mzIdentML

exchange format for peptides and proteins identified from mass spectra

Table of Contents

Preface

Status of This Document

This document presents a final specification for the mzIdentML data format developed by the HUPO Proteomics Standards Initiative. Distribution is unlimited.

Version of This Document

The current version of this document is: version 1.3.0 final, June 2024.

Date created: June 24, 2024

Last updated: *BUILDDATE*

Based on commit: *COMMITHASH* - Commit History

The current version of this document is: version version 1.3.0 final, June 2024

The latest (draft) version of this document may be found at https://github.com/HUPO-PSI/mzTab.

Type of This Document

This document is a *recommendation* for a common, community-driven standard data exchange format in proteomics.

Authors

Please see *Authors and Contributors* for details on the authors and editors of this document.

Abstract

The Human Proteome Organisation (HUPO) Proteomics Standards Initiative (PSI) defines community standards for data representation in proteomics to facilitate data comparison, exchange and verification. The Proteomics Informatics Working Group is developing standards for describing the results of identification and quantitation processes for proteins, peptides and protein modifications from mass spectrometry. This document defines an XML schema that can be used to describe the outputs of proteomics search engines or similar software for peptide/protein identification from mass spectrometry (MS) data.

Introduction

Background

This document addresses the systematic description of (poly)peptide identification and characterisation based upon mass spectrometry (MS). A large number of different proteomics search engines are available that produce output in a variety of different formats. It is intended that mzIdentML will provide a common standard format for identification results to support a range of scenarios encountered in proteome informatics. mzIdentML has been developed with a view to supporting the following general tasks (more specific use cases are provided in Section 2.):

1. *The discovery of relevant results,* so that, for example, data sets in a database that use a particular technique or combination of techniques can be identified and studied by experimentalists during experiment design or data analysis.

2. *The sharing of best practice,* so that, for example, analyses that have been particularly successful at identifying a certain group of peptides/proteins can be interpreted by consumers of the data.

3. *The evaluation of results*, so that, for example, sufficient information is provided about how a particular analysis was performed to allow the results to be critically evaluated.

4. *The sharing of data sets,* so that, for example, public repositories can import or export data, or multi-site projects can share results to support integrated analysis.

5. *The creation of a format for input to analysis software,* for example, allowing software to be designed that provides a meta-score over the output from several search engines.

6. *An internal format for pipeline analysis software, for example, allowing analysis software to store intermediate results from different stages of an identification pipeline, prior to the final results being assembled in a single mzIdentML file.*

This document presents a specification, not a tutorial. As such, the presentation of technical details is deliberately direct. The role of the text is to describe the model and justify design decisions made. The document does not discuss how the models should be used in practice, consider tool support for data capture or storage, or provide comprehensive examples of the models in use. Tutorial material can be located on the PSI website (see http://www.psidev.info/mzidentml).

Document Structure

The remainder of this document is structured as follows. Section 2. lists use cases mzIdentML is designed to support. Section 3. describes the terminology used. Section 4. describes how the specification presented in Section 4. relates to other specifications, both those that it extends and those that it is intended to complement. Section 6. contains the documentation for the XML schema which is generated automatically and several parts of the schema are documented in more detail in Section 7. Conclusions are presented in Section 8..

Use Cases for mzIdentML

The following use cases have driven the development of the mzIdentML data model and XML schema, and are used to define the scope of the format in the current version.

It should be possible to create a tool that loads an mzIdentML document and enables users to examine results from an MS or MSn analysis. As of mzIdentML version 1.3.0, there is support for aggregating evidence from multiple MS levels by using the encoding for identifications based on multiple spectra given in Section 7.11. There should be sufficient information for the tool to generate output reports that conform to the requirements made by journals for publication and that conform to the relevant MIAPE guidelines. For example:

· For a Peptide Mass Fingerprint (PMF) search, it should be possible to display the spectrum and show the matches of the peaks to the relevant peptides.

· For an MS/MS search, it should be possible to locate which spectrum matched to which peptide in the original file.

1. There should be sufficient information stored in the instance document to enable a user to run the same or a similar search on the same or another search engine. This means that all search parameters should be described in sufficient detail and that sufficient information is available to determine which database (if any) the data were searched against. The peak lists data do not need to be included in the instance document, but do need to be suitably referenced.

2. It should be possible to save the results of searching a decoy database in the same instance document as the results from the target database. It should then be possible to write a viewer application that enables a user to investigate the effect of changing, for example, a threshold value on the false discovery rate. This would only be possible if results that are generally considered lower quality from the search are also saved in the mzIdentML document (rather than just top matches) and if the results from the decoy search are also saved. It would only be possible to do this at the peptide level for an MS/MS search, because changing thresholds would normally have some effect on the protein grouping algorithm.

3. It should be possible to save manual or automated annotation of proteins/peptides in an instance document. A third-party tool could be used to save annotations and validations of identified proteins/peptides to an existing instance document.

4. It should be possible to save the results from a search of a metabolically labelled sample. For example, with a 14N/15N experiment, two separate sets of amino acid masses are used, and it must be possible to tell which masses were used for each peptide result.

5. For a search of multiple peak lists, it should be possible to identify the spectrum that matched a particular peptide or protein reported by the search engine. For example, in an LC-MS/MS run, it should be possible to refer back to the spectrum in the peak list file that was searched and from there, if the information is available, to be able to determine the retention time of the spectrum.

6. It should be possible to search a file to retrieve all molecules that have a specified modification.

7. It should be possible to store the results of a search of spectra against other spectra – *i.e.*, a spectral library search.

8. It should be possible to store the results of a top-down search, *i.e.*, analysis of complete proteins.

9. Support should be provided for storing fragmentation data so that for example viewers could display which ions in the input data match predicted ion fragment masses.

10. There should be support for storing the results of searches of peptides against nucleic acid databases, including the information about which translation frame the matches were found in.

11. It should be possible to combine the results from multiple search engines into one mzIdentML document. For example, the peptide spectrum matches (PSMs) from two or more different search engines could be combined using a third tool to give one set of protein results.

12. It should be possible to store *de novo* peptide sequencing results, to the extent that it will be possible to enumerate and record all possible matches found by a *de novo* technique. However, we anticipate that this can produce large files.

13. It should be possible to store the results of MS/MS crosslinking approaches, whereby two peptides ed using chemical reagents or biologically occurring modifications have been identified (newly added in mzIdentML 1.2). New use cases for crosslinking data have been added in mzIdentML 1.3. From this version, this information is available in the mzIdentML crosslinking extension document, and not in this main specification document.

14. It should be possible to store at a basic level of detail the molecular interaction data that can be inferred from crosslinking approaches (newly added in mzIdentML 1.2).

15. It should be possible to represent statistical values or scores associated with the positions of modifications on a peptide chain (newly added in mzIdentML 1.2).

16. It should be possible to represent statistical values or scores associated with peptide identifications, formed from groups of redundant peptide-spectrum matches (PSMs) reporting on the same peptide (newly added in mzIdentML 1.2).

17. It should be possible to capture the output of proteogenomics analyses such as the mapping of peptides to gene models and chromosomes (newly added in mzIdentML 1.2).

There should be limited support for:

1. Sequence tagging, in which short sequences defined by a *de novo* process are used to characterize spectra. The final results from a sequence-tag-filtered search can be stored in mzIdentML, but the details of tag generation and filtering cannot.

The following use cases will not be supported in version 1.2 of mzIdentML:

1. It should be possible to store relative and absolute quantitation information at the peptide and protein level using all the popular techniques – this is captured in mzQuantML and also in mzTab.

2. Support for complex workflows where multiple data processing algorithms are combined in a pipeline, *i.e.*, only "final" results are represented in mzIdentML v1.2 in one protein list.

Concepts and Terminology

This document assumes familiarity with XML Schema notation (www.w3.org/XML/Schema). The key words "MUST," "MUST NOT," "REQUIRED," "SHALL," "SHALL NOT," "SHOULD," "SHOULD NOT," "RECOMMENDED," "MAY," and "OPTIONAL" are to be interpreted as described in RFC-2119 (http://www.ietf.org/rfc/rfc2119.txt).

Relationship to Other Specifications

The specification described in this document is not being developed in isolation; indeed, it is designed to be complementary to, and thus used in conjunction with, several existing and emerging models. Related specifications include the following:

1. *MIAPE MSI* (http://www.psidev.info/groups/miape) The Minimum Information About a Proteomics Experiment: Mass spectrometry Informatics document defines a checklist of information that should be reported about such a study.

2. *mzML* (http://www.psidev.info/mzml). mzML is the PSI standard for capturing mass spectra / peak lists resulting from MS in proteomics. It is RECOMMENDED that mzIdentML should be used in conjunction with mzML, although it is possible to use mzIdentML with other formats of mass spectra. This document does not assume familiarity with mzML (1).

3. *mzQuantML* (http://www.psidev.info/mzquantml). mzQuantML is the PSI standard for capturing quantitative proteomics data from MS (2) mzQuantML files that report quantitative data MAY reference mzIdentML files containing the detailed identification data.

4. *mzTab* (http://www.psidev.info/mztab). mzTab is the PSI standard that can represent identification and quantification results at different levels of detail, in a tab-delimited format (3). mzTab results can represent a subset of all the information included in a mzidentML file. However, in mzTab 1.0 while there is support for some of the new features represented in mzidentML 1.2 (e.g. ambiguity in the modification position), other features are not explicitly supported (e.g. crosslinking). mzTab files MAY reference mzIdentML files.

5. *PSI-MI XML / MITAB* (http://www.psidev.info/groups/molecular-interactions). The PSI has developed specifications for molecular interaction evidence in XML format (PSI-MI) and tab-separated (MITAB). When MS crosslinking data is stored in mzIdentML, including evidence for protein interactions, the file can act in a complementary manner (e.g. providing source data) to a PSI-MI (4) or MITAB file.

The PSI Mass Spectrometry Controlled Vocabulary (CV)

The PSI-MS controlled vocabulary (CV) (5) is intended to provide terms for annotation of mzML, mzIdentML, mzQuantML and other PSI standard file formats. The CV has been generated by collection of terms from software vendors and academic groups working in the area of MS and proteome informatics. Some terms describe attributes that must be coupled with a numerical value attribute in the <cvParam> element (e.g. MS:1001191 "p-value") and optionally a unit for that value (e.g. MS:1001117, "theoretical mass", units = dalton). The terms that require a value are denoted by having a "datatype" key-value pair in the CV itself: MS:1001172 "mascot:expectation value" value-type:xsd:double. Terms that need to be qualified with units are denoted by having a "has_units" key in the CV itself (relationship: has_units: UO:0000221 ! dalton). The details of which terms are allowed or required in a given schema section is reported in the mapping file (Section 4.2).

As recommended by the PSI CV guidelines, psi-ms.obo should be dynamically maintained *via* the psidev-ms-vocab@lists.sourceforge.net mailing list that allows any user to request new terms, in agreement with the community involved. Once a consensus is reached among the community the

new terms are added within a few business days.

In general, modifications SHOULD be sourced from Unimod (http://www.unimod.org/obo/unimod.obo) where possible. For encoding crosslinking results, the XLMOD-CV SHOULD be used (https://raw.githubusercontent.com/HUPO-PSI/mzIdentML/master/cv/XLMOD.obo), unless suitable terms can be obtained from Unimod at a later date.

The following ontologies or controlled vocabularies specified below may also be suitable or required in certain instances:

- Unit Ontology (http://bioportal.bioontology.org/ontologies/UO?p=classes&conceptid=root).
- ChEBI (http://www.ebi.ac.uk/chebi/).
- For describing sample types, any suitable and stable ontologies MAY be used.

The PSI Protein modifications CV (http://psidev.cvs.sourceforge.net/viewvc/psidev/psi/mod/data/PSI-MOD.obo) is now DEPRECATED for use in mzIdentML. It is RECOMMENDED to use Unimod wherever possible.

Validation of controlled vocabulary terms

The correct usage of controlled vocabulary terms within mzIdentML is governed by the use of a mapping file that defines each XML location (XPath) where a <cvParam> instance can be used, and the allowed terms from the PSI-MS, or other CVs. The mapping file is read and interpreted by validation software, checking that the data annotation is consistent. The mapping file needs to be checked and updated when the structure of the CV is changed, and in some instances when new terms are added to the CV. XML paths are associated with CV terms along with a requirement level (MAY, SHOULD or MUST) defining what should be reported by validation software if one of the mapped terms is not provided in an instance document. Syntactic and semantic validation SHOULD be checked using the official Java-based validator available from http://www.psidev.info/mzidentml.

Changes from version 1.1.0 to version 1.2.0

The primary update requiring the change from version 1.1.0 to version 1.2.0 is in the inclusion of guidelines for encoding protein group results (Section 5.2.1). Several examples referenced throughout the document are annotated with version 1.1.0. In these cases, it can be assumed that these files are also valid 1.2.0 files, since they do not include protein inference results. Other minor changes have been made to the specification since version 1.1.0, with regards to the encoding of specific workflows – notably searches where pre-fractionation has been performed (Section 5.2.5), searches employing multiple search engines (Section 5.2.4), *de novo* sequencing (Section 5.2.2) and spectral library searches (5.2.3).

Several new CV terms are now mandatory within the <SpectrumIdentificationProtocol> element - enabling the new features to be differentiated and recognised automatically by processing software, as follows. **1...*n* of the following terms MUST be present:**

CV term name	Accession	Reference to section within this document
peptide-level scoring	MS:1002490	5.2.7
modification localization scoring	MS:1002491	5.2.8
consensus scoring	MS:1002492	5.2.4
sample pre-fractionation	MS:1002493	5.2.5
crosslinking search	MS:1002494	5.2.9
de novo search	MS:1001010	5.2.2
spectral library search	MS:1001031	5.2.3
proteogenomics search	MS:1002635	5.2.10
no special processing	MS:1002495	Used to indicate that none of the above (new) features have been included in the file.

Table 1 New CV terms now mandatory (1...n terms MUST be present) within the <SpectrumIdentificationProtocol> element in mzIdentML 1.2. Terms "de novo search" and "spectral library search" MUST appear under the <SearchType> element. All other terms MUST appear under the <AdditionalSearchParams> element.

Changes from version 1.2.0 to 1.3.0

The core of this specification document remains unchanged from 1.2.0, but version 1.3.0 now supports extensions for additional features or use cases, described in additional extension documents located in the same folder. At the time of writing, an extension for crosslinking data exists and an extension for glycopeptides is in progress, but others may take place in the future.

It is important to highlight that extension documents cannot introduce any changes to the mzIdentML schema. As such they are primarily about introducing and describing the use of CV terms.

Implementers only need to incorporate the extensions if supporting the specific extra features or use cases described there. A document signals which extensions it requires by including <cvParam> elements that are children of the term MS:1003373, "mzIdentML extension version", immediately after the <cvList> element inside the <MzIdentML> element.

This is the only change to the XML schema definition for mzIdentML 1.3.0: <cvParam> elements can be included immediately after the <cvList> element inside the top level <MzIdetnML> element, to permit declaring the version of extension documents.

A new section has also been added to explain how to encode identifications coming from multiple spectra (Section 7.11). The "combined spectra" type of input file format from version 1.2.0 has been retired and is not part of the 1.3.0 specification.

Additionally, two new CV terms have been introduced to provide an optional mechanism for

linking the <Modification> elements inside <Peptide> elements to <SearchModification> elements (Section 7.12). These are:

- "search modification id" (MS:1003392),
- "search modification id ref" (MS:1003393).

Furthermore, different typos and small details have been refined throughout the text. As an example, the hyphenated term "cross-linking" (used in version 1.2.0) has now been de-hyphenated throughout this 1.3.0 updated specification document.

Finally, in the section 5.2.2 ("Support for de novo sequencing approaches"), a clarification has been added providing more details about how to represent *de novo* searches, including the use of the CV param "*de novo* search or no database used" (MS:1000394).

Format scope and specific use cases

Handling updates to the controlled vocabulary

In brief, when a new term is required, the file producers must contact the CV working group *via* e-mail (psidev-ms-vocab@lists.sourceforge.net) and request the new term. It is anticipated that problems may arise if a consumer of the file encounters a new CV term and they are not working from the latest version of the CV file. It has been decided that rather than aim for a workaround to this issue, it can be expected that data file consumers must ensure that the OBO file is up-to-date.

Identifying the input spectrum for each result

A <SpectrumIdentificationResult> is linked to the source spectrum (in an external file) from which the identifications are made by way of a reference in the spectrumID attribute and *via* the <SpectraData> element which stores the URL of the file in the location attribute. It is advantageous if there is a consistent system for identifying spectra in different file formats. The following table is implemented in the PSI-MS CV for providing consistent identifiers for different spectrum file formats. A CV term MUST be imported into the <SpectraData> element to demonstrate which system for identifying input spectra is being used in the spectrumID attribute of <SpectrumIdentificationResult>.

It is encouraged but not mandatory that a valid mzIdentML file is accompanied by the set of spectra that were searched. It is acknowledged that in many cases it will be useful to have an mzIdentML file and the input spectra together, there are practical problems processing such data depending on the spectrum file format (e.g. in case of proprietary formats), and cases where an mzIdentML is useful even if the spectra data is not available.

Update from version 1.2.0:

Version 1.1.0 of the specification document states "the CV holds the definite specification for legal encodings of spectrumID values". In version 1.2, the only legal ways of referencing a spectrum identification format are provided below in Table 1. Any new spectral formats that cannot fit into this schema require an update to this document.

Update included in version 1.3.0:

All CV terms containing combined spectra input types have been deprecated.

ID	Term	Data type	Comment
MS:1000768	Thermo nativeID format	controllerType=xsd:nonNegativeInteger controllerNumber=xsd:positiveInteger scan=xsd:positiveInteger	controller=0 is usually the mass spectrometer. Space-separated values.

ID	Term	Data type	Comment
MS:1000769	Waters nativeID format	function=xsd:positiveInteger process=xsd:nonNegativeInteger scan=xsd:nonNegativeInteger	Space-separated values.
MS:1000770	WIFF nativeID format	sample=xsd:nonNegativeInteger period=xsd:nonNegativeInteger cycle=xsd:nonNegativeInteger experiment=xsd:nonNegativeInteger	Space-separated values.
MS:1000771	Bruker/Agilent YEP nativeID format	scan=xsd:nonNegativeInteger	
MS:1000772	Bruker BAF nativeID format	scan=xsd:nonNegativeInteger	
MS:1000773	Bruker FID nativeID format	file=xsd:IDREF	The nativeID must be the same as the source file ID.
MS:1000774	multiple peak list nativeID format	index=xsd:nonNegativeInteger	Used for referencing peak list files with multiple spectra, i.e. MGF, PKL, merged DTA files. Index is the spectrum number in the file, starting from 0.
MS:1000775	single peak list nativeID format	file=xsd:IDREF	The nativeID must be the same as the source file ID. Used for referencing peak list files with one spectrum per file, typically in a folder of PKL or DTAs, where each sourceFileRef is different.
MS:1000776	scan number only nativeID format	scan=xsd:nonNegativeInteger	Used for referencing mzXML, or a DTA folder where native scan numbers can be derived.

ID	Term	Data type	Comment
MS:1000777	spectrum identifier nativeID format	spectrum=xsd:nonNegativeInteger	Used for referencing mzData. The spectrum ID attribute is referenced.
MS:1001530	mzML unique identifier	xsd:string	Used for referencing mzML. The value of the spectrum ID attribute is referenced directly.

Table 2. CV terms and rules implemented in the PSI-MS CV for formulating the "nativeID" to identify spectra in different file formats.

In mzIdentML, the spectrum ID attribute should be constructed following the data type specification in Table 2. As an example, to reference the third spectrum (index=2) in an MGF (Mascot Generic Format) file:

<SpectrumIdentificationResult id="Res1" spectrumID="index=2" SpectraData_ref="InputSpectra1">

... <SpectraData location="local/mgf/merge.mgf" id="SD_1" >

<FileFormat>

<cvParam accession="MS:1001062" name="Mascot MGF file" cvRef="PSI-MS" />

</FileFormat>

<SpectrumIDFormat>

<cvParam accession="MS:1000774" name="multiple peak list nativeID format" cvRef="PSI-MS" />

</SpectrumIDFormat>

</SpectraData>

Comments on Specific use cases

Many special use cases for mzIdentML were considered during its development. Each of these use cases has a corresponding example file that exercises the relevant part of the schema and provides a reference implementation example (see supporting documentation). Authors of software that create mzIdentML are encouraged to examine the examples that accompany this format release before implementing the writer. Further, such authors are encouraged to use the validator before releasing any new writer code and working with the PSI PI Working Group to resolve any issues. In the subsections below, we comment on a few of the notable use cases that were considered – in particular those newly added in mzIdentML 1.2.

Protein grouping encoding

This section is newly inserted in the mzIdentML version 1.2.0 specifications. In version 1.1.0, CV

terms had been proposed for representing set relationships between different proteins within groups, but there was not a requirement that particular terms were used. A given data structure from software could be mapped onto the hierarchy <ProteinAmbiguityGroup> and <ProteinDetectionHypothesis> in mzIdentML in different ways, leading to difficulties for data consumers. As such, a working group has now agreed a more rigid encoding detailed as follows and in (6).

1. As in mzIdentML version 1.1, a single protein accession that has been cited by software is captured in mzIdentML in <ProteinDetectionHypothesis> (PDH).

    a. A PDH MAY contain scores or statistical values produced by the export software, encoded as CV terms.

2. A "protein group", representing a "biological entity" for which the software claims independent evidence is present, MUST be mapped onto <ProteinAmbiguityGroup> (PAG).

    a. A PAG MAY have additional scores produced by the export software, encoded as CV terms.

3. The reporting of protein identification thresholds is now mapped onto PAGs. There is no desire to change the core XML Schema Document (XSD) for mzIdentML and as such, a new CV term "protein group passes threshold" value= "xsd:boolean" MUST be present on every PAG (MS:1002415). If no thresholding has been done by the software, all protein groups MUST be annotated as "protein group passes threshold" value= "true".

    a. The attribute *passThreshold* = "true|false" remains present on PDH and MAY be used if software packages wish to report a two-level hierarchy of thresholds applied. However, it is not expected that consuming software will use this attribute to determine which proteins have been reported as identified.

