in **Error! Reference source not found..** Mascot identified troponin as a possible match for this peak list and assigned a comparatively good score of over 300 to it. Mascot scores above 65-79, depending on the size of the sequence database, are deemed significant, thus marking troponin as the analyzed protein.



**Figure 6: Mascot search results. Troponin is correctly identified with the highest score under the use of the default Mascot parameters using the NCBI nr database.**

Analyzing the results returned besides the original troponin shows that they are either mixtures consisting of multiple proteins themselves or are manifestations of troponin in different organisms or organs. Mixture one is described as the combination of gi|109039205 and gi|16975330. A closer analysis of the genes involved shows that the first one is a troponin homolog in *M. mulatta* and the marks the structure of the troponin c-domain in *H. sapiens*. While both of these proteins point to troponin it is obvious that this result would be a false positive in a mass spectrometry experiment unless the sample would contain protein from different species. Mixture 2 has a similar problem, with the first protein identifying cardiac troponin in *D. rerio* and the second protein marking the regulatory domain of cardiac protein in *H. sapiens*.

Even this optimized example shows the necessity of an analysis of biological significance for a prediction. While this run can be seen as a first filter on the data, it is would now be possible to set more stringent search parameters. While it is obvious that we are dealing with troponin in this case it is still of value in more complex identification problems. The development of the result prediction results can be seen through the number of hits diagram created by Mascot. Figure illustrates how the number of significant proteins decreases with more stringent parameters. While about twenty proteins are reported with a significant score in a

default search, this number is reduced to six when the taxonomy is limited to Human. Setting the missed cleavages to 0 eliminates all but one significant protein. This protein is still our original protein but contains a much lower score now, which is due to a smaller peptide database that does not have to account for the fragment sequences that contain a missed cleavage site.

A parameter restriction can be very valuable for identification results with many significant predictions, but the decrease in false positives has to be balanced to a possible increase in false negatives, in order not to miss the correct protein.

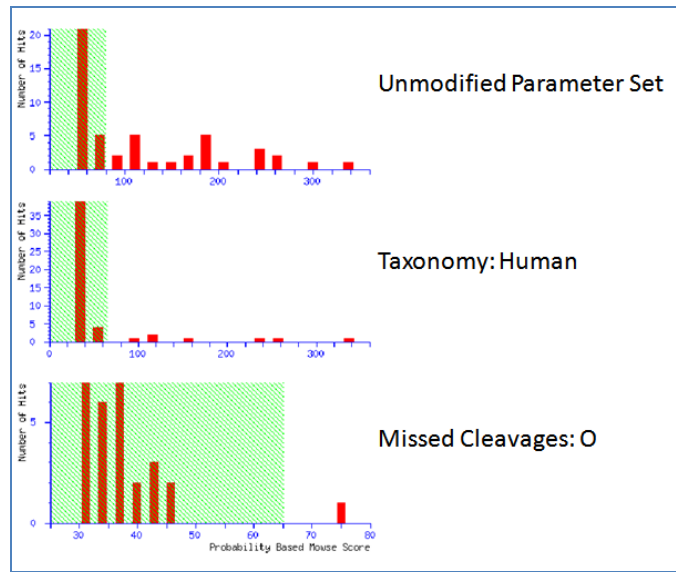


Figure 7: Result development in the troponin prediction for different parameter sets.

Known uses

**Biomarker Discovery.** A biomarker is a molecule injected into tissue to determine its biological state in order to determine if an organism is healthy or suffers from a marker specific disease. Biomarker discovery is the process of detecting biomarkers for a particular biological state, mainly disease state. The biomarker indicates changes in the expression of protein that is linked to the progression of a specific disease. Protein identification thus plays an integral part in biomarker discovery. A biomarker can be used to diagnose the manifestation of or susceptibility for a disease. Biomarker research is also the preliminary step in the development of disease treatments. By understanding the actors and interactions in a disease it is possible to derive methods to interrupt this process and thus cure or hamper a disease.