4. The <ProteinDetectionList> MUST contain the CV term "count of identified proteins" value= "xsd:integer" (MS:1002404). The value MUST be derived from the count of PAGs passing the threshold reported in the file and will be checked by validation software.

5. Few software packages report "protein clusters" at present, but for those packages that wish to report clusters, a CV term "cluster identifier" value = "xsd:integer" SHOULD be used (MS:1002407). The integer identifier MUST be shared by all PAGs belonging to the same cluster. An optional term "count of identified clusters" value = "xsd:integer" (MS:1002406) MAY be annotated on the <ProteinDetectionList>.

6. Every PDH MUST be annotated as either a "leading protein" (MS:1002401) or a "non-leading protein" (MS:1002402), as defined in Table 2, within a PAG. This recommendation thus makes it explicit for consuming software whether one or more proteins have stronger evidence than others in the group (see Table 2 for examples).

    a. An additional term, "group representative" (MS:1002403) MAY be used to annotate one PDH, which is also flagged as a "leading protein", if the export software wishes to enforce that only one of potential several "leading proteins" will be interpreted by the consuming software as the representative of the group, for example acting as a tiebreaker.

    b. If the export software does not explicitly flag one protein as the "group representative", it is assumed that if consuming software requires a single accession to represent the group, an arbitrary choice will be made (among "leading proteins" only if these exist).

7. Any PDHs MAY be annotated with terms present in the CV for spectrum/sequence same-set, spectrum/sequence subset, spectrum/sequence subsumable, marginally distinguished and so on

(Table 2).

    a. A PDH MAY be annotated with more than one of these terms if appropriate to describe the complex set relationships that exist within a group.

    b. Developers of software packages MAY propose additional terms for describing group membership of PDHs, which will be incorporated into the CV.

    c. The associated value for these CV terms MAY be used to annotate which PDH(s) are the super/same-set of the annotated PDH.

    d. There is no expectation that consuming software should be aware of these terms, but they may be useful in internal pipeline or visualization software packages that are specifically designed to work with this terminology set.

8. Some PDHs could be mapped to more than one PAG, for example where proteins are multiply subsumed. To capture these cases, multiple PDHs in different PAGs MAY reference the same <DBSequence>.

The semantic validation software has been updated to encode these rules and report errors ("MUST" rule), warnings ("SHOULD" rule) or informational messages ("MAY" rule).

mzIdentML context	CV term	Values	Require-ment level	Description
ProteinDetection-List	count of identified proteins	xsd:integer	MUST	The value reported MUST equal the number of PAGs with "protein group passes threshold" value = "true"
ProteinDetection-List	count of identified clusters	xsd:integer	MAY	If protein clusters have been reported in the file, the exporter may choose to annotate the ProteinDetectionList with the number identified above threshold.

mzIdentML context	CV term	Values	Require-ment level	Description
ProteinAmbiguity-Group	number of distinct protein sequences	xsd:integer	MAY	The number of distinct protein sequences among the PDHs in the group. For example, if there are two PDHs with different identifiers that have identical full length sequences, the value would be 1.
ProteinAmbiguity-Group	cluster identifier	xsd:integer	MAY	An identifier applied to protein groups to indicate that they are linked by shared peptides.

mzIdentML context	CV term	Values	Require-ment level	Description
ProteinDetection-Hypothesis	leading protein OR non-leading protein	-	MUST OR MUST	Every PDH in each PAG MUST be flagged as a leading protein or a non-leading protein and each PAG MUST contain at least one leading protein, but MAY contain more than one. A "leading protein" is defined as a protein that has the strongest or near strongest (further explained in Table 3) set of evidence for being present in the sample studied, amongst the grouped protein accessions. A "non-leading protein" is defined as a protein that has (substantially) less evidence than other proteins within the same group, and is thus less likely to have been present in the sample studied.
ProteinDetection-Hypothesis	leading protein 	-	MUST 	

mzIdentML context	CV term	Values	Require-ment level	Description
ProteinDetection-Hypothesis	group representative	-	MAY	Each PAG MAY contain zero or one PDH flagged as the group representative, if the software wishes to flag a preference (often arbitrary or for example based on alphabetical ordering) amongst the leading proteins. The group representative term can thus be viewed a "tiebreaker" if the export software wishes to make this distinction.
ProteinAmbiguity-Group	protein group passes threshold	xsd:Boolean	MUST	Each PAG MUST be annotated with a Boolean CV term indicating whether the PAG has passed the threshold reported in the ProteinDetectionProtocol.
ProteinDetection-Hypothesis	sequence same-set protein	xsd: "list_of_strings" space separated list of PDH IDs that are same-set.	MAY	A protein that is indistinguishable or equivalent to another protein in the group, having matches to an identical set of peptide sequences.

mzIdentML context	CV term	Values	Require-ment level	Description
ProteinDetection-Hypothesis	spectrum same-set protein	xsd: "list_of_strings" space separated list of PDH IDs that are same-set.	MAY	A protein that is indistinguishable or equivalent to another protein in the group, having PSMs derived from the same set of spectra.
ProteinDetection-Hypothesis	sequence sub-set protein	xsd: "list_of_strings" space separated list of PDH IDs that are super-set.	MAY	A protein for which the matched peptide sequences are a subset of the matched peptide sequences for another protein in the group.
ProteinDetection-Hypothesis	spectrum sub-set protein	xsd: "list_of_strings" space separated list of PDH IDs that are super-set.	MAY	A protein for which the matched spectra are a subset of the matched spectra for another protein in the group.
ProteinDetection-Hypothesis	sequence multiply subsumable protein	xsd: "list_of_strings" space separated list of PDH IDs that subsume this PDH.	MAY	A protein for which the matched peptide sequences are the same, or a subset of, the matched peptide sequences for two or more other proteins combined. These other proteins need not all be in the same group.

mzIdentML context	CV term	Values	Require-ment level	Description
ProteinDetection-Hypothesis	spectrum multiply subsumable protein	xsd: "list_of_ strings" space separated list of PDH IDs that subsume this PDH.	MAY	A protein for which the matched spectra are the same, or a subset of, the matched spectra for two or more other proteins combined. These other proteins need not all be in the same group.
ProteinDetection-Hypothesis	marginally distinguished protein	-	MAY	Assigned to a non-leading PDH that has some independent evidence to support its presence relative to the leading protein(s) e.g. the PDH may have a unique peptide but not sufficient to be promoted as, for example, a leading protein of another a PAG.

Table 3. New CV terms for reporting protein set (group) relationships and global statistics about the protein identification results. The semantic validation software for mzIdentML version 1.2.0 reports an error (MUST), a warning (SHOULD) or an informational message (MAY) if these terms are not reported within the file.

Scenario	Software preference	Encoding
Software scores A and B as same-set, C and D as subset.	Software wishes to make A the group representative (arbitrary)	A = leading protein & group representative B = leading protein C = non-leading protein D = non-leading protein (Use of formal same-set and subset notation is also allowed but optional)
As above	Software does not wish to choose which is the group representative	A = leading protein B = leading protein C = non-leading protein D = non-leading protein
Software scores A as best protein, B, C and D are all subset or subsumed	N/A	A = leading protein B = non-leading protein C = non-leading protein D = non-leading protein
Software scores all four proteins as same-set or more generally as having equal evidence	Software wishes to make A the group representative (arbitrary)	A = leading protein & group representative B = leading protein C = leading protein D = leading protein
As above	Software does not wish to choose which is the group representative	A = leading protein B = leading protein C = leading protein D = leading protein

Scenario	Software preference	Encoding
Software scores A as having slightly more evidence than B. B has additional weak independent evidence relative to A. C and D have less evidence than either A or B.	Software wishes to assign A as the leading protein and the independent evidence for B is not sufficient for it to form a new PAG.	A = leading protein B = non-leading protein & marginally distinguished (optional) C = non-leading protein D = non-leading protein
As above	Software does not wish to choose which is the leading protein out of A and B or group representative	A = leading protein B = leading protein C = non-leading protein D = non-leading protein
As above	Software does not wish to choose which is the leading protein but does select a group representative	A = leading protein & group representative B = leading protein C = non-leading protein D = non-leading protein

Table 4. A summary of grouping options and recommendation for CV term annotations, assuming a group of four related proteins A-D.

Support for *de novo* sequencing approaches

In mzIdentML version 1.1, <SpectrumIdentificationItem> had a mandatory sub-element <PeptideEvidenceRef> to link each PSM to all the proteins from which it could have originated. The inclusion of these mandatory requirements makes it difficult to represent results from *de novo* sequencing and spectral library search approaches where PSMs may not necessarily have originated from a protein database search. As such, in mzIdentML 1.2.0 <PeptideEvidenceRef> has a cardinality of 0...many. In all cases of sequence database search, export software MUST include all <PeptideEvidenceRef> elements for every PSM. In version mzIdentML 1.3.0, further clarifications are added: <SearchDatabase> will include the CV param MS:1000394 ("de novo search or no database used") and "No database" will be included as an userParam in its subelement <DatabaseName>.

De novo sequencing approaches are therefore supported, but only in a relatively straightforward manner, where complete peptide sequences are identified. Proposals for representing partial peptide sequences or sequence tags are encouraged for future iterations of the standard.

Spectral library searches

An alternative to sequence database searches for identifying peptides from MS data is to search a

pre-compiled library (spectral library) of PSMs. These spectral library searches are supported in mzIdentML. The recommended encoding is similar to sequence database search results. The main difference is that a <Peptide> entity SHOULD record each library entry that has been matched against. Additional scores or metadata about the library entity SHOULD be included as <cvParam> elements on <Peptide>. For searches against spectral archives i.e. where the identity of the library entry is unknown (there is no a peptide sequence assignment to the spectrum in the library), the encoding SHOULD include an empty string in <PeptideSequence>.

Note – there has been no formal change to the schema or CV requirements from mzIdentML 1.1 to 1.2.0 around spectral library encoding, but the intended encoding has changed. The mzIdentML 1.1. specifications stated that spectral library entries should be encoded within <DBSequence>, which does not well model the data produced.

Multiple database search engines

Proteomics research groups now commonly analyze MS data using multiple search engines and combine results to improve the number of peptide and protein identifications that can be made. The output of such approaches can be represented in mzIdentML as follows (see Section 6 for documentation of the model elements). Note that the RECOMMENDED encoding has changed since the version 1.1.0 specification as a result of community feedback. It has been decided that throughout mzIdentML, the spectrum referenced from a <SpectrumIdentificationResult> MUST be unique within a file *i.e.* only one set of ranked results can be provided per spectrum. This has implications for encoding the results of multiple search engines, as only consensus results (after they have been combined) can be represented in a valid mzIdentML 1.2.0 file. If exporters wish to maintain the original search engine results, these MAY be encoded using <cvParam> elements within <SpectrumIdentificationItem> containing additional scores, statistics and indicating the original rankings from the source search engine.

The <SpectrumIdentification> element MUST reference a <SpectrumIdentificationProtocol> holding representative parameters used across all search engines (i.e. search tolerances, enzyme and modifications), since these are MANDATORY elements. If the same search parameters were not employed in all source searches, the parameters should be set with superset or widest values i.e. all modifications that have been searched, widest tolerances and so on. All search engines that have been employed SHOULD be represented within the <AnalysisSoftwareList>. The <AnalysisSoftwareList> SHOULD also record the software used to combine results. It must also be highlighted that mzIdentML cannot be used to model the order in which the software was used (it does not support workflows).

The same encoding MAY also be used to describe other approaches where different search protocols are applied to the same spectra (for example using different parameter sets with the same search engine), and subsequently combined. In this case, it is RECOMMENDED that only a single (assumed best) score of any given type is represented once per <SpectrumIdentificationItem>.

Pre-fractionation of samples prior to MS and splitting of searches

It is common in many workflows for pre-fractionation of a sample to be performed prior to MS, for example *via* 1D or 2D gel electrophoresis or 2D LC. In some scenarios results of database searches are combined prior to protein inference and in other instances there is no combination of results prior to protein inference. We have identified the following scenarios and describe the

RECOMMENDED encoding in each case in Table 5 below.

Scenario	Encoding
Scenario 1. i) A sample is fractionated into *n* sub-samples, prior to *n* runs on the MS; ii) the search engine performs *n* searches, producing *n* protein-lists.	*n* mzIdentML files SHOULD be produced, each containing 1 <SpectrumIdentificationList>, 1 <SpectrumIdentificationProtocol>, 1<SpectrumIdentification>, 1 <ProteinDetection>, 1 <ProteinDetectionList>.
Scenario 2. i) A sample is fractionated into *n* sub-samples, prior to *n* runs on the MS; ii) the search engine imports *n* peak lists and performs *n* searches but internally integrates results to produce one protein list.	One single mzIdentML file SHOULD contain *n* <SpectrumIdentificationList>s, *n* <SpectrumIdentificationProtocol>s, *n* <SpectrumIdentification>s, 1 <ProteinDetection>, 1 <ProteinDetectionList>. The <SpectrumIdentificationProtocol>s MUST indicate that pre-fractionation has taken place, using the CV term indicated in Table 1.
Scenario 3. i) A sample is fractionated into *n* sub-samples, prior to *n* runs on the MS; ii) the search engine performs *n* searches, producing *n* lists of spectrum identifications; iii) post-processing software integrates results to produce one protein list.	As Scenario 2.
Scenario 4. i) There is no sample pre-fractionation and one run on the MS. ii) The spectra are split into *n* peak list files for searching (for example for parallelisation on a cluster), producing *n* lists of PSMs iii) post-processing software re-combines results into one mzIdentML file producing 1 protein list.	One single mzIdentML file SHOULD contain 1 <SpectrumIdentificationList>, 1 <SpectrumIdentificationProtocol>, 1 <SpectrumIdentification> referencing *n* <InputSpectra> sub-elements, 1 <ProteinDetection>, 1 <ProteinDetectionList>.

Table 5. A description of RECOMMENDED encodings in mzIdentML, where sample pre-fractionation has taken place.

Encoding replicate samples

One mzIdentML file is intended to capture the analysis of one sample, including rules for pre-fractionation as discussed in Section 5.2.5. For encoding replicate samples (biological or technical), separate mzIdentML files SHOULD be used. A naming convention using suffixes MAY be adopted but the specifications of such are beyond the scope of this document.

Peptide-level scores and statistical measures

The format was designed with explicit support for encoding scores or statistical measures for PSMs, for individual proteins and for protein groups. However, the original design contained no explicit (schema level) support for peptide-level scores i.e. after redundant PSMs reporting on the same peptide have been removed. One of the challenges in this space is defining the mechanism of

grouping multiple PSMs for the same *distinct peptide* – since in different contexts a distinct peptide could encompass one of the following concepts:

- A peptide sequence with a given set of modifications – located in specified positions, identified from a single charge state.

- A peptide sequence with a given set of modifications – located in specified positions, identified from different charge state ions.

- A peptide sequence with a given set of modifications – regardless of the positions of modifications.

- A peptide sequence regardless of the presence/absence of different modifications.

A mechanism for encoding these different types of distinct peptide grouping in the mzIdentML 1.2.0 specifications has been defined, using CV terms as described in Figure 1. Three CV terms have currently been added to the PSI-MS CV: "group PSMs by sequence" (MS:1002496), "group PSMs by sequence with modifications" (MS:1002497) and "group PSMs by sequence with modifications and charge" (MS:1002498).

The following additional features have also been added to mzIdentML 1.2.0 to support peptide-level scores (Figure 1). First, an additional CV term "peptide-level scoring" **(MS:1002490) MUST be included (when this process is being reported)** in <SpectrumIdentificationProtocol>, as shown in Figure 1A and Table 1. In addition, the <SpectrumIdentificationProtocol> contains a <Threshold> element, used in previous versions, for representing the threshold applied at the PSM level. In mzIdentML 1.2, the element can now be used to demonstrate the threshold applied at the PSM and/or peptide-level, through the use of appropriate CV terms.

Additionally, a mechanism is needed for capturing how different PSMs are grouped into a single entity. This is achieved by adding a CV term to every PSM in the file "peptide group ID" (MS:1002520), whereby the associated value is a unique identifier shared between all PSMs in the same peptide group. In the example in Figure 1D, the unique identifier used is the peptide sequence itself (since when grouping by sequence irrespective of modification status this value must be unique), although this could be an arbitrary value such as an integer code.

The mzIdentML file must be able to record scores or statistical values at the peptide level. This is achieved *via* adding CV terms with identical values to all PSMs within the same peptide-group, with an indication that it is a peptide-level value, *via* the convention of the prefix "peptide:" in the CV term name (Figure 1E). Finally, a mechanism has been added for recording peptides both above and below the threshold, to allow complete statistical re-evaluation by downstream software. PSM-level threshold is covered *via* the *passThreshold* attribute on the <SpectrumIdentificationItem> element. To enable additional thresholding at the peptide-level, a new CV term is required for all PSMs ("peptide passes threshold", MS:1002500) as shown in Figure 1F.

[image] | *img/mzidentml/image1.wmf*

Figure 1. The mechanism for encoding peptide-level statistics in mzIdentML 1.2.0.

Encoding modification localisation scores

A new addition to mzIdentML 1.2.0 is the ability to attach scores or statistical values to the position

of a modification, with regards to the peptide sequence. A variety of software packages now export such values, since it is common for there to be more than one possible site of modification. Evidence from the presence or absence of fragment ions can enable a calculation of the likelihood of different possibilities. Such evidence trail is particularly important for some downstream uses of the data, such as profiling motifs for positions of modifications or populating databases with "experimentally observed" modification sites.

The encoding of such scores is achieved in mzIdentML 1.2.0 by making use of a regular expression attached within a <cvParam> at the level of <SpectrumIdentificationItem>. The following additional features to be present in mzIdentML 1.2.0 (Figure 2).

To ensure that downstream software is aware that a file contains modification scores, a CV term is added to the <SpectrumIdentificationProtocol> - "Modification localization scoring" (MS:1002491), as shown in Figure 2A and Table 1. Some approaches apply a statistical threshold for accepting or rejecting that a modification position has been confidently identified. The (re-usable) <Peptide> element has an attribute *via* which the residue and location of a modification can be recorded. To remain backwards compatible, we recommend that the software implementing mod scoring (and export) in mzIdentML should continue to use these attributes, populating with the most likely modification position (Figure 2C). A new CV term (REQUIRED when MS:1002491 is present in the protocol) must be added to every <Modification> element – called "modification index" (MS:1002504), where the value serves as a unique identifier (local only to the containing <Peptide>) to be referenced from the PSM (Figure 2D).

The modification scores themselves are added as CV terms to the <SpectrumIdentificationItem> element referencing the peptide (e.g. "phosphoRS score", Figure 2E), with a value provided as a regular expression of four values in a defined order - **MOD_INDEX, SCORE, POSITION, PASS_THRESHOLD. MOD_INDEX is a reference to the** "modification index" **identifier provided in the referenced <Peptide> - <Modification> element. SCORE represents the score or statistical value (double data type) for the given position. POSITION is the scored modification position with respect to the peptide sequence (where position = 0 is the N-terminus, and the peptide length + 1 is used to indicate the C-terminus). The POSITION can include the bar symbol '|', as a logical OR, if the score relates to multiple positions that can be distinguished. PASS_THRESHOLD holds a Boolean (true, false) value to indicate whether the modification position passes the threshold described above.**

<cvParam accession="MS:1002380" cvRef="PSI-MS" value="1:0.03:2|3:true" name="modification rescored by false localization rate"/>
<cvParam accession="MS:1002380" cvRef="PSI-MS" value="1:0.97:8|9:false" name="modification rescored by false localization rate"/>

The mechanism described MAY be used in conjunction with peptide-level scoring, using specific CV terms for peptide-level modification re-scoring.

[image] | *img/mzidentml/image2.wmf*

Figure 2. The specification in mzIdentML 1.2.0 for encoding modification localization scores, using CV terms.

Encoding results of crosslinking searches

See the crosslinking extension document in the same folder as this document for all the details.

Encoding proteogenomics annotation data

It is now common to use tandem MS data to improve current gene model annotations, in so-called proteogenomics approaches, for example based on making peptide identifications against the official gene models or against alternative databases generated by gene finders, mapping mRNA transcripts or six frame genome translations. Where identifications do not match the official genes, they give evidence in support of updates to the gene models. One of the key concepts required is the mapping of peptides back to chromosomes, including, for example, where they map across splice junctions. File format specifications are under development that can be used directly for genome visualisation, such as adaptations of the BED and BAM (7) formats commonly used in genomics. To ensure a consistent export is possible from mzIdentML to formats designed for genome visualisation or annotation, in mzIdentML 1.2, a consistent encoding of the chromosomal mappings has been developed, as exemplified in Figure 3.

[image] | *img/mzidentml/image3.wmf*

Figure 3. The encoding for chromosomal coordinates in mzIdentML in support of proteogenomics approaches.

Other supporting materials

Example files demonstrating the different uses cases have been developed and are available from the following location: https://github.com/HUPO-PSI/mzIdentML/tree/master/examples/. The sub-folder names indicate the features of the format being used in each example.