**Proteomics Software Toolkits.** Based on the work on the mass database OWL Pappin presented in (Bleasby, Akrigg et al. 1994) one of the first protein identification algorithms, called “*MOWSE*” (Pappin, Hojrup et al. 1993) to use fragment masses for the rapid identification of proteins based on the number of matches between the mass spectrometry fingerprint and the database fingerprint of the protein. *MOWSE* gave way to numerous algorithms still in use, focused on the returned number of matches, among them the Emboss open source implementation *eMOWSE* (Rice, Longden et al. 2000). While the MS-Fit approach developed by Clauser extended *MOWSE* in (Clauser, Baker et al. 1999) by analyzing the role of more accurate mass measurements it is still very close to the original approach. MS-Fit and *MOWSE* create a frequency factor matrix in order to score the peptide data used in the identification process. Peptides masses are set up as rows of 100 Da intervals, while 10 kDa intervals for the masses of the intact proteins are used to identify the columns of the matrix. A peptide is thus not only stored based on its weight but also based on the mass of its originating protein. The elements of each column are divided by the largest element of the column in order to create a *MOWSE* factor matrix  $M$ . The score for a protein is created using the frequency  $m_{i,j}$  for each of its  $n$  matched peptides and the mass of the intact protein  $M_{Prot}$  and is normalized for 50 kDa:

$$s = \frac{50.000}{M_{Prot} * \prod_n m_{i,j}}$$

The protein identification in the two examples was conducted using the *Mascot* protein identification tool by Matrixscience (Perkins, Pappin et al. 1999). *Mascot* is a probability based identification approach based on *MOWSE* designed to predict protein with a very low number of false positives and negatives. While the basic probability model of *Mascot* is unpublished, the score calculation, based on a protein’s probability to be a correct match, is formalized as:

$$s = -10 \log(P)$$

A significance threshold is presented to help in the detection of false positives. This significant threshold is calculated using the complete number  $N$  of proteins in the database and by default a probability of 0.05, providing 95% confidence to predictions above this threshold. In the example shown in the following equation the 95% confidence threshold of a database consisting of 1.5 million proteins is calculated as 74.7, meaning that in this database scores larger than 75 are regarded significant.

Mascot is among the most popular protein identification tools at the time. Not only does it provide a free and easily accessible web-interface for simple identification, it is also exceptional fast. This speed is due to a highly distributed implementation. The Mascot web-server provides an interface for a 28 node cluster, allowing search times of few seconds for peak lists of varying length.

*Profound* (Zhang and Chait 2000) marks a probability based approach, focused on Bayesian Inference. The hypothesis that protein  $k$  represents the protein being analyzed is tested for each protein in the database. Based on experimental data  $D$ , which symbolizes the peak data, and available background information  $I$  about possible molecular weight or iso-electric point can be used to derive under usage of Baye's theorem and the Maximum Entropy the following formula that assigns a probability to each database protein:

$$P(k|DI) \propto \frac{(N-r)!}{N!} \prod_{i=1}^r \left\{ \sqrt{\frac{2}{\pi}} \frac{m_{\max} - m_{\min}}{\sigma_i} \times \sum_{j=1}^{g_i} \left[ -\frac{(m_i - m_{ij0})^2}{2\sigma_i^2} \right] \right\} F_{\text{Pattern}}$$

Profound defines the elements of this formula in the following way:

- N: Number of theoretical fragments generated by fragmentation
- r: number of hits between measured and calculated masses
- $(m_{\max} - m_{\min})$ : range of measured peptide masses
- $m_i$  : measured masses of hit  $i$
- $g_i$ : number of theoretical peptides that match  $i$
- $m_{ij0}$ : calculated mass of the  $j^{\text{th}}$  peptide in the  $i^{\text{th}}$  hit
- $F_{\text{pattern}}$ : empirical term to control missed cleavages

## Consequences

The protein identification pattern provides two distinct **benefits**:

*Database Setup.* As a requirement for protein identification a mass database has to be created. The presented pattern describes the generation of such a database based on existing protein sequence databases.