Model in XML Schema

An overview of the schema is presented in Figure 4. The following documentation is automatically generated from the XML Schema.
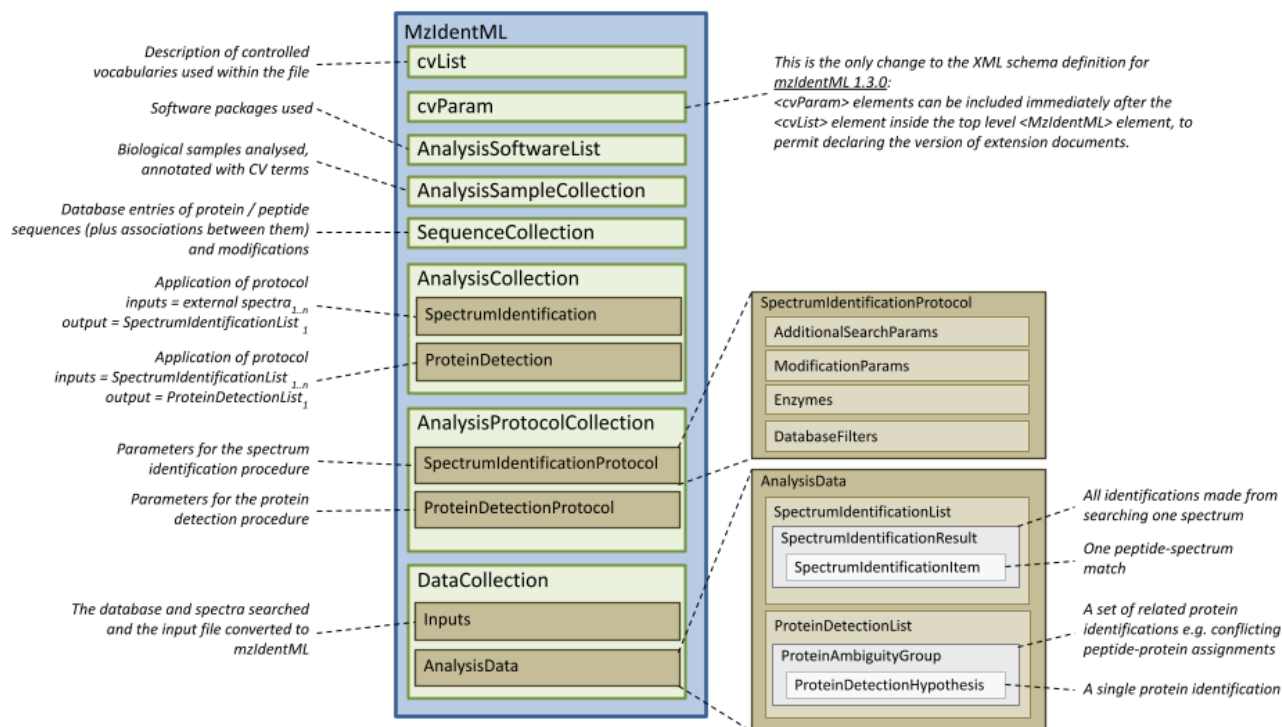
Figure 4. A diagrammatic overview of the mzIdentML schema (generated by xsddiagram).

Element <MzIdentML>

Definition:	The upper-most hierarchy level of mzIdentML with sub-containers for example describing software, protocols and search results (spectrum identifications or protein detection results).
Type:	MzIdentMLType
Attributes:	

|Attribute Name |Data Type |Use |Definition |creationDate |xsd:dateTime |optional |The date on which the file was produced. |id |xsd:string |required |An identifier is an unambiguous string that is unique within the scope (i.e. a document, a set of related documents, or a repository) of its use. |name |xsd:string |optional |The potentially ambiguous common identifier, such as a human-readable name for the instance. |version |versionRegex |required |The version of the schema this instance document refers to, in the format x.y.z. Changes to z should not affect prevent instance documents from validating.

Subelements:	

|Subelement Name |minOccurs |maxOccurs |Definition |cvList |1 |1 |The list of controlled

vocabularies used in the file. |cvParam |0 |unbounded |A single entry from an ontology or a controlled vocabulary. Here, it is used to declare which extension documents are used. |AnalysisSoftwareList |0 |1 |The software packages used to perform the analyses. |Provider |0 |1 |The Provider of the mzIdentML record in terms of the contact and software. |AuditCollection |0 |1 |The complete set of Contacts (people and organisations) for this file. |AnalysisSampleCollection |0 |1 |The samples analysed can optionally be recorded using CV terms for descriptions. If a composite sample has been analysed, the subsample association can be used to build a hierarchical description. |SequenceCollection |0 |1 |The collection of sequences (DBSequence or Peptide) identified and their relationship between each other (PeptideEvidence) to be referenced elsewhere in the results. |AnalysisCollection |1 |1 |The analyses performed to get the results, which map the input and output data sets. Analyses are for example: SpectrumIdentification (resulting in peptides) or ProteinDetection (assemble proteins from peptides). |AnalysisProtocolCollection |1 |1 |The collection of protocols which include the parameters and settings of the performed analyses. |DataCollection |1 |1 |The collection of input and output data sets of the analyses. |BibliographicReference |0 |unbounded |Any bibliographic references associated with the file

Graphical Context:	image::img/mzidentml/image5.png[image,width=282,height=496]

Example Context:	
| | ```
<MzIdentML
xmlns:xsi="http://www.w3.org/2001/XMLSch
ema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSch
ema" id=""
xsi:schemaLocation="http://psidev.info/p
si/pi/mzIdentML/1.2 mzIdentML1.2.0.xsd"
creationDate="2014-06-
03T14:52:58.665376Z" version="1.2.0"
xmlns="http://psidev.info/psi/pi/mzIdent
ML/1.2">

<cvList>

<cv fullName="Proteomics Standards
Initiative Mass Spectrometry
Vocabularies"
version="2.32.0"
uri="https://raw.githubusercontent.com/H
UPO-PSI/psi-ms-CV/master/psi-ms.obo"
id="PSI-MS" />

<cv fullName="UNIMOD"
uri="http://www.unimod.org/obo/unimod.ob
o" id="UNIMOD" />

<cv fullName="UNIT-ONTOLOGY"
uri="http://obo.cvs.sourceforge.net/*che
ckout*/obo/obo/ontology/phenotype/unit.o
bo" id="UO" />

</cvList>

<AnalysisSoftwareList>

...

</MzIdentML>
``` |

Element <AdditionalSearchParams>

Definition:	**The search parameters other than the modifications searched.**
Type:	ParamListType
Attributes:	none
Subelements:	

Subelement Name	minOccurs	maxOccurs	Definition
cvParam	1	unbounded	A single entry from an ontology or a controlled vocabulary.
userParam	1	unbounded	A single user-defined parameter.

Example Context:	<AdditionalSearchParams>
	<cvParam cvRef="PSI-MS" accession="MS:1001211" name="parent mass type mono"/>
	<cvParam cvRef="PSI-MS" accession="MS:1001256" name="fragment mass type mono"/>
	<cvParam cvRef="PSI-MS" accession="MS:1002492" name="consensus scoring"/>
	<cvParam cvRef="PSI-MS" accession="MS:1002490" name="peptide-level scoring"/>
	<cvParam cvRef="PSI-MS" accession="MS:1002497" name="group PSMs by sequence with modifications"/>
	<cvParam cvRef="PSI-MS" accession="MS:1002491" name="modification localization scoring"/>
	... </AdditionalSearchParams>

cvParam Mapping Rules:	Path /MzIdentML/AnalysisProtocolCollection/SpectrumIdentificationProtocol/AdditionalSearchParams

MAY supply a **child** term of MS:1001302 (search engine specific input parameter) one or more times

e.g.: MS:1001005 (SEQUEST:CleavesAt)

e.g.: MS:1001007 (SEQUEST:OutputLines)

e.g.: MS:1001009 (SEQUEST:DescriptionLines)

e.g.: MS:1001026 (SEQUEST:NormalizeXCorrValues)

e.g.: MS:1001028 (SEQUEST:SequenceHeaderFilter)

e.g.: MS:1001032 (SEQUEST:SequencePartialFilter)

e.g.: MS:1001037 (SEQUEST:ShowFragmentIons)

e.g.: MS:1001038 (SEQUEST:Consensus)

e.g.: MS:1001042 (SEQUEST:LimitTo)

e.g.: MS:1001046 (SEQUEST:sort by dCn)

et al.

MAY supply a **child** term of MS:1001066 (ions series considered in search) one or more times

MAY supply a **child** term of MS:1001210 (mass type settings) one or more times

e.g.: MS:1001211 (parent mass type mono)

e.g.: MS:1001212 (parent mass type average)

e.g.: MS:1001255 (fragment mass type average)

e.g.: MS:1001256 (fragment mass type mono)

MAY supply a **child** term of MS:1002489 (special processing) one or more times |

Example cvParams:	`<cvParam cvRef="PSI-MS" accession="MS:1001211" name="parent mass type mono"></cvParam>`
	`<cvParam cvRef="PSI-MS" accession="MS:1001256" name="fragment mass type mono"></cvParam>`
	`<cvParam cvRef="PSI-MS" accession="MS:1002490" name="peptide-level scoring"></cvParam>`
	`<cvParam cvRef="PSI-MS" accession="MS:1002496" name="group PSMs by sequence"></cvParam>`
	`<cvParam accession="MS:1001118" cvRef="PSI-MS" name="param: b ion"/>`
	`<cvParam accession="MS:1001149" cvRef="PSI-MS" name="param: b ion-NH3 DEPRECATED"/>`
	`<cvParam accession="MS:1001150" cvRef="PSI-MS" name="param: b ion-H2O DEPRECATED"/>`
	`<cvParam accession="MS:1001262" cvRef="PSI-MS" name="param: y ion"/>`
	`<cvParam accession="MS:1001151" cvRef="PSI-MS" name="param: y ion-NH3 DEPRECATED"/>`
	`<cvParam accession="MS:1001152" cvRef="PSI-MS" name="param: y ion-H2O DEPRECATED"/>`
	`<cvParam accession="MS:1002494" cvRef="PSI-MS" name="crosslinking search"/>`
	`<cvParam name="no special processing" cvRef="PSI-MS" accession="MS:1002495" />`
	`<cvParam cvRef="PSI-MS" accession="MS:1002492" name="consensus scoring"/>`
	`<cvParam cvRef="PSI-MS" accession="MS:1002497" name="group PSMs by sequence with modifications"/>`
	`<cvParam cvRef="PSI-MS" accession="MS:1002491" name="modification localization scoring"/>`

Example userParams:	`<userParam value="true" name="TargetDecoyApproach"/>`
	`<userParam value="-1" name="MinIsotopeError"/>`
	`<userParam value="2" name="MaxIsotopeError"/>`
	`<userParam value="HCD" name="FragmentMethod"/>`
	`<userParam value="QExactive" name="Instrument"/>`
	`<userParam value="iTRAQ" name="Protocol"/>`
	`<userParam value="2" name="NumTolerableTermini"/>`
	`<userParam value="1" name="NumMatchesPerSpec"/>`
	`<userParam value="2" name="MaxNumModifications"/>`
	`<userParam value="6" name="MinPepLength"/>`
	`<userParam value="40" name="MaxPepLength"/>`
	`<userParam value="2" name="MinCharge"/>`
	`<userParam value="3" name="MaxCharge"/>`
	`<userParam value="2a_uniprot" name="Mascot User Comment"/>`
	`<userParam value="ESI-QUAD" name="Mascot Instrument Name"/>`
	`<userParam name="input_consensusXML" unitName="xsd:string" value="leitner004.consensusXML"/>`
	`<userParam name="input_decoys" unitName="xsd:string" value=""/>`
	`<userParam name="decoy_prefix" unitName="xsd:integer" value="1"/>`

Example for peptide-level statistics:

> <cvParam cvRef="PSI-MS" accession="MS:1001211" name="parent mass type mono"></cvParam>
>
> <cvParam cvRef="PSI-MS" accession="MS:1001256" name="fragment mass type mono"></cvParam>
>
> <cvParam cvRef="PSI-MS" accession="MS:1002490" name="peptide-level scoring"></cvParam>
>
> <cvParam cvRef="PSI-MS" accession="MS:1002496" name="group PSMs by sequence"></cvParam>

Example for sample pre-fractionation:

> <cvParam cvRef="PSI-MS" accession="MS:1001256" name="fragment mass type mono"></cvParam>
>
> <cvParam cvRef="PSI-MS" accession="MS:1001211" name="parent mass type mono"></cvParam>
>
> <cvParam cvRef="PSI-MS" accession="MS:1002493" name="sample pre-fractionation" value="MudPIT"></cvParam>

Example for proteogenomics:

> <cvParam cvRef="PSI-MS" accession="MS:1002635" name="proteogenomics search" value=""></cvParam>

Example for crosslinking:

> <cvParam cvRef="PSI-MS" accession="MS:1001211" name="parent mass type mono"></cvParam>
>
> <cvParam cvRef="PSI-MS" accession="MS:1002494" name="crosslinking search"></cvParam>
>
> <cvParam cvRef="PSI-MS" accession="MS:1001256" name="fragment mass type mono"></cvParam>

Example for modification position scoring:

> <cvParam cvRef="PSI-MS" accession="MS:1002491" name="modification

localization scoring"></cvParam>

Example for de novo sequencing:

<cvParam cvRef="PSI-MS" accession="MS:1001010" name=" de novo search "></cvParam>

Example for consensus scoring:

<cvParam cvRef="PSI-MS" accession="MS:1002492" name="consensus scoring"></cvParam>

<cvParam cvRef="PSI-MS" accession="MS:1002392" name="PIA:PSM sets created" value="true"></cvParam>

<cvParam cvRef="PSI-MS" accession="MS:1002391" name="PIA:Combined FDRScore calculated" value="true"></cvParam>

Element <Affiliation>

Definition:	The organization a person belongs to.
Type:	AffiliationType
Attributes:	

|**Attribute Name** |**Data Type** |**Use** |**Definition** |organization_ref |xsd:string |required |A reference to the organization this contact belongs to.

Subelements:	none
Example Context:	<Affiliation organization_ref="ORG_DOC_OWNER"></Affiliation>

Element <AmbiguousResidue>

Definition:	Ambiguous residues e.g. X can be specified by the Code attribute and a set of parameters for example giving the different masses that will be used in the search.
Type:	AmbiguousResidueType
Attributes:	

|**Attribute Name** |**Data Type** |**Use** |**Definition** |code |chars |required |The single letter code of the ambiguous residue e.g. X.

Subelements:	

| **Subelement Name** | **minOccurs** | **maxOccurs** | **Definition** | cvParam | 1 | unbounded | A single entry from an ontology or a controlled vocabulary. | userParam | 1 | unbounded | A single user-defined parameter.

Example Context:	`<AmbiguousResidue code="X">`
	
	`<cvParam accession="MS:1001360" cvRef="PSI-MS" value="A C D E F G H I K L M N O P Q R S T U V W Y" name="alternate single letter codes"/>`
	
	`</AmbiguousResidue>`

cvParam Mapping Rules:	Path /MzIdentML/AnalysisProtocolCollection/SpectrumIdentificationProtocol/MassTable/AmbiguousResidue MAY supply a **child** term of MS:1001359 (ambiguous residues) one or more times e.g.: MS:1001360 (alternate single letter codes) e.g.: MS:1001361 (alternate mass)
Example cvParams:	`<cvParam accession="MS:1001360" cvRef="PSI-MS" value="D N" name="alternate single letter codes"/>`

Element <AnalysisCollection>

Definition:	**The analyses performed to get the results, which map the input and output data sets. Analyses are for example: SpectrumIdentification (resulting in peptides) or ProteinDetection (assemble proteins from peptides).**
Type:	AnalysisCollectionType
Attributes:	none
Subelements:	

| **Subelement Name** | **minOccurs** | **maxOccurs** | **Definition** | SpectrumIdentification | 1 | unbounded | An Analysis which tries to identify peptides in input spectra, referencing the database searched, the input spectra, the output results and the protocol that is run. | ProteinDetection | 0 | 1 | An Analysis which assembles a set of peptides (e.g. from a spectra search analysis) to proteins.

Example Context:	<AnalysisCollection > <SpectrumIdentification spectrumIdentificationProtocol_ref="SearchProtocol_1_4299" spectrumIdentificationList_ref="SII_LIST_1_1_4299_120114_20_Orbi2_ZC_QC_220_HSAd0-d4-1to1-3_Din.raw" id="SpecIdent__4299_120114_20_Orbi2_ZC_QC_220_HSAd0-d4-1to1-3_Din.raw"> <InputSpectra spectraData_ref="SD_4299_120114_20_Orbi2_ZC_QC_220_HSAd0-d4-1to1-3_Din.raw"></InputSpectra> <SearchDatabaseRef searchDatabase_ref="SDB_4299_203"></SearchDatabaseRef> </SpectrumIdentification> <SpectrumIdentification spectrumIdentificationProtocol_ref="SearchProtocol_1_4299" spectrumIdentificationList_ref="SII_LIST_1_1_4299_120114_09_Orbi2_ZC_QC_220_HSAd0-d4-1to4-2_Din.raw" id="SpecIdent__4299_120114_09_Orbi2_ZC_QC_220_HSAd0-d4-1to4-2_Din.raw"> <InputSpectra spectraData_ref="SD_4299_120114_09_Orbi2_ZC_QC_220_HSAd0-d4-1to4-2_Din.raw"></InputSpectra> ... </AnalysisCollection>

Element <AnalysisData>

Definition:	**Data sets generated by the analyses, including peptide and protein lists.**
Type:	AnalysisDataType
Attributes:	none
Subelements:	

|Subelement Name |minOccurs |maxOccurs |Definition |SpectrumIdentificationList |1

|unbounded |Represents the set of all search results from SpectrumIdentification. |ProteinDetectionList |0 |1 |The protein list resulting from a protein detection process.

| Example Context: | <AnalysisData> <SpectrumIdentificationList xmlns="http://psidev.info/psi/pi/mzIdentML/1.2" id="SII_LIST_1"> <SpectrumIdentificationResult spectrumID="index=6451" spectraData_ref="SD_COMBINED_SE_0" id="SIR_8947"> <SpectrumIdentificationItem chargeState="2" experimentalMassToCharge="679.817322" calculatedMassToCharge="679.818488" peptide_ref="AVMDDFAAFVEK_##Oxidation(M):3" rank="1" passThreshold="false | A_ENSP00000401820 | p:known_378_389"></PeptideEvidenceRef> <PeptideEvidenceRef peptideEvidence_ref="AVMDDFAAFVEK_generic | A_ENSP00000421027 | p:putative_420_431"></PeptideEvidenceRef> <PeptideEvidenceRef peptideEvidence_ref="AVMDDFAAFVEK_generic | A_ENSP00000483421 | p:known_357_368"></PeptideEvidenceRef> ... </AnalysisData> |

Element <AnalysisParams>

Definition:	**The parameters and settings for the protein detection given as CV terms.**
Type:	ParamListType
Attributes:	none
Subelements:	

|**Subelement Name** |**minOccurs** |**maxOccurs** |**Definition** |cvParam |1 |unbounded |A single entry from an ontology or a controlled vocabulary. |userParam |1 |unbounded |A single user-defined parameter.

Example Context:	<AnalysisParams> <cvParam name="mascot:SigThreshold" value="0.05" cvRef="PSI-MS" accession="MS:1001316" /> <cvParam name="mascot:MaxProteinHits" value="Auto" cvRef="PSI-MS" accession="MS:1001317" /> <cvParam name="mascot:ProteinScoringMethod" value="MudPIT" cvRef="PSI-MS" accession="MS:1001318" /> <cvParam name="mascot:MinMSMSThreshold" value="0" cvRef="PSI-MS" accession="MS:1001319" /> <cvParam name="mascot:ShowHomologousProteinsWithSamePeptides" value="1" cvRef="PSI-MS" accession="MS:1001320" /> <cvParam name="mascot:ShowHomologousProteinsWithSubsetOfPeptides" value="10" cvRef="PSI-MS" accession="MS:1001321" /> ... </AnalysisParams>

cvParam Mapping Rules:	Path /MzIdentML/AnalysisProtocolCollection/Protein DetectionProtocol/AnalysisParams
	MAY supply a **child** term of MS:1001302 (search engine specific input parameter) one or more times
	e.g.: MS:1001005 (SEQUEST:CleavesAt
	e.g.: MS:1001007 (SEQUEST:OutputLines)
	e.g.: MS:1001009 (SEQUEST:DescriptionLines)
	e.g.: MS:1001026 (SEQUEST:NormalizeXCorrValues)
	e.g.: MS:1001028 (SEQUEST:SequenceHeaderFilter)
	e.g.: MS:1001032 (SEQUEST:SequencePartialFilter)
	e.g.: MS:1001037 (SEQUEST:ShowFragmentIons)
	e.g.: MS:1001038 (SEQUEST:Consensus)
	e.g.: MS:1001042 (SEQUEST:LimitTo)
	e.g.: MS:1001046 (SEQUEST:sort by dCn)
	et al.
	MAY supply a **child** term of MS:1001194 (quality estimation with decoy database) one or more times

Example cvParams:	`<cvParam name="mascot:SigThreshold" value="0.05" cvRef="PSI-MS" accession="MS:1001316" />`
	
	`<cvParam name="mascot:MaxProteinHits" value="Auto" cvRef="PSI-MS" accession="MS:1001317" />`
	
	`<cvParam name="mascot:ProteinScoringMethod" value="MudPIT" cvRef="PSI-MS" accession="MS:1001318" />`
	
	`<cvParam name="mascot:MinMSMSThreshold" value="0" cvRef="PSI-MS" accession="MS:1001319" />`
	
	`<cvParam name="mascot:ShowHomologousProteinsWithSamePeptides" value="1" cvRef="PSI-MS" accession="MS:1001320" />`
	
	`<cvParam name="mascot:ShowHomologousProteinsWithSubsetOfPeptides" value="10" cvRef="PSI-MS" accession="MS:1001321" />`
	
	`<cvParam name="mascot:RequireBoldRed" value="0" cvRef="PSI-MS" accession="MS:1001322" />`
	
	`<cvParam name="mascot:UseUnigeneClustering" value="false" cvRef="PSI-MS" accession="MS:1001323" />`
	
	`<cvParam name="mascot:IncludeErrorTolerantMatches" value="1" cvRef="PSI-MS" accession="MS:1001324" />`
	
	`<cvParam name="mascot:ShowDecoyMatches" value="0" cvRef="PSI-MS" accession="MS:1001325" />`

Element <AnalysisProtocolCollection>

Definition:	The collection of protocols which include the parameters and settings of the performed analyses.
Type:	AnalysisProtocolCollectionType
Attributes:	none
Subelements:	

|Subelement Name |minOccurs |maxOccurs |Definition |SpectrumIdentificationProtocol |1 |unbounded |The parameters and settings of a SpectrumIdentification analysis. |ProteinDetectionProtocol |0 |1 |The parameters and settings of a ProteinDetection process.

Example Context:	<AnalysisProtocolCollection xmlns="http://psidev.info/psi/pi/mzIdentML/1.2"> <SpectrumIdentificationProtocol analysisSoftware_ref="AS_mascot_server" id="SIP"> <SearchType> <cvParam accession="MS:1001083" cvRef="PSI-MS" value="" name="ms-ms search"/> </SearchType> <AdditionalSearchParams> <cvParam accession="MS:1001211" cvRef="PSI-MS" name="parent mass type mono"/> ... </AnalysisProtocolCollection>

Element <AnalysisSampleCollection>

Definition:	The samples analysed can optionally be recorded using CV terms for descriptions. If a composite sample has been analysed, the subsample association can be used to build a hierarchical description.
Type:	AnalysisSampleCollectionType
Attributes:	none
Subelements:	

|Subelement Name |minOccurs |maxOccurs |Definition |Sample |1 |unbounded |A description of the sample analysed by mass spectrometry using CVParams or UserParams. If a

composite sample has been analysed, a parent sample should be defined, which references subsamples. This represents any kind of substance used in an experimental workflow, such as whole organisms, cells, DNA, solutions, compounds and experimental substances (gels, arrays etc.).

Example Context:	

Element \<AnalysisSoftware\>

Definition:	The software used for performing the analyses.
Type:	AnalysisSoftwareType
Attributes:	

|**Attribute Name** |**Data Type** |**Use** |**Definition** |id |xsd:string |required |An identifier is an unambiguous string that is unique within the scope (i.e. a document, a set of related documents, or a repository) of its use. |name |xsd:string |optional |The potentially ambiguous common identifier, such as a human-readable name for the instance. |uri |xsd:anyURI |optional |URI of the analysis software e.g. manufacturer's website |version |xsd:string |optional |The version of Software used.

Subelements:	

|**Subelement Name** |**minOccurs** |**maxOccurs** |**Definition** |ContactRole |0 |1 |The Contact that provided the document instance. |SoftwareName |1 |1 |The name of the analysis software package, sourced from a CV if available. |Customizations |0 |1 |Any customizations to the software, such as alternative scoring mechanisms implemented, should be documented here as free text.

Example Context:	\<AnalysisSoftware uri="http://code.google.com/p/ehu-bio/downloads/list" version="1.1-beta4" name="PAnalyzer (v1.1-beta4)" id="PAnalyzer"\> \<ContactRole contact_ref="PAnalyzer_Author"\> \<Role\> \<cvParam accession="MS:1001271" cvRef="PSI-MS" name="researcher"/\> \</Role\> \</ContactRole\> \<SoftwareName\> ... \</AnalysisSoftware\>

Element <AnalysisSoftwareList>

Definition:	The software packages used to perform the analyses.
Type:	AnalysisSoftwareListType
Attributes:	none
Subelements:	

|**Subelement Name** |**minOccurs** |**maxOccurs** |**Definition** |AnalysisSoftware |1 |unbounded |The software used for performing the analyses.

Example Context:	<AnalysisSoftwareList xmlns="http://psidev.info/psi/pi/mzIdentML/1.2"> <AnalysisSoftware version="Beta (v9979)" name="MS-GF+" id="ID_software"> <SoftwareName> <cvParam accession="MS:1002048" cvRef="PSI-MS" name="MS-GF+"/> </SoftwareName> </AnalysisSoftware> <AnalysisSoftware name="FalseDiscoveryRate_2014-07-02 12-04-18" id="FalseDiscoveryRate_2014-07-02 12-04-18"> ... </AnalysisSoftwareList>

Element <AuditCollection>

Definition:	The complete set of Contacts (people and organisations) for this file.
Type:	AuditCollectionType
Attributes:	none
Subelements:	

|**Subelement Name** |**minOccurs** |**maxOccurs** |**Definition** |Person |1 |unbounded |A person's name and contact details. Any additional information such as the address, contact email etc. should be supplied using CV parameters or user parameters. |Organization |1 |unbounded |Organizations are entities like companies, universities, government agencies. Any additional information such as the address, email etc. should be supplied either as CV parameters or as user

parameters.

Example Context:	`<AuditCollection xmlns="http://psidev.info/psi/pi/mzIdentML/1.2">`
	
	`<Person lastName="secondName" firstName="firstname" id="PERSON_DOC_OWNER">`
	
	`<cvParam cvRef="PSI-MS" accession="MS:1000587" name="contact address" value="address"></cvParam>`
	
	`<Affiliation organization_ref="ORG_DOC_OWNER"></Affiliation>`
	
	`</Person>`
	
	`<Organization id="ORG_DOC_OWNER" name="myworkplace">`
	
	`<cvParam cvRef="PSI-MS" accession="MS:1000586" name="contact name" value="address"></cvParam>`
	
	`... </AuditCollection>`

Element <BibliographicReference>

Definition:	**Any bibliographic references associated with the file**
Type:	BibliographicReferenceType
Attributes:	

|**Attribute Name** |**Data Type** |**Use** |**Definition** |authors |xsd:string |optional |The names of the authors of the reference. |doi |xsd:string |optional |The DOI of the referenced publication. |editor |xsd:string |optional |The editor(s) of the reference. |id |xsd:string |required |An identifier is an unambiguous string that is unique within the scope (i.e. a document, a set of related documents, or a repository) of its use. |issue |xsd:string |optional |The issue name or number. |name |xsd:string |optional |The potentially ambiguous common identifier, such as a human-readable name for the instance. |pages |xsd:string |optional |The page numbers. |publication |xsd:string |optional |The name of the journal, book etc. |publisher |xsd:string |optional |The publisher of the publication. |title |xsd:string |optional |The title of the BibliographicReference. |volume |xsd:string |optional |The volume name or number. |year |xsd:int |optional |The year of publication.

Subelements:	none

Example Context:	<BibliographicReference id="10.1002/(SICI)1522-2683(19991201)20:18<3551::AID-ELPS3551>3.0.CO;2-2" name="Probability-based protein identification by searching sequence databases using mass spectrometry data" authors="David N. Perkins, Darryl J. C. Pappin, David M. Creasy, John S. Cottrell" publication="Electrophoresis" publisher="Wiley VCH" editor="" year="1999" volume="20" issue="18" pages="3551-3567" title="Probability-based protein identification by searching sequence databases using mass spectrometry data" />

Element <ContactRole>

Definition:	**Depending on context:** **1: The contact details of the organisation or person that produced the software** **2: Contact details for the Material. The association to ContactRole could specify, for example, the creator or provider of the Material.** **3: The Contact that provided the document instance.**
Type:	ContactRoleType
Attributes:	

|**Attribute Name** |**Data Type** |**Use** |**Definition** |contact_ref |xsd:string |required |When a ContactRole is used, it specifies which Contact the role is associated with.

Subelements:	

|**Subelement Name** |**minOccurs** |**maxOccurs** |**Definition** |Role |1 |1 |The roles (lab equipment sales, contractor, etc.) the Contact fills.

Example Context:	<ContactRole contact_ref="PERSON_DOC_OWNER">
	<Role>
	<cvParam cvRef="PSI-MS" accession="MS:1001271" name="researcher"></cvParam>
	</Role>
	</ContactRole>

Element <Customizations>

Definition:	**Any customizations to the software, such as alternative scoring mechanisms implemented, should be documented here as free text.**
Type:	xsd:string
Attributes:	none
Subelements:	none
Example Context:	<Customizations>
	No customisations
	</Customizations>

Element <cv>

Definition:	**A source controlled vocabulary from which cvParams will be obtained.**
Type:	cvType
Attributes:	

|**Attribute Name** |**Data Type** |**Use** |**Definition** |fullName |xsd:string |required |The full name of the CV. |id |xsd:string |required |The unique identifier of this cv within the document to be referenced by cvParam elements. |uri |xsd:anyURI |required |The URI of the source CV. |version |xsd:string |optional |The version of the CV.

Subelements:	none

| Example Context: | ```
<cv fullName="Proteomics Standards
Initiative Mass Spectrometry
Vocabularies"
version="2.32.0"
uri="
https://raw.githubusercontent.com/HUPO-
PSI/psi-ms-CV/master/psi-ms.obo"
id="PSI-MS" />
``` |
|---|---|