*Identification.* The protein identification pattern provides a solution to the problem of detecting corresponding proteins for a specified peak list from a mass database. Scoring methods are used to evaluate the matched proteins and determine the confidence that a detected protein is the protein apparent in the analyzed protein sample that gave rise to the peak list.

The protein identification pattern has the following **liabilities**:

*Lack of proteins.* Proteomic sequencing is an ongoing effort. While the sequencing of the Human genome was finished in 2003, the Human proteome is not completely sequenced, which is due mainly to alternative splicing of protein coding genes, making the in-silico sequencing of protein based on genomic information impossible. This is a liability that will be overcome over time however.

*Errors in existing databases.* While protein sequencing research is done around the world in respectable laboratories, the submission of protein sequences and their annotation is often conducted by personnel that is not directly involved in the research and is susceptible to erroneous entries. This leads to many wrongly annotated proteins in the sequence databases and constant revisions of database entries. Rigorous quality of service protocols have to be enforced by the sequence database providers to overcome this liability.

*Multiple possible identities.* Since mass spectrometers have error ranges it is not possible to calculate perfect peak lists. This leads to the inclusions of error in the matching process and a multitude of false positive predictions. Some proteins are very similar in their mass fingerprint and can thus not be distinctively identified.

See Also

No similar patterns are known at the moment.

Credits

Thanks to John J. Kopchick's proteomics research group at the Edison Biotechnology Institute. The team consisting of Shigeru Okada, Lucila Sackmann-Sala and Sudha Sankaran provided a great introduction to the area of proteomics and has been a very patient tester and user of various implementations of this pattern.

Thanks as well to Dazhang Gu, Thomas Conley and the CIDDS writer's workshop participants, who were involved in the testing and deployment of our protein identification solutions.

## References

- Benson, D. A., I. Karsch-Mizrachi, et al. (2006). GenBank. **34**: D16-20.
- Bleasby, A. J., D. Akrigg, et al. (1994). "OWL - a non-redundant composite protein sequence database." Nucleic acids research **22**(17): 3574-3577.
- Clauser, K. R., P. Baker, et al. (1999). "Role of accurate mass measurement (+/- 10 ppm) in protein identification strategies employing MS or MS/MS and database searching." Analytical Chemistry **71**(14): 2871-2882.
- Irving, H. M. N. H., H. Freiser, et al. (1978). Compendium of analytical nomenclature : definitive rules 1977. Oxford ; New York, Pergamon Press.
- Kaiser, J. (2002). "Proteomics. Public-private group maps out initiatives." Science **296**(5569): 827.
- Liebler DC (2002) "Introduction to Proteomics: tools for the new biology" Humana Press.
- Mangalam, H. (2002). "The Bio\* toolkits--a brief overview." Brief Bioinform **3**(3): 296-302.
- Pappin, D. J. C., P. Hojrup, et al. (1993). "Rapid identification of proteins by peptide-mass fingerprinting." Current biology : CB **3**(6): 327-332.
- Perkins, D. N. (2006). MSDB. **2007**.
- Perkins, D. N., D. J. C. Pappin, et al. (1999). "Probability-based protein identification by searching sequence databases using mass spectrometry data." Electrophoresis **20**(18): 3551-3567.
- Rice, P., I. Longden, et al. (2000). "EMBOSS: The European Molecular Biology Open Software Suite." Trends in Genetics **16**(6): 276-277.
- Schmidtman, A., C. Lindow, et al. (2005). Cardiac troponin C-L29Q, related to hypertrophic cardiomyopathy, hinders the transduction of the protein kinase A dependent phosphorylation signal from cardiac troponin I to C. **272**: 6087-6097.
- Stajich, J. E., D. Block, et al. (2002). "The Bioperl toolkit: Perl modules for the life sciences." Genome Res **12**(10): 1611-8.
- Wu, C. H., R. Apweiler, et al. (2006). The Universal Protein Resource (UniProt): an expanding universe of protein information. **34**: D187-191.
- Zhang, W. Z. and B. T. Chait (2000). "Profound: An expert system for protein identification using mass spectrometric peptide mapping information." Analytical Chemistry **72**(11): 2482-2489.