Element <cvList>

Definition:	The list of controlled vocabularies used in the file.
Type:	CVListType
Attributes:	none
Subelements:	

|**Subelement Name** |**minOccurs** |**maxOccurs** |**Definition** | cv |1 |unbounded |A source-controlled vocabulary from which cvParams will be obtained.

Example Context:	<cvList>
	<cv id="PSI-MS" fullName="Proteomics Standards Initiative Mass Spectrometry Vocabularies" uri="https://raw.githubusercontent.com/HUPO-PSI/psi-ms-CV/master/psi-ms.obo" version="3.15.0"></cv>
	<cv id="UNIMOD" fullName="UNIMOD" uri="http://www.unimod.org/obo/unimod.obo"></cv>
	<cv id="UO" fullName="UNIT-ONTOLOGY" uri="https://raw.githubusercontent.com/bio-ontology-research-group/unit-ontology/master/unit.obo"></cv>
	<cv id="XLMOD" fullName="PSI cross-link modifications" uri="https://raw.githubusercontent.com/HUPO-PSI/mzIdentML/master/cv/XLMOD-1.0.0.obo"></cv>
	</cvList>

Element <cvParam>

Definition:	A single entry from an ontology or a controlled vocabulary.
Type:	CVParamType
Attributes:	

|**Attribute Name** |**Data Type** |**Use** |**Definition** |accession |xsd:string |required |The accession or ID number of this CV term in the source CV. |cvRef |xsd:string |required |A reference to the cv element from which this term originates. |name |xsd:string |required |The name of the parameter. |unitAccession |xsd:string |optional |An accession number identifying the unit within the OBO foundry Unit CV. |unitCvRef |xsd:string |optional |If a unit term is referenced, this attribute MUST refer to the CV 'id' attribute defined in the cvList in this file. |unitName |xsd:string |optional |The name of the unit. |value |xsd:string |optional |The user-entered value of the parameter.

Subelements:	none
Example Context:	<cvParam cvRef="PSI-MS" accession="MS:1002520" name="peptide group ID" value="CCPQCCSSGCSQNLCGPLCVTTPYYCTR_Carbamidomethyl©:1Carbamidomethyl©:2##Carbamidomethyl(C :5Carbamidomethyl©:6Carbamidomethyl©:10Carbamidomethyl©:15Carbamidomethyl(C :19##Carbamidomethyl©:26"></cvParam>

Element <DatabaseFilters>

Definition:	The specification of filters applied to the database searched.
Type:	DatabaseFiltersType
Attributes:	none
Subelements:	

|**Subelement Name** |**minOccurs** |**maxOccurs** |**Definition** |Filter |1 |unbounded |Filters applied to the search database. The filter MUST include at least one of Include and Exclude. If both are used, it is assumed that inclusion is performed first.

Example Context:	<DatabaseFilters>
	<Filter>
	<FilterType>
	<cvParam accession="MS:1001020" cvRef="PSI-MS" name="DB filter taxonomy"/>
	</FilterType>
	</Filter>
	</DatabaseFilters>

Element <DatabaseName>

Definition:	**The database name may be given as a cvParam if it maps exactly to one of the release databases listed in the CV, otherwise a userParam should be used.**
Type:	ParamType
Attributes:	none
Subelements:	

|**Subelement Name** |**minOccurs** |**maxOccurs** |**Definition** |cvParam |1 |1 |A single entry from an ontology or a controlled vocabulary. |userParam |1 |1 |A single user-defined parameter.

Example Context:	<DatabaseName>
	<userParam name="uniprot-human-reviewed-trypsin-april-2016_concatenated_target_decoy.fasta"/>
	</DatabaseName>

cvParam Mapping Rules:	Path /MzIdentML/DataCollection/Inputs/SearchDatabase/DatabaseName MAY supply a **child** term of MS:1001013 (database name) one or more times e.g.: MS:1001084 (database nr) e.g.: MS:1001104 (database UniProtKB/Swiss-Prot) e.g.: MS:1001142 (database IPI_human) e.g.: MS:1001285 (database IPI_mouse) e.g.: MS:1001286 (database IPI_rat) e.g.: MS:1001287 (database IPI_zebrafish) e.g.: MS:1001288 (database IPI_chicken) e.g.: MS:1001289 (database IPI_cow) e.g.: MS:1001290 (database IPI_arabidopsis) e.g.: MS:1002060 (database UniProtKB/TrEMBL)
Example cvParams:	<cvParam accession="MS:1001104" cvRef="PSI-MS" name="database UniProtKB/Swiss-Prot"/>

Example userParams:	<userParam name="fawaz_PXD000652_combined_concatenated_target_decoy.fasta"></userParam> <userParam name="no description"/> <userParam name="Rosetta_uniprot_20130402_mouse_SWISS_can_iso_ECOLI.fasta"/> <userParam name="Ros_Uniprot_20130402.fasta"/> <userParam name="26Syeast_test.fasta"/> <userParam name="Ros_Uniprot_Ecoli_20130402.fasta" /> <userParam name="uniprot-human-reviewed-trypsin-april-2016_concatenated_target_decoy.fasta"/> <userParam name="HSA-Active.FASTA"></userParam>

Element <DatabaseTranslation>

Definition:	**A specification of how a nucleic acid sequence database was translated for searching.**
Type:	DatabaseTranslationType
Attributes:	

|**Attribute Name** |**Data Type** |**Use** |**Definition** |frames |listOfAllowedFrames |optional |The frames in which the nucleic acid sequence has been translated as a space separated list

Subelements:	

|**Subelement Name** |**minOccurs** |**maxOccurs** |**Definition** |TranslationTable |1 |unbounded |The table used to translate codons into nucleic acids e.g. by reference to the NCBI translation table.

Example Context:	<DatabaseTranslation frames="1 2 3 -1 -2 -3">
	<TranslationTable id="TT_1" name="Standard">
	<cvParam accession="MS:1001025" name="translation table" cvRef="PSI-MS" value="FFLLSSSSYY**CC*WLLLLPPPPHHQQRRR RIIIMTTTTNNKKSSRRVVVVAAAADDEEGGGG" />
	<cvParam accession="MS:1001410" name="translation start codons" cvRef="PSI-MS" value="---M--------------M-------------- M-------------------------" />
	<cvParam accession="MS:1001423" name="translation table description" cvRef="PSI-MS" value="http://www.ncbi.nlm.nih.gov/Taxonomy/ taxonomyhome.html/index.cgi?chapter=cgencod es#SG1" />
	</TranslationTable>
	<TranslationTable id="TT_2" name="Vertebrate Mitochondrial">
	... </DatabaseTranslation>

Element <DataCollection>

Definition:	The collection of input and output data sets of the analyses.
Type:	DataCollectionType
Attributes:	none
Subelements:	

|**Subelement Name** |**minOccurs** |**maxOccurs** |**Definition** |Inputs |1 |1 |The inputs to the analyses including the databases searched, the spectral data and the source file converted to mzIdentML. |AnalysisData |1 |1 |Data sets generated by the analyses, including peptide and protein lists.

Graphical Context:	image::img/mzidentml/image6.png[http://www.p eptideatlas.org/PSI/schemas/mzIdentML/1.2/ figures/ DataCollection.png,width=566,height=642]

Example Context:	<DataCollection>
	<Inputs xmlns="http://psidev.info/psi/pi/mzIdentML/1.2">
	<SearchDatabase numDatabaseSequences="57566" location="E:\Work\PSI\mzIdentML\ProteinInference\Rosetta2\FASTAswithecoli\Rosetta_uniprot_20130402_
	mouse_SWISS_can_iso_ECOLI.fasta" id="SearchDB_1">
	<FileFormat>
	<cvParam accession="MS:1001348" cvRef="PSI-MS" name="FASTA format"/>
	</FileFormat>
	<DatabaseName>
	... </DataCollection>

Element <DBSequence>

Definition:	**A database sequence from the specified SearchDatabase (nucleic acid or amino acid). If the sequence is nucleic acid, the source nucleic acid sequence should be given in the seq attribute rather than a translated sequence.**
Type:	DBSequenceType
Attributes:	

|**Attribute Name** |**Data Type** |**Use** |**Definition** |accession |xsd:string |required |The unique accession of this sequence. |id |xsd:string |required |An identifier is an unambiguous string that is unique within the scope (i.e. a document, a set of related documents, or a repository) of its use. |length |xsd:int |optional |The length of the sequence as a number of bases or residues. |name |xsd:string |optional |The potentially ambiguous common identifier, such as a human-readable name for the instance. |searchDatabase_ref |xsd:string |required |The source database of this sequence.

Subelements:	

|**Subelement Name** |**minOccurs** |**maxOccurs** |**Definition** |Seq |0 |1 |The actual sequence of

amino acids or nucleic acid. |cvParam |0 |unbounded |A single entry from an ontology or a controlled vocabulary. |userParam |0 |unbounded |A single user-defined parameter.

| Example Context: | <DBSequence accession="sp | P20029 | GRP78_MOUSE 78 kDa glucose-regulated protein OS=Mus musculus GN=Hspa5 PE=1..." searchDatabase_ref="SearchDB_1" length="655" name="sp | P20029 | GRP78_MOUSE 78 kDa glucose-regula ted protein OS=Mus musculus GN=Hspa5 PE=1 SV=3" id="dbseq_sp | P20029 | GRP78_MOUSE 78 kDa glucose-regulated protein OS=Mus musculus GN=Hspa5 PE=1..."> <Seq>MMKFTVVAAALLLLGAVRAEEEDKKEDVGTVVGIDLGTTYSCVGVFKNGRVEIIANDQGNRITPSYVAFTPEGERLIGDAAKNQLTSNPENTVFDA KRLIGRTWNDPSVQQDIKFLPFKVVEKKTKPYIQVDIGGGQTKTFAPEEISAMVLTKMKETAEAYLGKKVTHAVVTVPAYFNDAQRQATKDAGTIAGLNVM RIINEPTAAAIAYGLDKREGEKNILVFDLGGGTFDVSLLTIDNGVFEVVATNGDTHLGGEDFDQ |

Example for proteogenomics search:

```
<DBSequence    searchDatabase_ref="SearchDB_1"    accession="generic|A_ENSP00000284981|"
id="dbseq_generic|A_ENSP00000284981|">
```

> <cvParam cvRef="PSI-MS" accession="MS:1002637" name="chromosome name" value="21"></cvParam>
>
> <cvParam cvRef="PSI-MS" accession="MS:1002638" name="chromosome strand" value="-"></cvParam>
>
> <cvParam cvRef="PSI-MS" accession="MS:1002644" name="genome reference version" value="Homo_sapiens.GRCh38.77.gff3"></cvParam>

```
</DBSequence>
```

Element <Enzyme>

Definition:	The details of an individual cleavage enzyme should be provided by giving a regular expression or a CV term if a "standard" enzyme cleavage has been performed.
Type:	EnzymeType
Attributes:	

|**Attribute Name** |**Data Type** |**Use** |**Definition** |cTermGain a| xsd:string with restriction

[A-Za-z0-9]+

|optional |Element formula gained at CTerm. |id |xsd:string |required |An identifier is an unambiguous string that is unique within the scope (i.e. a document, a set of related documents, or a repository) of its use. |minDistance |xsd:int |optional |Minimal distance for another cleavage (minimum: 1). |missedCleavages |xsd:int |optional |The number of missed cleavage sites allowed by the search. The attribute MUST be provided if an enzyme has been used. |nTermGain a| xsd:string with restriction

[A-Za-z0-9]+

|optional |Element formula gained at NTerm. |name |xsd:string |optional |The potentially ambiguous common identifier, such as a human-readable name for the instance. |semiSpecific |xsd:boolean |optional |Set to true if the enzyme cleaves semi-specifically (i.e. one terminus MUST cleave according to the rules, the other can cleave at any residue), false if the enzyme cleavage is assumed to be specific to both termini (accepting for any missed cleavages).

Subelements:	

|**Subelement Name** |**minOccurs** |**maxOccurs** |**Definition** |SiteRegexp |0 |1 |Regular expression

for specifying the enzyme cleavage site. |EnzymeName |0 |1 |The name of the enzyme from a CV.

Example Context:	`<Enzyme missedCleavages="1" semiSpecific="false" cTermGain="OH" nTermGain="H" id="ENZ_0">`
	`<SiteRegexp>(?⇐[KR])</SiteRegexp>`
	`<EnzymeName>`
	`<cvParam accession="MS:1001313" cvRef="PSI-MS" name="Trypsin/P"/>`
	`</EnzymeName>`
	`</Enzyme>`

Element <EnzymeName>

Definition:	**The name of the enzyme from a CV.**
Type:	ParamListType
Attributes:	none
Subelements:	

|**Subelement Name** |**minOccurs** |**maxOccurs** |**Definition** |cvParam |1 |unbounded |A single entry from an ontology or a controlled vocabulary. |userParam |1 |unbounded |A single user-defined parameter.

Example Context:	`<EnzymeName>`
	`<cvParam cvRef="PSI-MS" accession="MS:1001251" name="Trypsin"/>`
	`</EnzymeName>`

cvParam Mapping Rules:	Path /MzIdentML/AnalysisProtocolCollection/SpectrumIdentificationProtocol/Enzymes/Enzyme/EnzymeName
	MAY supply a **child** term of MS:1001045 (cleavage agent name) only once
	e.g.: MS:1001091 (NoEnzyme)
	e.g.: MS:1001251 (Trypsin)
	e.g.: MS:1001303 (Arg-C)
	e.g.: MS:1001304 (Asp-N)
	e.g.: MS:1001305 (Asp-N_ambic)
	e.g.: MS:1001306 (Chymotrypsin)
	e.g.: MS:1001307 (CNBr)
	e.g.: MS:1001308 (Formic_acid)
	e.g.: MS:1001309 (Lys-C)
	e.g.: MS:1001310 (Lys-C/P)
	et al.
Example cvParams:	<cvParam cvRef="PSI-MS" accession="MS:1001251" name="Trypsin"></cvParam>
	<cvParam accession="MS:1001313" cvRef="PSI-MS" name="Trypsin/P"/>

Element <Enzymes>

Definition:	**The list of enzymes used in experiment**
Type:	EnzymesType
Attributes:	

|**Attribute Name** |**Data Type** |**Use** |**Definition** |independent |xsd:boolean |optional |If there are multiple enzymes specified, this attribute is set to true if cleavage with different enzymes is performed independently.

Subelements:	

|**Subelement Name** |**minOccurs** |**maxOccurs** |**Definition** |Enzyme |1 |unbounded |The details

of an individual cleavage enzyme should be provided by giving a regular expression or a CV term if a "standard" enzyme cleavage has been performed.

Example Context:	<Enzymes>
	<Enzyme missedCleavages="1" semiSpecific="false" cTermGain="OH" nTermGain="H" id="ENZ_0">
	<SiteRegexp>(?⇐[KR])</SiteRegexp>
	<EnzymeName>
	<cvParam accession="MS:1001313" cvRef="PSI-MS" name="Trypsin/P"/>
	</EnzymeName>
	</Enzyme>
	... </Enzymes>

Element <Exclude>

Definition:	**All sequences fulfilling the specifed criteria are excluded.**
Attributes:	ParamListType
Attributes:	none
Subelements:	

|**Subelement Name** |**minOccurs** |**maxOccurs** |**Definition** |cvParam |1 |unbounded |A single entry from an ontology or a controlled vocabulary. |userParam |1 |unbounded |A single user-defined parameter.

Example Context:	

cvParam Mapping Rules:	Path /MzIdentML/AnalysisProtocolCollection/SpectrumIdentificationProtocol/DatabaseFilters/Filter/Exclude
	MAY supply a **child** term of MS:1001512 (Sequence database filters) one or more times
	e.g.: MS:1001090 (taxonomy nomenclature)
	e.g.: MS:1001201 (DB MW filter maximum)
	e.g.: MS:1001202 (DB MW filter minimum)
	e.g.: MS:1001203 (DB PI filter maximum)
	e.g.: MS:1001204 (DB PI filter minimum)
	e.g.: MS:1001467 (taxonomy: NCBI TaxID)
	e.g.: MS:1001468 (taxonomy: common name)
	e.g.: MS:1001469 (taxonomy: scientific name)
	e.g.: MS:1001470 (taxonomy: Swiss-Prot ID)
	e.g.: MS:1001513 (DB sequence filter pattern)
	et al.

Element <ExternalFormatDocumentation>

Definition:	**A URI to access documentation and tools to interpret the external format of the ExternalData instance. For example, XML Schema or static libraries (APIs) to access binary formats.**
Type:	xsd:anyURI
Attributes:	none
Subelements:	none
Example Context:	

Element <FileFormat>

Definition:	**The format of the ExternalData file, for example "tiff" for image files.**
Type:	FileFormatType

Definition:	The format of the ExternalData file, for example "tiff" for image files.
Attributes:	none
Subelements:	

|**Subelement Name** |**minOccurs** |**maxOccurs** |**Definition** |cvParam |1 |1 |A single entry from an ontology or a controlled vocabulary.

Example Context:	<FileFormat> <cvParam cvRef="PSI-MS" accession="MS:1001401" name="X!Tandem xml format"/> </FileFormat>

cvParam Mapping Rules:	Path /MzIdentML/DataCollection/Inputs/SearchDatabase/FileFormat
	MUST supply a **child** term of MS:1001347 (database file formats) one or more times
	e.g.: MS:1001348 (FASTA format)
	e.g.: MS:1001349 (ASN.1)
	e.g.: MS:1001350 (NCBI **.p**)
	e.g.: MS:1001351 (clustal aln)
	e.g.: MS:1001352 (embl em)
	e.g.: MS:1001353 (NBRF PIR)
	e.g.: MS:1001462 (PEFF format)
	e.g.: MS:1002659 (UniProtKB text sequence format)
	e.g.: MS:1002660 (UniProtKB XML sequence format)
	Path /MzIdentML/DataCollection/Inputs/SourceFile/FileFormat
	MUST supply a **child** term of MS:1001040 (intermediate analysis format) only once
	e.g.: MS:1000742 (Bioworks SRF format)
	e.g.: MS:1001107 (data stored in database)
	e.g.: MS:1001199 (Mascot DAT format)
	e.g.: MS:1001200 (SEQUEST out file format)
	e.g.: MS:1001242 (SEQUEST out folder)
	e.g.: MS:1001243 (SEQUEST summary)
	e.g.: MS:1001275 (ProteinScape SearchEvent)
	e.g.: MS:1001276 (ProteinScape Gel)
	e.g.: MS:1001399 (OMSSA csv format)

Example cvParams:	`<cvParam cvRef="PSI-MS" accession="MS:1001400" name="OMSSA xml file"></cvParam>`
	`<cvParam cvRef="PSI-MS" accession="MS:1001348" name="FASTA format"></cvParam>`
	`<cvParam cvRef="PSI-MS" accession="MS:1001062" name="Mascot MGF file"></cvParam>`
	`<cvParam accession="MS:1001401" cvRef="PSI-MS" name="X\!Tandem xml file"/>`
	`<cvParam accession="MS:1001199" cvRef="PSI-MS" name="Mascot DAT format"/>`
	`<cvParam accession="MS:1000584" cvRef="PSI-MS" name="mzML format"/>`
	`<cvParam cvRef="PSI-MS" accession="MS:1000563" name="Thermo Raw file"></cvParam>`

Element <Filter>

Definition:	**Filters applied to the search database. The filter MUST include at least one of Include and Exclude. If both are used, it is assumed that inclusion is performed first.**
Type:	FilterType
Attributes:	none
Subelements:	

|**Subelement Name** |**minOccurs** |**maxOccurs** |**Definition** |FilterType |1 |1 |The type of filter e.g. database taxonomy filter, pi filter, mw filter |Include |0 |1 |All sequences fulfilling the specfed criteria are included. |Exclude |0 |1 |All sequences fulfilling the specfed criteria are excluded.

Example Context:	`<Filter>`
	`<FilterType>`
	`<cvParam accession="MS:1001020" cvRef="PSI-MS" name="DB filter taxonomy"/>`
	`</FilterType>`
	`</Filter>`

Element \<FilterType>

Definition:	**The type of filter e.g. database taxonomy filter, pi filter, mw filter**
Type:	ParamType
Attributes:	none
Subelements:	

|**Subelement Name** |**minOccurs** |**maxOccurs** |**Definition** |cvParam |1 |1 |A single entry from an ontology or a controlled vocabulary. |userParam |1 |1 |A single user-defined parameter.

Example Context:	`<FilterType>`
	`<cvParam accession="MS:1001020" cvRef="PSI-MS" name="DB filter taxonomy"/>`
	`</FilterType>`
cvParam Mapping Rules:	Path /MzIdentML/AnalysisProtocolCollection/SpectrumIdentificationProtocol/DatabaseFilters/Filter/FilterType
	MUST supply a **child** term of MS:1001511 (Sequence database filter types) one or more times
	e.g.: MS:1001020 (DB filter taxonomy)
	e.g.: MS:1001021 (DB filter on accession numbers)
	e.g.: MS:1001022 (DB MW filter)
	e.g.: MS:1001023 (DB PI filter)
	e.g.: MS:1001027 (DB filter on sequence pattern)

Example cvParams:	<cvParam accession="MS:1001020" cvRef="PSI-MS" name="DB filter taxonomy"/>

Element \<FragmentArray\>

Definition:	An array of values for a given type of measure and for a particular ion type, in parallel to the index of ions identified.
Type:	FragmentArrayType
Attributes:	

|**Attribute Name** |**Data Type** |**Use** |**Definition** |measure_ref |xsd:string |required |A reference to the Measure defined in the FragmentationTable |values |listOfFloats |required |The values of this particular measure, corresponding to the index defined in ion type

Subelements:	none
Example Context:	\<FragmentArray measure_ref="Measure_Error" values="4.173258879802688E-4 -1.9794682032170385E-5 1.618474794895519E-5 0.001690052197886871 -0.0037214683721344954 0.0020060110579152024 -2.3719321211501665E-4 2.7168621795681247E-4 -0.0019049343519554895 0.0019553613780090018 2.6704080801209784E-4 0.007734020238103767 0.0013568713879976713 1.571508180404635E-4 -0.0017703817320580129 0.013774177127970688 0.0056154565579618065 0.004415735988004599 0.006145015418042021 -0.005059517131940083 0.01419863401793009 0.007626913448120831 0.007892192877989146"/\>

Element \<Fragmentation\>

Definition:	The product ions identified in this result.
Type:	FragmentationType
Attributes:	none
Subelements:	

|**Subelement Name** |**minOccurs** |**maxOccurs** |**Definition** |IonType |1 |unbounded |IonType defines the index of fragmentation ions being reported, importing a CV term for the type of ion e.g. b ion. Example: if b3 b7 b8 and b10 have been identified, the index attribute will contain 3 7 8 10, and the corresponding values will be reported in parallel arrays below

Example Context:	<Fragmentation>
	<IonType charge="1" index="1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23">
	<FragmentArray measure_ref="Measure_MZ" values="175.1193695 232.1403961 289.1618958 452.2268982 509.2429504 566.2701416 653.2999268 710.3218994 767.3411865 854.3770752 911.3968506 968.4257813 1065.472168 1122.492432 1285.553833 1399.612305 1456.62561 1513.645874 1570.669067 1733.721191 1830.793213 1887.808105 1944.829834"/>
	<FragmentArray measure_ref="Measure_Int" values="5939.5844726563 4933.5014648438 13310.7265625 5077.6694335938 5685.9287109375 13253.552734375 7620.0947265625 7724.3696289063 16868.541015625 10552.126953125 11589.0576171875 7839.9741210938 47821.64453125 60335.71484375 21759.3984375 8742.5595703125 11512.0908203125 18130.890625 30577.375 3801.3923339844 8051.07421875 1954.5501708984 4844.9125976563"/>
	<FragmentArray measure_ref="Measure_Error" values="4.173258879802688E-4 -1.9794682032170385E-5 1.618474794895519E-5 0.001690052197886871 -0.0037214683721344954 0.0020060110579152024 -2.3719321211501665E-4 2.7168621795681247E-4 -0.0019049343519554895 0.0019553613780090018 2.6704080801209784E-4 0.007734020238103767 0.0013568713879976713 1.571508180404635E-4 -0.0017703817320580129 0.013774177127970688 0.0056154565579618065 0.004415735988004599 0.006145015418042021 -0.005059517131940083 0.01419863401793009 0.007626913448120831 0.007892192877989146"/>
	<cvParam cvRef="PSI-MS" accession="MS:1001220" name="frag: y ion"/>
	</IonType>

Element <FragmentationTable>

Definition:	Contains the types of measures that will be reported in generic arrays for each SpectrumIdentificationItem e.g. product ion m/z, product ion intensity, product ion m/z error
Type:	FragmentationTableType
Attributes:	none
Subelements:	

|**Subelement Name** |**minOccurs** |**maxOccurs** |**Definition** |Measure |1 |unbounded |References to CV terms defining the measures about product ions to be reported in SpectrumIdentificationItem

Example Context:	<FragmentationTable> <Measure id="Measure_MZ"> <cvParam cvRef="PSI-MS" accession="MS:1001225" name="product ion m/z" unitCvRef="PSI-MS" unitAccession="MS:1000040" unitName="m/z" /> </Measure> <Measure id="Measure_Int"> <cvParam cvRef="PSI-MS" accession="MS:1001226" name="product ion intensity" unitCvRef="PSI-MS" unitAccession="MS:1000131" unitName="number of detector counts"/> </Measure> ... </FragmentationTable>

Element <FragmentTolerance>

Definition:	The tolerance of the search given as a plus and minus value with units.
Type:	ToleranceType
Attributes:	none
Subelements:	

|**Subelement Name** |**minOccurs** |**maxOccurs** |**Definition** |cvParam |1 |unbounded |A single

entry from an ontology or a controlled vocabulary.

Example Context:	`<FragmentTolerance>`
	
	`<cvParam cvRef="PSI-MS" accession="MS:1001412" name="search tolerance plus value" value="20.0 ppm" unitAccession="UO:0000169" unitName="parts per million" unitCvRef="UO"></cvParam>`
	
	`<cvParam cvRef="PSI-MS" accession="MS:1001413" name="search tolerance minus value" value="20.0 ppm" unitAccession="UO:0000169" unitName="parts per million" unitCvRef="UO"></cvParam>`
	
	`</FragmentTolerance>`

cvParam Mapping Rules:	Path /MzIdentML/AnalysisProtocolCollection/SpectrumIdentificationProtocol/FragmentTolerance MUST supply term MS:1001412 (search tolerance plus value) only once MUST supply term MS:1001413 (search tolerance minus value) only once

Example cvParams:	`<cvParam cvRef="PSI-MS" accession="MS:1001412" name="search tolerance plus value" value="0.7" unitAccession="UO:0000221" unitName="dalton" unitCvRef="UO"></cvParam>`
	
	`<cvParam cvRef="PSI-MS" accession="MS:1001413" name="search tolerance minus value" value="0.7" unitAccession="UO:0000221" unitName="dalton" unitCvRef="UO"></cvParam>`

Element <Include>

Definition:	**All sequences fulfilling the specifed criteria are included.**
Type:	ParamListType
Attributes:	none
Subelements:	

|Subelement Name |minOccurs |maxOccurs |Definition |cvParam |1 |unbounded |A single

entry from an ontology or a controlled vocabulary. |userParam |1 |unbounded |A single user-defined parameter.

Example Context:	
cvParam Mapping Rules:	Path /MzIdentML/AnalysisProtocolCollection/SpectrumIdentificationProtocol/DatabaseFilters/Filter/Include

MAY supply a **child** term of MS:1001512 (Sequence database filters) one or more times

e.g.: MS:1001090 (taxonomy nomenclature)

e.g.: MS:1001201 (DB MW filter maximum)

e.g.: MS:1001202 (DB MW filter minimum)

e.g.: MS:1001203 (DB PI filter maximum)

e.g.: MS:1001204 (DB PI filter minimum)

e.g.: MS:1001467 (taxonomy: NCBI TaxID)

e.g.: MS:1001468 (taxonomy: common name)

e.g.: MS:1001469 (taxonomy: scientific name)

e.g.: MS:1001470 (taxonomy: Swiss-Prot ID)

e.g.: MS:1001513 (DB sequence filter pattern)

et al. |

Element <Inputs>

Definition:	**The inputs to the analyses including the databases searched, the spectral data and the source file converted to mzIdentML.**
Type:	InputsType
Attributes:	none
Subelements:	

|**Subelement Name** |**minOccurs** |**maxOccurs** |**Definition** |SourceFile |0 |unbounded |A file from which this mzIdentML instance was created. |SearchDatabase |0 |unbounded |A database for searching mass spectra. Examples include a set of amino acid sequence entries, nucleotide databases (e.g. 6 frame translated) or annotated spectra libraries. |SpectraData |1 |unbounded |A data set containing spectra data (consisting of one or more spectra).

Example Context:	`<Inputs xmlns="http://psidev.info/psi/pi/mzIdentML/1.2">`
	
	`<SourceFile location="C:\Work\PSI\mzIdentML\ProteinInference\Rosetta2\tandem\peaklist2a_plus_ecoli_versus_unimod_`
	
	`full.xml" id="SourceFile_1">`
	
	`<FileFormat>`
	
	`<cvParam accession="MS:1001401" cvRef="PSI-MS" name="X\!Tandem xml file"/>`
	
	`</FileFormat>`
	
	`</SourceFile>`
	
	`<SearchDatabase numDatabaseSequences="163648" location="C:/Work/PSI/mzIdentML/ProteinInference/Rosetta2/FASTAs, neat/Rosetta_uniprot_20130402_mouse_FULL_UNIPROT_can+iso.fasta" id="SearchDB_1">`
	
	`... </Inputs>`

Element <InputSpectra>

Definition:	**One of the spectra data sets used.**
Type:	InputSpectraType
Attributes:	

|**Attribute Name** |**Data Type** |**Use** |**Definition** |spectraData_ref |xsd:string |required |A reference to the SpectraData element which locates the input spectra to an external file.

Subelements:	none
Example Context:	`<InputSpectra spectraData_ref="SD_4299_120114_20_Orbi2_ZC_QC_220_HSAd0-d4-1to1-3_Din.raw"></InputSpectra>`

Element <InputSpectrumIdentifications>

Definition:	The lists of spectrum identifications that are input to the protein detection process.
Type:	InputSpectrumIdentificationsType
Attributes:	

|**Attribute Name** |**Data Type** |**Use** |**Definition** |spectrumIdentificationList_ref |xsd:string |required |A reference to the list of spectrum identifications that were input to the process.

Subelements:	none
Example Context:	<InputSpectrumIdentifications spectrumIdentificationList_ref="SII_LIST_1"/>

Element <IonType>

Definition:	IonType defines the index of fragmentation ions being reported, importing a CV term for the Type of ion e.g. b ion. Example: if b3 b7 b8 and b10 have been identified, the index attribute will contain 3 7 8 10, and the corresponding values will be reported in parallel arrays below
Type:	IonTypeType
Attributes:	

|**Attribute Name** |**Data Type** |**Use** |**Definition** |charge |xsd:int |required |The charge of the identified fragmentation ions. |index |listOfIntegers |optional |The index of ions identified as integers, following standard notation for a-c, x-z e.g. if b3 b5 and b6 have been identified, the index would store "3 5 6". For internal ions, the index contains pairs defining the start and end point - see specification document for examples. For immonium ions, the index is the position of the identified ion within the peptide sequence - if the peptide contains the same amino acid in multiple positions that cannot be distinguished, all positions should be given. For precursor ions, including neutral losses, the index value MUST be 0. For any other ions not related to the position within the peptide sequence e.g. quantification reporter ions, the index value MUST be 0.

Subelements:	

|**Subelement Name** |**minOccurs** |**maxOccurs** |**Definition** |FragmentArray |0 |unbounded |An array of values for a given type of measure and for a particular ion type, in parallel to the index of ions identified. |userParam |0 |unbounded |A single user-defined parameter. |cvParam |1 |unbounded |A single entry from an ontology or a controlled vocabulary.

Example Context:	<IonType charge="1" index="1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23">
	<FragmentArray measure_ref="Measure_MZ" values="175.1193695 232.1403961 289.1618958
	452.2268982 509.2429504 566.2701416 653.2999268 710.3218994 767.3411865 854.3770752 911.3968506 968
	4257813 1065.472168 1122.492432 1285.553833 1399.612305 1456.62561 1513.645874 1570.669067 1733.721191
	1830.793213 1887.808105 1944.829834"/>
	<FragmentArray measure_ref="Measure_Int" values="5939.5844726563 4933.5014648438
	13310.7265625 5077.6694335938 5685.9287109375 13253.552734375 7620.0947265625 7724.3696289063
	16868.541015625 10552.126953125 11589.0576171875 7839.9741210938 47821.64453125 60335.71484375
	21759.3984375 8742.5595703125 11512.0908203125 18130.890625 30577.375 3801.3923339844 8051.07421875
	1954.5501708984 4844.9125976563"/>
	<FragmentArray measure_ref="Measure_Error" values="4.173258879802688E-4
	-1.9794682032170385E-5 1.618474794895519E-5 0.001690052197886871 -0.0037214683721344954
	0.0020060110579152024 -2.3719321211501665E -4 2.7168621795681247E-4 -0.0019049343519554895
	0.0019553613780090018 2.6704080801209784E-4 0.007734020238103767 0.0013568713879976713
	1.571508180404635E-4 -0.0017703817320580129 0.013774177127970688 0.0056154565579618065
	0.004415735988004599 0.006145015418042021 -0.00505951713940083 0.01419863401793009

cvParam Mapping Rules:	Path /MzIdentML/DataCollection/AnalysisData/SpectrumIdentificationList/SpectrumIdentificationResult/
	SpectrumIdentificationItem/Fragmentation/IonType
	MAY supply a **child** term of MS:1001221 (fragmentation information) one or more times
	e.g.: MS:1000903 (product ion series ordinal)
	e.g.: MS:1000904 (product ion m/z delta)
	e.g.: MS:1000926 (product interpretation rank)
	e.g.: MS:1001220 (frag: y ion)
	e.g.: MS:1001222 (frag: b ion - H2O)
	e.g.: MS:1001223 (frag: y ion - H2O)
	e.g.: MS:1001224 (frag: b ion)
	e.g.: MS:1001225 (product ion m/z)
	e.g.: MS:1001227 (product ion m/z error)
	e.g.: MS:1001228 (frag: x ion)
	et al.

Example cvParams:	`<cvParam accession="MS:1001224" cvRef="PSI-MS" name="frag: b ion"/>`
	
	`<cvParam accession="MS:1001220" cvRef="PSI-MS" name="frag: y ion"/>`
	
	`<cvParam accession="MS:1002681" cvRef="PSI-MS" name="OpenXQuest:combined score" value="21.9678562261903"/>`
	
	`<cvParam accession="MS:1002511" cvRef="PSI-MS" name="crosslink spectrum identification item" value="30506749077891582 63"/>`
	
	`<cvParam accession="MS:1000894" cvRef="PSI-MS" name="retention time" value="5468.0193" unitAccession="second" unitName="" unitCvRef="se"/>`
	
	`<cvParam cvRef="PSI-MS" accession="MS:1001523" name="frag: precursor ion"/>`
	
	`<cvParam cvRef="PSI-MS" accession="MS:1002466" name="PeptideShaker PSM score" value="0.0"/>`
	
	`<cvParam cvRef="PSI-MS" accession="MS:1002467" name="PeptideShaker PSM confidence" value="0.0"/>`
	
	`<cvParam cvRef="PSI-MS" accession="MS:1002469" name="PeptideShaker peptide confidence" value="4.0000000000000036"/>`
	
	`<cvParam cvRef="PSI-MS" accession="MS:1002468" name="PeptideShaker peptide score" value="-0.0"/>`
	
	`<cvParam cvRef="PSI-MS" accession="MS:1002500" name="peptide passes threshold" value="false"/>`
	
	`<cvParam cvRef="PSI-MS" accession="MS:1002520" name="peptide group ID" value="QKAQAAATVVK"/>`
	
	`<cvParam cvRef="PSI-MS" accession="MS:1001328" name="OMSSA:evalue"`

Example userParams:	`<userParam name="crosslink_chain" unitName="xsd:string" values="alpha"/>`
	`<userParam name="crosslink_ioncategory" unitName="xsd:string" values="ci"/>`

Element \<MassTable>

Definition:	**The masses of residues used in the search.**
Type:	MassTableType
Attributes:	

|**Attribute Name** |**Data Type** |**Use** |**Definition** |id |xsd:string |required |An identifier is an unambiguous string that is unique within the scope (i.e. a document, a set of related documents, or a repository) of its use. |msLevel |listOfIntegers |required |The MS spectrum that the MassTable refers to e.g. "1" for MS1 "2" for MS2 or "1 2" for MS1 or MS2. |name |xsd:string |optional |The potentially ambiguous common identifier, such as a human-readable name for the instance.

Subelements:	

|**Subelement Name** |**minOccurs** |**maxOccurs** |**Definition** |Residue |0 |unbounded |The specification of a single residue within the mass table. |AmbiguousResidue |0 |unbounded |Ambiguous residues e.g. X can be specified by the Code attribute and a set of parameters for example giving the different masses that will be used in the search. |cvParam |0 |unbounded |A single entry from an ontology or a controlled vocabulary. |userParam |0 |unbounded |A single user-defined parameter.

Example Context:	`<MassTable msLevel="1 2" id="MT">`
	`<Residue mass="71.03712" code="A"/>`
	`<Residue mass="103.009186" code="C"/>`
	`<Residue mass="115.02694" code="D"/>`
	`<Residue mass="129.04259" code="E"/>`
	`<Residue mass="147.06842" code="F"/>`
	`<Residue mass="57.021465" code="G"/>`
	`... </MassTable>`

cvParam Mapping Rules:	Path /MzIdentML/AnalysisProtocolCollection/SpectrumIdentificationProtocol/MassTable
	MAY supply a **child** term of MS:1001354 (mass table options) one or more times
	e.g.: MS:1001346 (AAIndex mass table)

Element <Measure>

Definition:	**References to CV terms defining the measures about product ions to be reported in SpectrumIdentificationItem**
Type:	MeasureType
Attributes:	

|**Attribute Name** |**Data Type** |**Use** |**Definition** |id |xsd:string |required |An identifier is an unambiguous string that is unique within the scope (i.e. a document, a set of related documents, or a repository) of its use. |name |xsd:string |optional |The potentially ambiguous common identifier, such as a human-readable name for the instance.

Subelements:	

|**Subelement Name** |**minOccurs** |**maxOccurs** |**Definition** |cvParam |1 |unbounded |A single entry from an ontology or a controlled vocabulary.

Example Context:	<Measure id="Measure_Int">
	<cvParam cvRef="PSI-MS" accession="MS:1001226" name="product ion intensity" unitCvRef="PSI-MS" unitAccession="MS:1000131" unitName="number of detector counts"/>
	</Measure>

cvParam Mapping Rules:	Path /MzIdentML/DataCollection/AnalysisData/SpectrumIdentificationList/FragmentationTable/Measure
	MUST supply term MS:1001226 (product ion intensity) only once
	MUST supply term MS:1001225 (product ion m/z) only once
	MUST supply term MS:1001227 (product ion m/z error) only once
Example cvParams:	`<cvParam accession="MS:1001225" cvRef="PSI-MS" unitCvRef="PSI-MS" unitName="m/z" unitAccession="MS:1000040" name="product ion m/z"/>`
	`<cvParam accession="MS:1001226" cvRef="PSI-MS" unitCvRef="PSI-MS" unitName="number of detector counts" unitAccession="MS:1000131" name="product ion intensity"/>`
	`<cvParam accession="MS:1001227" cvRef="PSI-MS" unitCvRef="PSI-MS" unitName="m/z" unitAccession="MS:1000040" name="product ion m/z error"/>`

Element <Modification>

Definition:	A molecule modification specification. If n modifications have been found on a peptide, there should be n instances of Modification. If multiple modifications are provided as cvParams, it is assumed that the modification is ambiguous i.e. one modification or another. A cvParam MUST be provided with the identification of the modification sourced from a suitable CV e.g. UNIMOD. If the modification is not present in the CV (and this will be checked by the semantic validator within a given tolerance window), there is a "unknown modification" CV term that MUST be used instead. A neutral loss should be defined as an additional CVParam within Modification. If more complex information should be given about neutral losses (such as presence/absence on particular product ions), this can additionally be encoded within the FragmentationArray. MAY also contain the CV term "search modification id ref" (MS:1003393) once to link the Modification to a SearchModification defined in the ModificationParams of the related SpectrumIdentificationProtocol (Section 7.12). The value of this term is the id of the SearchModification as defined by its "search modification id" (MS:1003392) CV term.
Type:	ModificationType
Attributes:	

|**Attribute Name** |**Data Type** |**Use** |**Definition** |avgMassDelta |xsd:double |optional |Atomic mass delta considering the natural distribution of isotopes in Daltons. |location |xsd:int |optional |Location of the modification within the peptide - position in peptide sequence, counted from the N-terminus residue, starting at position 1. Specific modifications to the N-terminus should be given the location 0. Modification to the C-terminus should be given as peptide length + 1. If the modification location is unknown e.g. for PMF data, this attribute should be omitted. |monoisotopicMassDelta |xsd:double |optional |Atomic mass delta when assuming only the most common isotope of elements in Daltons. |residues |listOfChars |optional |Specification of the residue (amino acid) on which the modification occurs. If multiple values are given, it is assumed that the exact residue modified is unknown i.e. the modification is to ONE of the residues listed. Multiple residues would usually only be specified for PMF data.

Subelements:	

|**Subelement Name** |**minOccurs** |**maxOccurs** |**Definition** |cvParam |1 |unbounded |A single entry from an ontology or a controlled vocabulary.

Example Context:	<Modification location="10" residues="K" monoisotopicMassDelta="138.0680796"> <cvParam accession="XLMOD:02001" cvRef="XLMOD" name="DSS"/> <cvParam accession="MS:1002509" cvRef="PSI-MS" name="crosslink donor" value="11309529182388590588"/> </Modification>
Example Context:	<Modification location="10" residues="K" monoisotopicMassDelta="138.0680796"> <cvParam accession="XLMOD:02001" cvRef="XLMOD" name="DSS"/>

Example cvParams:	`<cvParam cvRef="PSI-MS" accession="MS:1003393" name="search modification id ref" value="DSSO_donor">`
	
	`<cvParam cvRef="UNIMOD" accession="UNIMOD:35" name="Oxidation"></cvParam>`
	
	`<cvParam cvRef="UNIMOD" accession="UNIMOD:37" name="Trimethyl"></cvParam>`
	
	`<cvParam cvRef="UNIMOD" accession="UNIMOD:4" name="Carbamidomethyl"></cvParam>`
	
	`<cvParam cvRef="UNIMOD" accession="UNIMOD:27" name="Glu→pyro-Glu"></cvParam>`
	
	`<cvParam cvRef="UNIMOD" accession="UNIMOD:1" name="Acetyl"></cvParam>`
	
	`<cvParam cvRef="UNIMOD" accession="UNIMOD:385" name="Ammonia-loss"></cvParam>`
	
	`<cvParam cvRef="UNIMOD" accession="UNIMOD:28" name="Gln→pyro-Glu"></cvParam>`
	
	`<cvParam cvRef="UNIMOD" accession="UNIMOD:575" name="Gly→Val"></cvParam>`
	
	`<cvParam accession="UNIMOD:214" cvRef="UNIMOD" name="iTRAQ4plex"/>`
	
	`<cvParam accession="UNIMOD:7" cvRef="UNIMOD" name="Deamidated"/>`
	
	`<cvParam accession="UNIMOD:39" cvRef="UNIMOD" name="Methylthio"/>`
	
	`<cvParam accession="XLMOD:02001" cvRef="XLMOD" name="DSS"/>`
	
	`<cvParam accession="MS:1002509" cvRef="PSI-MS" name="crosslink donor"`

Example for crosslinking:

<Modification location="2" residues="Y" monoisotopicMassDelta="156.07864430999996">

> <cvParam cvRef="XLMOD" accession="XLMOD:01000" name="hydrolysed BS3"></cvParam>

</Modification>

Element <ModificationParams>

Definition:	The specification of static/variable modifications (e.g. Oxidation of Methionine) that are to be considered in the spectra search.
Type:	ModificationParamsType
Attributes:	none
Subelements:	

|**Subelement Name** |**minOccurs** |**maxOccurs** |**Definition** |SearchModification |1 |unbounded |Specification of a search modification as parameter for a spectra search. Contains the name of the modification, the mass, the specificity and whether it is a static modification.

Example Context:	<ModificationParams>
	<SearchModification residues="M" massDelta="15.9949" fixedMod="false">
	<cvParam accession="UNIMOD:35" cvRef="UNIMOD" name="Oxidation"/>
	</SearchModification>
	<SearchModification residues="." massDelta="144.102" fixedMod="false">
	<cvParam accession="UNIMOD:214" cvRef="UNIMOD" name="iTRAQ4plex"/>
	<cvParam accession="MS:1001189" cvRef="PSI-MS" name="modification specificity peptide N-term"/>
	... </ModificationParams>

Element <Organization>

Definition:	Organizations are entities like companies, universities, government agencies. Any additional information such as the address, email etc. should be supplied either as CV parameters or as user parameters.
Type:	OrganizationType
Attributes:	

|**Attribute Name** |**Data Type** |**Use** |**Definition** |id |xsd:string |required |An identifier is an unambiguous string that is unique within the scope (i.e. a document, a set of related documents, or a repository) of its use. |name |xsd:string |optional |The potentially ambiguous common identifier, such as a human-readable name for the instance.

Subelements:	

|**Subelement Name** |**minOccurs** |**maxOccurs** |**Definition** |cvParam |0 |unbounded |A single entry from an ontology or a controlled vocabulary. |userParam |0 |unbounded |A single user-defined parameter. |Parent |0 |1 |The containing organization (the university or business which a lab belongs to, etc.)

Example Context:	<Organization name="PeptideShaker developers" id="PS_DEV"> <cvParam cvRef="PSI-MS" accession="MS:1000586" name="contact name" value="PeptideShaker developers"/> <cvParam cvRef="PSI-MS" accession="MS:1000587" name="contact address" value="Proteomics Unit, Building for Basic Biology, University of Bergen, Jonas Liesvei 91, N-5009 Bergen, Norway"/> <cvParam cvRef="PSI-MS" accession="MS:1000588" name="contact URL" value="http://compomics.github.io/projects/peptide-shaker.html"/> <cvParam cvRef="PSI-MS" accession="MS:1000589" name="contact email" value="peptide-shaker@googlegroups.com"/> </Organization>

cvParam Mapping Rules:	Path /MzIdentML/AuditCollection/Organization
	SHOULD supply term MS:1000588 (contact URL) one or more times
	SHOULD supply term MS:1000587 (contact address) one or more times
	SHOULD supply term MS:1000589 (contact email) one or more times
	SHOULD supply term MS:1000586 (contact name) one or more times
Example cvParams:	<cvParam cvRef="PSI-MS" accession="MS:1000586" name="contact name" value="address"></cvParam>
	<cvParam accession="MS:1000588" cvRef="PSI-MS" value="http://www.matrixscience.com" name="contact URL"/>
	<cvParam cvRef="PSI-MS" accession="MS:1000587" name="contact address" value="test"/>
	<cvParam cvRef="PSI-MS" accession="MS:1000589" name="contact email" value="test"/>

Element <Parent>

Definition:	**The containing organization (the university or business which a lab belongs to, etc.)**
Type:	ParentOrganizationType
Attributes:	

|Attribute Name |Data Type |Use |Definition |organization_ref |xsd:string |required |A reference to the organization this contact belongs to.

Subelements:	none
Example Context:	

Element <ParentTolerance>

Definition:	The tolerance of the search given as a plus and minus value with units.
Type:	ToleranceType
Attributes:	none
Subelements:	

|**Subelement Name** |**minOccurs** |**maxOccurs** |**Definition** |cvParam |1 |unbounded |A single entry from an ontology or a controlled vocabulary.

Example Context:	<ParentTolerance> <cvParam cvRef="PSI-MS" accession="MS:1001412" name="search tolerance plus value" value="6.0 ppm" unitAccession="UO:0000169" unitName="parts per million" unitCvRef="UO"></cvParam> <cvParam cvRef="PSI-MS" accession="MS:1001413" name="search tolerance minus value" value="6.0 ppm" unitAccession="UO:0000169" unitName="parts per million" unitCvRef="UO"></cvParam> </ParentTolerance>
cvParam Mapping Rules:	Path /MzIdentML/AnalysisProtocolCollection/SpectrumIdentificationProtocol/ParentTolerance MUST supply term MS:1001412 (search tolerance plus value) only once MUST supply term MS:1001413 (search tolerance minus value) only once
Example cvParams:	<cvParam cvRef="PSI-MS" accession="MS:1001412" name="search tolerance plus value" value="10.0" unitAccession="UO:0000221" unitName="dalton" unitCvRef="UO"></cvParam> <cvParam cvRef="PSI-MS" accession="MS:1001413" name="search tolerance minus value" value="10.0" unitAccession="UO:0000221" unitName="dalton" unitCvRef="UO"></cvParam>

Element <Peptide>

Definition:	One (poly)peptide (a sequence with modifications). The combination of Peptide sequence and modifications MUST be unique in the file.
Type:	PeptideType
Attributes:	

|**Attribute Name** |**Data Type** |**Use** |**Definition** |id |xsd:string |required |An identifier is an unambiguous string that is unique within the scope (i.e. a document, a set of related documents, or a repository) of its use. |name |xsd:string |optional |The potentially ambiguous common identifier, such as a human-readable name for the instance.

Subelements:	

|**Subelement Name** |**minOccurs** |**maxOccurs** |**Definition** |PeptideSequence |1 |1 |The amino acid sequence of the (poly)peptide. If a substitution modification has been found, the original sequence should be reported. |Modification |0 |unbounded |A molecule modification specification. If n modifications have been found on a peptide, there should be n instances of Modification. If multiple modifications are provided as cvParams, it is assumed that the modification is ambiguous i.e. one modification or another. A cvParam MUST be provided with the identification of the modification sourced from a suitable CV e.g. UNIMOD. If the modification is not present in the CV (and this will be checked by the semantic validator within a given tolerance window), there is a "unknown modification" CV term that MUST be used instead. A neutral loss should be defined as an additional CVParam within Modification. If more complex information should be given about neutral losses (such as presence/absence on particular product ions), this can additionally be encoded within the FragmentationArray. |SubstitutionModification |0 |unbounded |A modification where one residue is substituted by another (amino acid change). |cvParam |0 |unbounded |A single entry from an ontology or a controlled vocabulary. |userParam |0 |unbounded |A single user-defined parameter.

Example Context:	<Peptide id="CCPQCCSSGCSQNLCGPLCVTTPYYCTR_<mark>Carbamidomethyl©:1</mark>Carbamidomethyl©:2##Carbamidomethyl©: 5<mark>Carbamidomethyl©:6</mark>Carbamidomethyl©:10<mark>Carbamidomethyl©:15</mark>Carbamidomethyl©: 19##Carbamidomethyl©:26"> <PeptideSequence>CCPQCCSSGCSQNLCGPLCVTTPYYCTR</PeptideSequence> <Modification location="1" residues="C" monoisotopicMassDelta="57.02147"> <cvParam cvRef="UNIMOD" accession="UNIMOD:4" name="Carbamidomethyl"></cvParam> </Modification> <Modification location="2" residues="C" monoisotopicMassDelta="57.02147"> <cvParam cvRef="UNIMOD" accession="UNIMOD:4" name="Carbamidomethyl"></cvParam> ... </Peptide>
cvParam Mapping Rules:	Path /MzIdentML/SequenceCollection/Peptide MAY supply a **child** term of MS:1001355 (peptide descriptions) one or more times

Example for crosslinking:

```
<Peptide id="54603257_54604608_2_1_p1">

<PeptideSequence>KYLYEIAR</PeptideSequence>

<Modification                    location="2"                    residues="Y"
monoisotopicMassDelta="156.07864430999996">

<cvParam cvRef="XLMOD" accession="XLMOD:01000" name="hydrolysed
BS3"></cvParam>

</Modification>
```

```
<Modification                        location="4"                        residues="Y"
monoisotopicMassDelta="155.094628715">

<cvParam  cvRef="XLMOD"  accession="XLMOD:01001"  name="amidated
BS3"></cvParam>

</Modification>

<Modification location="2" monoisotopicMassDelta="0.0">

<cvParam    cvRef="PSI-MS"    accession="MS:1002510"    name="crosslink
acceptor" value="1"></cvParam>

</Modification>

</Peptide>

<Peptide id="54603257_54604608_2_1_p0">

<PeptideSequence>LSVEAFEK</PeptideSequence>

<Modification location="3" monoisotopicMassDelta="138.06807961">

<cvParam            cvRef="XLMOD"            accession="XLMOD:02000"
name="BS3"></cvParam>

<cvParam cvRef="PSI-MS" accession="MS:1002509" name="crosslink donor"
value="1"></cvParam>

</Modification>

</Peptide>
```

Element <PeptideEvidence>

Definition:	PeptideEvidence links a specific Peptide element to a specific position in a DBSequence. There MUST only be one PeptideEvidence item per Peptide-to-DBSequence-position.
Type:	PeptideEvidenceType
Attributes:	

|**Attribute Name** |**Data Type** |**Use** |**Definition** |dBSequence_ref |xsd:string |required |A reference to the protein sequence in which the specified peptide has been linked. |end |xsd:int

|optional |The index position of the last amino acid of the peptide inside the protein sequence, where the first amino acid of the protein sequence is position 1. Must be provided unless this is a de novo search. |frame |allowed_frames |optional |The translation frame of this sequence if this is PeptideEvidence derived from nucleic acid sequence |id |xsd:string |required |An identifier is an unambiguous string that is unique within the scope (i.e. a document, a set of related documents, or a repository) of its use. |isDecoy |xsd:boolean |optional |Set to true if the peptide is matched to a decoy sequence. |name |xsd:string |optional |The potentially ambiguous common identifier, such as a human-readable name for the instance. |peptide_ref |xsd:string |required |A reference to the identified (poly)peptide sequence in the Peptide element. |post a| xsd:string with restriction

[ABCDEFGHIJKLMNOPQRSTUVWXYZ?\-]{1}

|optional |Post flanking residue. If the peptide is C-terminal, post="-" and not post="". If for any reason it is unknown (e.g. denovo), post="?" should be used. |pre a| xsd:string with restriction

[ABCDEFGHIJKLMNOPQRSTUVWXYZ?\-]{1}

|optional |Previous flanking residue. If the peptide is N-terminal, pre="-" and not pre="". If for any reason it is unknown (e.g. denovo), pre="?" should be used. |start |xsd:int |optional |Start position of the peptide inside the protein sequence, where the first amino acid of the protein sequence is position 1. Must be provided unless this is a de novo search. |translationTable_ref |xsd:string |optional |A reference to the translation table used if this is PeptideEvidence derived from nucleic acid sequence

Subelements:	

|**Subelement Name** |**minOccurs** |**maxOccurs** |**Definition** |cvParam |0 |unbounded |A single entry from an ontology or a controlled vocabulary. |userParam |0 |unbounded |A single user-defined parameter.

Example Context:	<PeptideEvidence dBSequence_ref="dbseq_generic	B_GENSCAN00 000036974_REV ERSED	p:genscan" peptide_ref="CRCQYSGVNNLCHTSSHCPNQGSTCENVDTCLKPDEPK_Carbamidomethyl©:1Carbamidomethyl©: 3Carbamidomethyl©:12Carbamidomethyl©: 18Carbamidomethyl©:25Carbamidomethyl©: 31##Ammonia-loss©:1" start="494" end="531" pre="R" post="L" isDecoy="true" id="CRCQYSGVNNLCHTSSHCPNQGSTCENVDTCLKPDEPK_generic	B_GENSCAN00 000036974_REV ERSED	p:genscan_494_531"> </PeptideEvidence>

Example for proteogenomics search:

<PeptideEvidence dBSequence_ref="dbseq_generic|A_ENSP00000287611|" peptide_ref="YNSQNQSNNQFVLYR_" start="44" end="58" pre="K" post="I" isDecoy="false" id="YNSQNQSNNQFVLYR_generic|A_ENSP00000287611|_44_58">

<cvParam cvRef="PSI-MS" accession="MS:1002640" name="peptide end on chromosome" value="186717716"></cvParam>

<cvParam cvRef="PSI-MS" accession="MS:1002641" name="peptide exon count" value="1"></cvParam>

<cvParam cvRef="PSI-MS" accession="MS:1002642" name="peptide exon nucleotide sizes" value="45"></cvParam>

<cvParam cvRef="PSI-MS" accession="MS:1002643" name="peptide start

positions on chromosome" value="186717673"></cvParam>

</PeptideEvidence>

Element <PeptideEvidenceRef>

Definition:	Reference to the PeptideEvidence element identified. If a specific sequence can be assigned to multiple proteins and or positions in a protein all possible PeptideEvidence elements should be referenced here.
Type:	PeptideEvidenceRefType
Attributes:	

|**Attribute Name** |**Data Type** |**Use** |**Definition** |peptideEvidence_ref |xsd:string |required |A reference to the PeptideEvidenceItem element(s).

Subelements:	none
Example Context:	<PeptideEvidenceRef peptideEvidence_ref="GEGGAQDGSGTEGVGATG AAGGRGAQGAPGGTGGAGSGSGLHHQQDSGYQG ASGSGGAQSGGR_generic
A_ENSP 00000352272_REVERSED	p:known_125_187"></PeptideEvidenceRef>

Example cvParams:	\<cvParam cvRef="PSI-MS" accession="MS:1002356" name="PSM-level combined FDRScore" value="3.9523759266648643E-7">\</cvParam>

\<cvParam cvRef="PSI-MS" accession="MS:1002359" name="distinct peptide-level local FDR" value="0.0">\</cvParam>

\<cvParam cvRef="PSI-MS" accession="MS:1001868" name="distinct peptide-level q-value" value="0.0">\</cvParam>

\<cvParam cvRef="PSI-MS" accession="MS:1002360" name="distinct peptide-level FDRScore" value="3.0117913560694526E-7">\</cvParam>

\<cvParam cvRef="PSI-MS" accession="MS:1002500" name="peptide passes threshold" value="true">\</cvParam>

\<cvParam cvRef="PSI-MS" accession="MS:1002520" name="peptide group ID" value="AVMDDFAAFVEK_##Oxidation(M):3">\</cvParam>

\<cvParam cvRef="PSI-MS" accession="MS:1002439" name="final PSM list UNDER DISCUSSION">\</cvParam>

\<cvParam cvRef="PSI-MS" accession="MS:1002511" name="Crosslinked spectrum identification item." value="21">\</cvParam>

\<cvParam cvRef="PSI-MS" accession="MS:1002545" name="The xi result 'Score'." value="2.769918944845425">\</cvParam>

\<cvParam cvRef="PSI-MS" accession="MS:1000797" name="peak list scans" value="6655">\</cvParam> |
| **Example userParams:** | \<userParam name="search engines identifying PSM" value="12">\</userParam> |

Element \<PeptideHypothesis\>

Definition:	Peptide evidence on which this ProteinHypothesis is based by reference to a PeptideEvidence element.
Type:	PeptideHypothesisType
Attributes:	

|**Attribute Name** |**Data Type** |**Use** |**Definition** |peptideEvidence_ref |xsd:string |required |A reference to the PeptideEvidence element on which this hypothesis is based.

Subelements:	

|**Subelement Name** |**minOccurs** |**maxOccurs** |**Definition** |SpectrumIdentificationItemRef |1 |unbounded |Reference(s) to the SpectrumIdentificationItem element(s) that support the given PeptideEvidence element. Using these references it is possible to indicate which spectra were actually accepted as evidence for this peptide identification in the given protein.

Example Context:	\<PeptideHypothesis peptideEvidence_ref="PepEv_1864"\> \<SpectrumIdentificationItemRef spectrumIdentificationItem_ref="SII_1780_1"/\> \<SpectrumIdentificationItemRef spectrumIdentificationItem_ref="SII_2217_1"/\> \<SpectrumIdentificationItemRef spectrumIdentificationItem_ref="SII_3245_1"/\> \<SpectrumIdentificationItemRef spectrumIdentificationItem_ref="SII_4362_1"/\> \<SpectrumIdentificationItemRef spectrumIdentificationItem_ref="SII_5349_1"/\> \<SpectrumIdentificationItemRef spectrumIdentificationItem_ref="SII_5621_1"/\> ... \</PeptideHypothesis\>

Example cvParams:	`<cvParam accession="MS:1001097" cvRef="PSI-MS" value="1" name="distinct peptide sequences"/>`
	`<cvParam accession="MS:1002235" cvRef="PSI-MS" value="81.01860914459425" name="ProteoGrouper:PDH score"/>`
	`<cvParam accession="MS:1002401" cvRef="PSI-MS" name="leading protein"/>`
	`<cvParam accession="MS:1002403" cvRef="PSI-MS" name="group representative"/>`
	`<cvParam accession="MS:1001594" cvRef="PSI-MS" value="PDH_15" name="sequence same-set protein"/>`
	`<cvParam accession="MS:1002415" cvRef="PSI-MS" value="true" name="protein group passes threshold"/>`
	`<cvParam accession="MS:1002236" cvRef="PSI-MS" value="43.73236628236426" name="ProteoGrouper:PAG score"/>`
	`<cvParam accession="MS:1002407" cvRef="PSI-MS" value="2" name="cluster identifier"/>`
	`<cvParam accession="MS:1002404" cvRef="PSI-MS" value="4" name="count of identified proteins"/>`
	`<cvParam accession="MS:1001596" cvRef="PSI-MS" value="PDH_239" name="sequence sub-set protein"/>`
	`<cvParam accession="MS:1002402" cvRef="PSI-MS" name="non-leading protein"/>`
	`<cvParam accession="MS:1001598" cvRef="PSI-MS" value="PDH_167" name="sequence subsumable protein"/>`
	`<cvParam name="mascot:score" value="1416.6296969697" cvRef="PSI-MS" accession="MS:1001171" />`
	`<cvParam name="PAnalyzer:conclusive protein" cvRef="PSI-MS" accession="MS:1002213" />`

Example userParams:	<userParam value="IINEPTAAAIAYGLDK" name="razor peptides"/>
	<userParam value="SLSDTLEEVLSSSGEK" name="unique peptides"/>

Element <PeptideSequence>

Definition:	**The amino acid sequence of the (poly)peptide. If a substitution modification has been found, the original sequence should be reported.**
Type:	sequence
Attributes:	none
Subelements:	none
Example Context:	<PeptideSequence>GEGGAQDGSGTEGVGATGAAGGRGAQGAPGGTGGAGSGSGLHHQQDSGYQGASGSGGAQSGGR</PeptideSequence>

Element <Person>

Definition:	**A person's name and contact details. Any additional information such as the address, contact email etc. should be supplied using CV parameters or user parameters.**
Type:	PersonType
Attributes:	

|**Attribute Name** |**Data Type** |**Use** |**Definition** |firstName |xsd:string |optional |The Person's first name. |id |xsd:string |required |An identifier is an unambiguous string that is unique within the scope (i.e. a document, a set of related documents, or a repository) of its use. |lastName |xsd:string |optional |The Person's last/family name. |midInitials |xsd:string |optional |The Person's middle initial. |name |xsd:string |optional |The potentially ambiguous common identifier, such as a human-readable name for the instance.

Subelements:	

|**Subelement Name** |**minOccurs** |**maxOccurs** |**Definition** |cvParam |0 |unbounded |A single entry from an ontology or a controlled vocabulary. |userParam |0 |unbounded |A single user-defined parameter. |Affiliation |0 |unbounded |The organization a person belongs to.

Example Context:	`<Person firstName="test" lastName="test" id="PROVIDER">`
	`<cvParam cvRef="PSI-MS" accession="MS:1000587" name="contact address" value="test"/>`
	`<cvParam cvRef="PSI-MS" accession="MS:1000588" name="contact URL" value="test"/>`
	`<cvParam cvRef="PSI-MS" accession="MS:1000589" name="contact email" value="testtest"/>`
	`<Affiliation organization_ref="ORG_DOC_OWNER"/>`
	`</Person>`
cvParam Mapping Rules:	Path /MzIdentML/AuditCollection/Person
	SHOULD supply term MS:1000588 (contact URL) one or more times
	SHOULD supply term MS:1000587 (contact address) one or more times
	SHOULD supply term MS:1000589 (contact email) one or more times
Example cvParams:	`<cvParam cvRef="PSI-MS" accession="MS:1000587" name="contact address" value="address"></cvParam>`
	`<cvParam accession="MS:1000589" cvRef="PSI-MS" value="smartinez@proteored.org" name="contact email"/>`
	`<cvParam cvRef="PSI-MS" accession="MS:1000588" name="contact URL" value="test"/>`

Element <ProteinAmbiguityGroup>

Definition:	**A set of logically related results from a protein detection, for example to represent conflicting assignments of peptides to proteins.**
Type:	ProteinAmbiguityGroupType

Definition:	A set of logically related results from a protein detection, for example to represent conflicting assignments of peptides to proteins.
Attributes:	

|**Attribute Name** |**Data Type** |**Use** |**Definition** |id |xsd:string |required |An identifier is an unambiguous string that is unique within the scope (i.e. a document, a set of related documents, or a repository) of its use. |name |xsd:string |optional |The potentially ambiguous common identifier, such as a human-readable name for the instance.

Subelements:	

|**Subelement Name** |**minOccurs** |**maxOccurs** |**Definition** |ProteinDetectionHypothesis |1 |unbounded |A single result of the ProteinDetection analysis (i.e. a protein). |cvParam |0 |unbounded |A single entry from an ontology or a controlled vocabulary. |userParam |0 |unbounded |A single user-defined parameter.

Example Context:	`<ProteinAmbiguityGroup id="PAG_1">` `<ProteinDetectionHypothesis passThreshold="true" dBSequence_ref="dbseq_sp`	Q64467	G3PT_MOUSE Glyceraldehyde-3-phosphate dehydrogenase, testis-specific OS=Mus..." id="PDH_11"> `<PeptideHypothesis peptideEvidence_ref="PE13_2_63">` `<SpectrumIdentificationItemRef spectrumIdentificationItem_ref="SII_13_1"/>` `</PeptideHypothesis>` `<cvParam accession="MS:1001097" cvRef="PSI-MS" value="1" name="distinct peptide sequences"/>` `<cvParam accession="MS:1002235" cvRef="PSI-MS" value="34.57557513936 462" name="ProteoGrouper: PDH score"/>` ... `</ProteinAmbiguityGroup>`

Example for protein grouping:

`<ProteinAmbiguityGroup id="PAG_4266">`

>   `<ProteinDetectionHypothesis dBSequence_ref="DBSeq_RRRRRQ7TMJ9|Q7TMJ9_MOUSE" passThreshold="true" id="PDH_RRRRRQ7TMJ9|Q7TMJ9_MOUSE_PAG_4266">`
>
>   `<PeptideHypothesis   peptideEvidence_ref="PE_APVPPSQAR(0;144.1021)-332-`

340-RRRRRQ7TMJ9|Q7TMJ9_MOUSE">

<SpectrumIdentificationItemRef
spectrumIdentificationItem_ref="2:[0,144.1021]:APVPPSQAR:index=26699">
</SpectrumIdentificationItemRef>

</PeptideHypothesis>

<cvParam cvRef="PSI-MS" accession="MS:1002394" name="PIA:protein
score" value="2.0881360887005513"></cvParam>

<cvParam cvRef="PSI-MS" accession="MS:1002401" name="leading
protein"></cvParam>

</ProteinDetectionHypothesis>

cvParam cvRef="PSI-MS" accession="MS:1002415" name="protein group
passes threshold" value="true"></cvParam>

cvParam cvRef="PSI-MS" accession="MS:1002407" name="cluster identifier"
value="2814"></cvParam>

</ProteinAmbiguityGroup>

Element <ProteinDetection>

Definition:	An Analysis which assembles a set of peptides (e.g. from a spectra search analysis) to proteins.
Type:	ProteinDetectionType
Attributes:	

|**Attribute Name** |**Data Type** |**Use** |**Definition** |activityDate |xsd:dateTime |optional |When the protocol was applied. |id |xsd:string |required |An identifier is an unambiguous string that is unique within the scope (i.e. a document, a set of related documents, or a repository) of its use. |name |xsd:string |optional |The potentially ambiguous common identifier, such as a human-readable name for the instance. |proteinDetectionList_ref |xsd:string |required |A reference to the ProteinDetectionList in the DataCollection section. |proteinDetectionProtocol_ref |xsd:string |required |A reference to the detection protocol used for this ProteinDetection.

Subelements:	

|**Subelement Name** |**minOccurs** |**maxOccurs** |**Definition** |InputSpectrumIdentifications |1 |unbounded |The lists of spectrum identifications that are input to the protein detection process.

Example Context:	`<ProteinDetection id="PD_1" activityDate="2014-01-11T19:42:49" proteinDetectionList_ref="PDL_PAnalyzer" proteinDetectionProtocol_ref="PDP_PAnalyzer">`
	
	`<InputSpectrumIdentifications spectrumIdentificationList_ref="SIL_1" />`
	
	`</ProteinDetection>`

Element <ProteinDetectionHypothesis>

Definition:	A single result of the ProteinDetection analysis (i.e. a protein).
Type:	ProteinDetectionHypothesisType
Attributes:	

|Attribute Name |Data Type |Use |Definition |dBSequence_ref |xsd:string |required |A reference to the corresponding DBSequence entry. Note - this attribute was optional in mzIdentML 1.1 but is now mandatory in mzIdentML 1.2. Consuming software should assume that the DBSequence entry referenced here is the definitive identifier for the protein. |id |xsd:string |required |An identifier is an unambiguous string that is unique within the scope (i.e. a document, a set of related documents, or a repository) of its use. |name |xsd:string |optional |The potentially ambiguous common identifier, such as a human-readable name for the instance. |passThreshold |xsd:boolean |required |Set to true if the producers of the file has deemed that the ProteinDetectionHypothesis has passed a given threshold or been validated as correct. If no such threshold has been set, value of true should be given for all results.

Subelements:	

|Subelement Name |minOccurs |maxOccurs |Definition |PeptideHypothesis |1 |unbounded |Peptide evidence on which this ProteinHypothesis is based by reference to a PeptideEvidence element. |cvParam |0 |unbounded |A single entry from an ontology or a controlled vocabulary. |userParam |0 |unbounded |A single user-defined parameter.

Example Context:	<ProteinDetectionHypothesis passThreshold="true" dBSequence_ref="dbseq_tr	Q3V2I5	Q3V2I5_MOUSE Glyceraldehyde-3-phosphate dehydrogenase (Fragment) OS=Mus..." id="PDH_10"> <PeptideHypothesis peptideEvidence_ref="PE13_2_62"> <SpectrumIdentificationItemRef spectrumIdentificationItem_ref="SII_13_1"/> </PeptideHypothesis> <cvParam accession="MS:1001097" cvRef="PSI-MS" value="1" name="distinct peptide sequences"/> <cvParam accession="MS:1002235" cvRef="PSI-MS" value="34.575575139364 62" name="ProteoGrouper: PDH score"/> <cvParam accession="MS:1001594" cvRef="PSI-MS" value="PDH_11" name="sequence same-set protein"/> ... </ProteinDetectionHypothesis>

Example for protein grouping:

<cvParam cvRef="PSI-MS" accession="MS:1002394" name="PIA:protein

score" value="107.73038501509386"></cvParam>

<cvParam cvRef="PSI-MS" accession="MS:1002401" name="leading protein"></cvParam>

<cvParam cvRef="PSI-MS" accession="MS:1001594" name="sequence same-set protein" value="PDH_Q3TWF2_PAG_2106 PDH_Q3U7T8_PAG_2106 PDH_Q3U9G2_PAG_2106"></cvParam>

</ProteinDetectionHypothesis>

Element <ProteinDetectionList>

Definition:	The protein list resulting from a protein detection process.
Type:	ProteinDetectionListType
Attributes:	

|Attribute Name |Data Type |Use |Definition |id |xsd:string |required |An identifier is an unambiguous string that is unique within the scope (i.e. a document, a set of related documents, or a repository) of its use. |name |xsd:string |optional |The potentially ambiguous common identifier, such as a human-readable name for the instance.

Subelements:	

|Subelement Name |minOccurs |maxOccurs |Definition |ProteinAmbiguityGroup |0 |unbounded |A set of logically related results from a protein detection, for example to represent conflicting assignments of peptides to proteins. |cvParam |0 |unbounded |A single entry from an ontology or a controlled vocabulary. |userParam |0 |unbounded |A single user-defined parameter.

Example Context:	`<ProteinDetectionList id="PDL_1" xmlns="http://psidev.info/psi/pi/mzIdentML/1.2">` `<ProteinAmbiguityGroup id="PAG_0">` `<ProteinDetectionHypothesis passThreshold="true" dBSequence_ref="dbseq_sp`	P16627	HS71L_MOUSE Heat shock 70 kDa protein 1-like OS=Mus musculus GN=Hspa1l..." id="PDH_15"> `<PeptideHypothesis peptideEvidence_ref="PE16_2_69">` `<SpectrumIdentificationItemRef spectrumIdentificationItem_ref="SII_16_1"/>` `</PeptideHypothesis>` `<PeptideHypothesis peptideEvidence_ref="PE17_2_107">` ... `</ProteinDetectionList>`

Element <ProteinDetectionProtocol>

Definition:	**The parameters and settings of a ProteinDetection process.**
Type:	ProteinDetectionProtocolType
Attributes:	

|**Attribute Name** |**Data Type** |**Use** |**Definition** |analysisSoftware_ref |xsd:string |required |The protein detection software used, given as a reference to the SoftwareCollection section. |id |xsd:string |required |An identifier is an unambiguous string that is unique within the scope (i.e. a document, a set of related documents, or a repository) of its use. |name |xsd:string |optional |The potentially ambiguous common identifier, such as a human-readable name for the instance.

Subelements:	

|**Subelement Name** |**minOccurs** |**maxOccurs** |**Definition** |AnalysisParams |0 |1 |The parameters and settings for the protein detection given as CV terms. |Threshold |1 |1 |The threshold(s) applied to determine that a result is significant. If multiple terms are used it is assumed that all conditions are satisfied by the passing results.

Graphical Context:	image::img/mzidentml/image7.png[http://www.peptideatlas.org/PSI/schemas/mzIdentML/1.2/figures/ProteinDetectionProtocol.png,width=566,height=336]
Example Context:	<ProteinDetectionProtocol id="PDP_PAnalyzer" analysisSoftware_ref="PAnalyzer"> <AnalysisParams> <cvParam name="mascot:SigThreshold" value="0.05" cvRef="PSI-MS" accession="MS:1001316" /> <cvParam name="mascot:MaxProteinHits" value="Auto" cvRef="PSI-MS" accession="MS:1001317" /> <cvParam name="mascot:ProteinScoringMethod" value="MudPIT" cvRef="PSI-MS" accession="MS:1001318" /> <cvParam name="mascot:MinMSMSThreshold" value="0" cvRef="PSI-MS" accession="MS:1001319" /> <cvParam name="mascot:ShowHomologousProteinsWithSamePeptides" value="1" cvRef="PSI-MS" accession="MS:1001320" /> ... </ProteinDetectionProtocol>

Element <Provider>

Definition:	**The Provider of the mzIdentML record in terms of the contact and software.**
Type:	ProviderType
Attributes:	

|**Attribute Name** |**Data Type** |**Use** |**Definition** |analysisSoftware_ref |xsd:string |optional |The Software that produced the document instance. |id |xsd:string |required |An identifier is an unambiguous string that is unique within the scope (i.e. a document, a set of related documents, or a repository) of its use. |name |xsd:string |optional |The potentially ambiguous common identifier, such as a human-readable name for the instance.

Subelements:	

|**Subelement Name** |**minOccurs** |**maxOccurs** |**Definition** |ContactRole |0 |1 |The Contact that provided the document instance.

Example Context:	`<Provider id="PROVIDER" xmlns="http://psidev.info/psi/pi/mzIdentML/1.2" >`
	
	`<ContactRole contact_ref="PERSON_DOC_OWNER">`
	
	`<Role>`
	
	`<cvParam accession="MS:1001271" cvRef="PSI-MS" name="researcher"/>`
	
	`</Role>`
	
	`</ContactRole>`
	
	`</Provider>`

Element <Residue>

Definition:	**The specification of a single residue within the mass table.**
Type:	ResidueType
Attributes:	

|**Attribute Name** |**Data Type** |**Use** |**Definition** |code |chars |required |The single letter code for the residue. |mass |xsd:float |required |The residue mass in Daltons (not including any fixed modifications).

Subelements:	none
Example Context:	`<Residue code="C" mass="103.009186" />`

Element <Role>

Definition:	**The roles (lab equipment sales, contractor, etc.) the Contact fills.**
Type:	RoleType
Attributes:	none
Subelements:	

|**Subelement Name** |**minOccurs** |**maxOccurs** |**Definition** |cvParam |1 |1 |A single entry from an ontology or a controlled vocabulary.

Example Context:	<Role>
	<cvParam accession="MS:1001267" cvRef="PSI-MS" name="software vendor"/>
	</Role>

cvParam Mapping Rules:	Path /MzIdentML/Provider/ContactRole/Role
	MUST supply a **child** term of MS:1001266 (role type) one or more times
	e.g.: MS:1001267 (software vendor)
	e.g.: MS:1001268 (programmer)
	e.g.: MS:1001269 (instrument vendor)
	e.g.: MS:1001270 (lab personnel)
	e.g.: MS:1001271 (researcher)
	Path /MzIdentML/AnalysisSoftwareList/AnalysisSoftware/ContactRole/Role
	MUST supply a **child** term of MS:1001266 (role type) one or more times
	e.g.: MS:1001267 (software vendor)
	e.g.: MS:1001268 (programmer)
	e.g.: MS:1001269 (instrument vendor)
	e.g.: MS:1001270 (lab personnel)
	e.g.: MS:1001271 (researcher)
	Path /MzIdentML/AnalysisSampleCollection/Sample/ContactRole/Role
	MUST supply a **child** term of MS:1001266 (role type) one or more times
	e.g.: MS:1001267 (software vendor)
	e.g.: MS:1001268 (programmer)
	e.g.: MS:1001269 (instrument vendor)
	e.g.: MS:1001270 (lab personnel)
	e.g.: MS:1001271 (researcher)
cvParam Mapping Rules:	

Example cvParams:	`<cvParam cvRef="PSI-MS" accession="MS:1001271" name="researcher"></cvParam>`
	
	`<cvParam accession="MS:1001267" cvRef="PSI-MS" name="software vendor"/>`

Element <Sample>

Definition:	**A description of the sample analysed by mass spectrometry using CVParams or UserParams. If a composite sample has been analysed, a parent sample should be defined, which references subsamples. This represents any kind of substance used in an experimental workflow, such as whole organisms, cells, DNA, solutions, compounds and experimental substances (gels, arrays etc.).**
Type:	SampleType
Attributes:	

|**Attribute Name** |**Data Type** |**Use** |**Definition** |id |xsd:string |required |An identifier is an unambiguous string that is unique within the scope (i.e. a document, a set of related documents, or a repository) of its use. |name |xsd:string |optional |The potentially ambiguous common identifier, such as a human-readable name for the instance.

Subelements:	

|**Subelement Name** |**minOccurs** |**maxOccurs** |**Definition** |ContactRole |0 |unbounded |The Contact that provided the document instance. |SubSample |0 |unbounded |References to the individual component samples within a mixed parent sample. |cvParam |0 |unbounded |A single entry from an ontology or a controlled vocabulary. |userParam |0 |unbounded |A single user-defined parameter.

Example Context:	

Element <SearchDatabase>

Definition:	**A database for searching mass spectra. Examples include a set of amino acid sequence entries, nucleotide databases (e.g. 6 frame translated) or annotated spectra libraries.**
Type:	SearchDatabaseType

Definition:	A database for searching mass spectra. Examples include a set of amino acid sequence entries, nucleotide databases (e.g. 6 frame translated) or annotated spectra libraries.
Attributes:	

|**Attribute Name** |**Data Type** |**Use** |**Definition** |id |xsd:string |required |An identifier is an unambiguous string that is unique within the scope (i.e. a document, a set of related documents, or a repository) of its use. |location |xsd:anyURI |required |The location of the data file. |name |xsd:string |optional |The potentially ambiguous common identifier, such as a human-readable name for the instance. |numDatabaseSequences |xsd:long |optional |The total number of sequences in the database. |numResidues |xsd:long |optional |The number of residues in the database. |releaseDate |xsd:dateTime |optional |The date and time the database was released to the public; omit this attribute when the date and time are unknown or not applicable (e.g. custom databases). |version |xsd:string |optional |The version of the database.

Subelements:	

|**Subelement Name** |**minOccurs** |**maxOccurs** |**Definition** |ExternalFormatDocumentation |0 |1 |A URI to access documentation and tools to interpret the external format of the ExternalData instance. For example, XML Schema or static libraries (APIs) to access binary formats. |FileFormat |1 |1 |The format of the ExternalData file, for example "tiff" for image files. |DatabaseName |1 |1 |The database name may be given as a cvParam if it maps exactly to one of the release databases listed in the CV, otherwise a userParam should be used. |cvParam |0 |unbounded |A single entry from an ontology or a controlled vocabulary.

Example Context:	`<SearchDatabase numDatabaseSequences="40400" location="C:\Users\hba041\My_Git_Applications\ peptide-shaker.wiki\data\2016_04_05\uniprot-human-reviewed-trypsin-april-2016_concatenated_target_decoy.fasta" id="SearchDB_1">`
	`<FileFormat>`
	`<cvParam                    cvRef="PSI-MS" accession="MS:1001348"            name="FASTA format"/>`
	`</FileFormat>`
	`<DatabaseName>`
	`<userParam    name="uniprot-human-reviewed-trypsin-april-2016_concatenated_target_decoy.fasta"/>`
	`</DatabaseName>`
	`... </SearchDatabase>`

cvParam Mapping Rules:	Path /MzIdentML/DataCollection/Inputs/SearchDatabase
	MAY supply a **child** term of MS:1000561 (data file checksum type) one or more times
	e.g.: MS:1000568 (MD5)
	e.g.: MS:1000569 (SHA-1)
	MAY supply a **child** term of MS:1001011 (search database details) one or more times
	e.g.: MS:1001014 (database local file path)
	e.g.: MS:1001015 (database original uri)
	e.g.: MS:1001016 (database version)
	e.g.: MS:1001017 (database release date)
	e.g.: MS:1001020 (DB filter taxonomy)
	e.g.: MS:1001021 (DB filter on accession numbers)
	e.g.: MS:1001022 (DB MW filter)
	e.g.: MS:1001023 (DB PI filter)
	e.g.: MS:1001024 (translation frame)
	e.g.: MS:1001025 (translation table)
	et al.

Element <SearchDatabaseRef>

Definition:	**One of the search databases used.**
Type:	SearchDatabaseRefType
Attributes:	

|**Attribute Name** |**Data Type** |**Use** |**Definition** |searchDatabase_ref |xsd:string |required |A reference to the database searched.

Subelements:	none

Example Context:	`<SearchDatabaseRef searchDatabase_ref="SDB_4299_203"></SearchDatabaseRef>`

Element \<SearchModification\>

Definition:	**Specification of a search modification as parameter for a spectra search. Contains the name of the modification, the mass, the specificity and whether it is a static modification. It MAY provide the CV term "search modification id" (MS:1003392) once, to provide an identifier for this SearchModification (Section 7.12). This identifier MUST be unique within the \<SpectrumIdentificationProtocol\> element. If there are multiple \<SpectrumIdentificationProtocol\> elements with in the file, then the values of "search modification id" SHOULD be identical for identical modifications.**
Type:	SearchModificationType
Attributes:	

|**Attribute Name** |**Data Type** |**Use** |**Definition** |fixedMod |xsd:boolean |required |True, if the modification is static (i.e. occurs always). |massDelta |xsd:float |required |The mass delta of the searched modification in Daltons. |residues |listOfCharsOrAny |required |The residue(s) searched with the specified modification. For N or C terminal modifications that can occur on any residue, the . character should be used to specify any, otherwise the list of amino acids should be provided.

Subelements:	

|**Subelement Name** |**minOccurs** |**maxOccurs** |**Definition** |SpecificityRules |0 |unbounded |The specificity rules of the searched modification including for example the probability of a modification's presence or peptide or protein termini. Standard fixed or variable status should be provided by the attribute fixedMod. |cvParam |1 |unbounded |A single entry from an ontology or a controlled vocabulary.

Example Context:	\<SearchModification residues="E" massDelta="-18.010565" fixedMod= "false" \>
	\<SpecificityRules\>
	\<cvParam cvRef="PSI-MS" accession="MS:1001189" name="modification specificity peptide N-term"/\>
	\</SpecificityRules\>
	\<cvParam cvRef="UNIMOD" accession="UNIMOD:27" name="Glu→pyro-Glu"/\>
	\<cvParam cvRef="PSI-MS" accession="MS:1002504" name="modification index" value="3"/\>
	\</SearchModification\>

cvParam Mapping Rules:	Path /MzIdentML/AnalysisProtocolCollection/SpectrumIdentificationProtocol/ModificationParams/ SearchModification
	MAY supply term MS:1003392 (search modification id) only once
	MAY supply term MS:1002509 (crosslink donor) only once
	MAY supply term MS:1002510 (crosslink acceptor) only once
	MAY supply a **child** term of UNIMOD:0 (unimod root node) only once
	MAY supply a **child** term of MS:1001471 (peptide modification details) one or more times
	e.g.: MS:1001460 (unknown modification)
	e.g.: MS:1001524 (fragment neutral loss)
	e.g.: MS:1001525 (precursor neutral loss)
	e.g.: MS:1001972 (PTM scoring algorithm version)
	e.g.: MS:1002028 (nucleic acid base modification)
	e.g.: MS:1002029 (original nucleic acid sequence)
	e.g.: MS:1002030 (modified nucleic acid sequence)
	MUST supply term MS:1001460 (unknown modification) only once
	MAY supply a **child** term of XLMOD:00002 (cross-linker related PTM) only once
	MAY supply term MS:1002504 (modification index) only once
	MAY supply a **child** term of XLMOD:00004 (cross-linker) only once
	MAY supply a **child** term of MOD:00000 (protein

Example cvParams:	`<cvParam cvRef="UNIMOD" accession="UNIMOD:35" name="Oxidation"></cvParam>`
	`<cvParam cvRef="UNIMOD" accession="UNIMOD:4" name="Carbamidomethyl"></cvParam>`
	`<cvParam accession="UNIMOD:214" cvRef="UNIMOD" name="iTRAQ4plex"/>`
	`<cvParam accession="MS:1001189" cvRef="PSI-MS" name="modification specificity peptide N-term"/>`
	`<cvParam accession="UNIMOD:39" cvRef="UNIMOD" name="Methylthio"/>`
	`<cvParam accession="UNIMOD:7" cvRef="UNIMOD" name="Deamidated"/>`
	`<cvParam cvRef="PSI-MS" accession="MS:1002504" name="modification index" value="0"/>`
	`<cvParam cvRef="XLMOD" accession="XLMOD:02000" name="BS3"></cvParam>`
	`<cvParam cvRef="PSI-MS" accession="MS:1002509" name="crosslink donor" value="0"></cvParam>`
	`<cvParam cvRef="PSI-MS" accession="MS:1002510" name="crosslink acceptor" value="0"></cvParam>`
	`<cvParam cvRef="XLMOD" accession="XLMOD:02004" name="BS3-d4"></cvParam>`
	`<cvParam cvRef="XLMOD" accession="XLMOD:01001" name="amidated BS3"></cvParam>`
	`<cvParam cvRef="XLMOD" accession="XLMOD:01000" name="hydrolyzed BS3"></cvParam>`
	`<cvParam cvRef="UNIMOD"`

Element \<SearchType\>

Definition:	The type of search performed e.g. PMF, Tag searches, MS-MS
Type:	ParamType
Attributes:	none
Subelements:	

|**Subelement Name** |**minOccurs** |**maxOccurs** |**Definition** |cvParam |1 |1 |A single entry from an ontology or a controlled vocabulary. |userParam |1 |1 |A single user-defined parameter.

Example Context:	\<SearchType\> \<cvParam accession="MS:1001083" cvRef="PSI-MS" value="" name="ms-ms search"/\> \</SearchType\>
cvParam Mapping Rules:	Path /MzIdentML/AnalysisProtocolCollection/SpectrumIdentificationProtocol/SearchType MUST supply a **child** term of MS:1001080 (search type) one or more times e.g.: MS:1001010 (de novo search) e.g.: MS:1001031 (spectral library search) e.g.: MS:1001081 (pmf search) e.g.: MS:1001082 (tag search) e.g.: MS:1001083 (ms-ms search) e.g.: MS:1001584 (combined pmf + ms-ms search) e.g.: MS:1002490 (peptide-level scoring) e.g.: MS:1002491 (modification localization scoring) e.g.: MS:1002492 (consensus scoring) e.g.: MS:1002493 (sample pre-fractionation) et al.

Example cvParams:	<cvParam cvRef="PSI-MS" accession="MS:1001083" name="ms-ms search"></cvParam>

Element <Seq>

Definition:	**The actual sequence of amino acids or nucleic acid.**
Type:	sequence
Attributes:	none
Subelements:	none
Example Context:	<Seq>MMKFTVVAAALLLLGAVRAEEEDKKEDVG TVVGIDLGTTYSCVGVFKNGRVEIIANDQGNRITPS YVAFTPEGERLIGDAAKNQLTSNPENTVFDA KRLIGRTWNDPSVQQDIKFLPFKVVEKKTKPYIQV DIGGGQTKTFAPEEISAMVLTKMKETAEAYLGKKV THAVVTVPAYFNDAQRQATKDAGTIAGLNVM RIINEPTAAAIAYGLDKREGEKNILVFDLGGGTFDV SLLTIDNGVFEVVATNGDTHLGGEDFDQRVMEHF IKLYKKKTGKDVRKDNRAVQKLRREVEKAKR ALSSQHQARIEIESFFEREDFSETLTRAKFEELNMD LFRSTMKPVQKVLEDSDLKKSDIDEIVLVGGSTRIP KIQQLVKEFFNGKEPSRGINPDEAVAYGA AVQAGVLSGGQDTGDLVLLDVCPLTLGIETVGGV MTKLIPRNTVVPTKKSQIFSTASDNQPTVTIKVYEG ERPLTKDNHLLGTFDLTGIPPAPRGVPQIEV TFEIDVNGILRVTAEDKGTGNKNKITITNDQNRLTP EEIERMVNDAEKFAEEDKKLKERIDTRNELESYAY SLKNQIGDKEKLGGKLSSEDKETMEKAVEE KIEWLESHQDADIEDFKAKKKELEEIVQPIISKLYG SGGPPPTGEEDTSEKDEL</Seq>

Element <SequenceCollection>

Definition:	**The collection of sequences (DBSequence or Peptide) identified and their relationship between each other (PeptideEvidence) to be referenced elsewhere in the results.**
Type:	SequenceCollectionType

Definition:	The collection of sequences (DBSequence or Peptide) identified and their relationship between each other (PeptideEvidence) to be referenced elsewhere in the results.
Attributes:	none
Subelements:	

|**Subelement Name** |**minOccurs** |**maxOccurs** |**Definition** |DBSequence |0 |unbounded |A database sequence from the specified SearchDatabase (nucleic acid or amino acid). If the sequence is nucleic acid, the source nucleic acid sequence should be given in the seq attribute rather than a translated sequence. |Peptide |0 |unbounded |One (poly)peptide (a sequence with modifications). The combination of Peptide sequence and modifications MUST be unique in the file. |PeptideEvidence |0 |unbounded |PeptideEvidence links a specific Peptide element to a specific position in a DBSequence. There MUST only be one PeptideEvidence item per Peptide-to-DBSequence-position.

Graphical Context:	image::img/mzidentml/image8.png[http://www.peptideatlas.org/PSI/schemas/mzIdentML/1.2/figures/SequenceCollection.png,width=576,height=652]
Example Context:	<SequenceCollection xmlns="http://psidev.info/psi/pi/mzIdentML/1.2"> <DBSequence accession="sp
Q64467	G3PT_MOUSE Glyceraldehyde-3-phosphate dehydrogenase, testis-speci fic OS=Mus..." searchDatabase_ref="SearchDB_1" length="440" name="sp
Q64467	G3PT_MOUSE Glyceraldehyde-3 -phosphate dehydrogenase, testis-specific OS=Mus musculus GN=Gapdhs PE=1 SV=1" id="dbseq_sp

Q64467	G3PT _MOUSE Glyceraldehyde-3-phosphate dehydrogenase, testis-specific OS=Mus..."> <Seq>MSRRDVVLTNVTVVQLRRDRCPCPCPCPC PCPVIRPPPPKLEDPPPTVEEQPPPPPPPPPPPPPPP PPPPPQIEPDKFEEAPPPPP PPPPPPPPPP PPLQKPARELTVGINGFGRIGRLVLRVCMEKGIRVV AVNDPFIDPEYMVYMFKYDSTHGRYKGNVEHKNG QLVVDNLEINTYQCKDPKEIPWSSIGNPYVVEC TGVYLSIEAASAHISSGARRVVVTAPSPDAPMFVM GVNEKDYNPGSMTIVSNASCTTNCLAPLAKVIHEN FGIVEGLMTTVHSYTATQKTVDGPSKKDWRGGR GAHQNIIPSSTGAAKAVGKVIPELKGKLTGMAFRV PTPNVSVVDLTCRLAKPASYSAITEAVKAAAKGPLA GILAYTEDQVVSTDFNGNPHSSIFDAKAGIAL NDNFVKLVAWYDNEYGYSNRVVDLLRYMFSREK< /Seq> </DBSequence> <DBSequence accession="tr
Q3UEM8	Q3UEM8_MOUSE Putative uncharacterized protein (Fragment) OS=Mus musculus..." searchDatabase_ref="SearchDB_1" length="520" name="tr
Q3UEM8	Q3UEM8_MOUSE Putative unchar acterized protein (Fragment) OS=Mus musculus GN=Hspa5 PE=2 SV=1" id="dbseq_tr

Q3UEM8	Q3UEM8_MOUSE Puta
	tive uncharacterized protein (Fragment) OS=Mus musculus...">
	<Seq>MMKFTVVAAALLLLGAVRAEEEDKKEDVG TVVGIDLGTTYSCVGVFKNGRVEIIANDQGNRITPS YVAFTPEGERLIGDAAKNQLTSNP
	ENTVFDAK
	RLIGRTWNDPSVQQDIKFLPFKVVEKKTKPYIQVDI GGGQTKTFAPEEISAMVLTKMKETAEAYLGKKVT HAVVTVPAYFNDAQRQATKDAGTIAGLNVMRI
	INEPTAAAIAYGLDKREGEKNILVFDLGGGTFDVSL LTIDNGVFEVVATNGDTHLGGEDFDQRVMEHFIK LYKKKTGKDVRKDNRAVQKLRREVEKAKRALS
	SQHQARIEIESFFEGEDFSETLTRAKFEELNMDLFR STIKPVQKVLEDSDLKKSDIDEIVLVGGSTRIPKIQQ LVKEFFNGKEPSRGINPDEAVAYGAAVQA
	GVLSGDQDTGDLVLLDVCPLTLGIETVGGVMTKLI PRNTVVPTKKSQIFSTASDNQPTVTIKVYEGERPLT KDNHLLGTFDLTGIPPAPRGVPQIEVTFEID
	VNGILRVTAEDKGTG</Seq>
	</DBSequence>
	... </SequenceCollection>

Element <SiteRegexp>

Definition:	**Regular expression for specifying the enzyme cleavage site.**
Type:	xsd:string
Attributes:	none
Subelements:	none
Example Context:	<SiteRegexp>(?⇐[KR])</SiteRegexp>

Element <SoftwareName>

Definition:	**The name of the analysis software package, sourced from a CV if available.**
Type:	ParamType
Attributes:	none
Subelements:	

|**Subelement Name** |**minOccurs** |**maxOccurs** |**Definition** |cvParam |1 |1 |A single entry from an ontology or a controlled vocabulary. |userParam |1 |1 |A single user-defined parameter.

Example Context:	<SoftwareName> <cvParam accession="MS:1002244" cvRef="PSI-MS" name="mzidLib:FalseDiscoveryRate"/> </SoftwareName>
cvParam Mapping Rules:	Path /MzIdentML/AnalysisSoftwareList/AnalysisSoftware/SoftwareName MUST supply a **child** term of MS:1001456 (analysis software) one or more times e.g.: MS:1000532 (Xcalibur) e.g.: MS:1000533 (Bioworks) e.g.: MS:1000534 (MassLynx) e.g.: MS:1000535 (FlexAnalysis) e.g.: MS:1000536 (Data Explorer) e.g.: MS:1000537 (4700 Explorer) e.g.: MS:1000539 (Voyager Biospectrometry Workstation System) e.g.: MS:1000551 (Analyst) e.g.: MS:1000600 (Proteios) e.g.: MS:1000601 (ProteinLynx Global Server) et al.

Example cvParams:	`<cvParam cvRef="PSI-MS" accession="MS:1001475" name="OMSSA"></cvParam>`
	`<cvParam cvRef="PSI-MS" accession="MS:1002237" name="mzidLib"></cvParam>`
	`<cvParam accession="MS:1001476" cvRef="PSI-MS" name="X\!Tandem"/>`
	`<cvParam accession="MS:1002239" cvRef="PSI-MS" name="mzidLib:Tandem2Mzid"/>`
	`<cvParam accession="MS:1002244" cvRef="PSI-MS" name="mzidLib:FalseDiscoveryRate"/>`
	`<cvParam accession="MS:1002242" cvRef="PSI-MS" name="mzidLib:Thresholder"/>`
	`<cvParam accession="MS:1002241" cvRef="PSI-MS" name="mzidLib:ProteoGrouper"/>`
	`<cvParam accession="MS:1002048" cvRef="PSI-MS" name="MS-GF+"/>`
	`<cvParam accession="MS:1001207" cvRef="PSI-MS" name="Mascot"/>`
	`<cvParam accession="MS:1001478" cvRef="PSI-MS" name="Mascot Parser"/>`
	`<cvParam accession="MS:1002076" cvRef="PSI-MS" name="PAnalyzer"/>`
	`<cvParam accession="MS:1001456" cvRef="PSI-MS" name="analysis software"/>`
	`<cvParam accession="MS:1000752" cvRef="PSI-MS" name="TOPP software"/>`
	`<cvParam cvRef="PSI-MS" accession="MS:1002458" name="PeptideShaker"/>`
	`<cvParam cvRef="PSI-MS" accession="MS:1002544" name="xiFDR"></cvParam>`
	`<cvParam cvRef="PSI-MS" accession="MS:1002543"`

Element <SourceFile>

Definition:	A file from which this mzIdentML instance was created.
Type:	SourceFileType
Attributes:	

|**Attribute Name** |**Data Type** |**Use** |**Definition** |id |xsd:string |required |An identifier is an unambiguous string that is unique within the scope (i.e. a document, a set of related documents, or a repository) of its use. |location |xsd:anyURI |required |The location of the data file. |name |xsd:string |optional |The potentially ambiguous common identifier, such as a human-readable name for the instance.

Subelements:	

|**Subelement Name** |**minOccurs** |**maxOccurs** |**Definition** |ExternalFormatDocumentation |0 |1 |A URI to access documentation and tools to interpret the external format of the ExternalData instance. For example, XML Schema or static libraries (APIs) to access binary formats. |FileFormat |1 |1 |The format of the ExternalData file, for example "tiff" for image files. |cvParam |0 |unbounded |A single entry from an ontology or a controlled vocabulary. |userParam |0 |unbounded |A single user-defined parameter.

Example Context:	<SourceFile location="C:\Users\hba041\My_Git_Applications\ peptide- shaker.wiki\data\2016_04_05\.PeptideShaker_un zip_temp\searchgui_out_PeptideShaker_temp\qE xactive01819 *t.xml" id="SourceFile_2">* <FileFormat> <cvParam cvRef="PSI-MS" accession="MS:1001401" name="X!Tandem xml format"/> </FileFormat> </SourceFile>

cvParam Mapping Rules:	Path /MzIdentML/DataCollection/Inputs/SourceFile
	MAY supply a **child** term of MS:1000561 (data file checksum type) one or more times
	e.g.: MS:1000568 (MD5)
	e.g.: MS:1000569 (SHA-1)

Element <SpecificityRules>

Definition:	**The specificity rules of the searched modification including for example the probability of a modification's presence or peptide or protein termini. Standard fixed or variable status should be provided by the attribute fixedMod.**
Type:	SpecificityRulesType
Attributes:	none
Subelements:	

|**Subelement Name** |**minOccurs** |**maxOccurs** |**Definition** |cvParam |1 |unbounded |A single entry from an ontology or a controlled vocabulary.

Example Context:	<SpecificityRules>
	<cvParam cvRef="PSI-MS" accession="MS:1002057" name="modification specificity protein N-
	term"/>
	</SpecificityRules>

cvParam Mapping Rules:	Path /MzIdentML/AnalysisProtocolCollection/SpectrumIdentificationProtocol/ModificationParams/SearchModifi
	cation/SpecificityRules
	MUST supply a **child** term of MS:1001056 (modification specificity rule) only once
	e.g.: MS:1001189 (modification specificity peptide N-term)
	e.g.: MS:1001190 (modification specificity peptide C-term)
	e.g.: MS:1001875 (modification motif)
	e.g.: MS:1001876 (modification probability)
	e.g.: MS:1002057 (modification specificity protein N-term)
	e.g.: MS:1002058 (modification specificity protein C-term)
Example cvParams:	<cvParam accession="MS:1001189" cvRef="PSI-MS" name="modification specificity peptide N-term"/>
	<cvParam accession="MS:1002057" cvRef="PSI-MS" name="modification specificity protein N-term"/>

Element <SpectraData>

Definition:	**A data set containing spectra data (consisting of one or more spectra).**
Type:	SpectraDataType
Attributes:	

|**Attribute Name** |**Data Type** |**Use** |**Definition** |id |xsd:string |required |An identifier is an unambiguous string that is unique within the scope (i.e. a document, a set of related documents, or a repository) of its use. |location |xsd:anyURI |required |The location of the data file. |name |xsd:string |optional |The potentially ambiguous common identifier, such as a human-readable name for the instance.

Subelements:	

Subelement Name	minOccurs	maxOccurs	Definition
ExternalFormatDocumentation	0	1	A URI to access documentation and tools to interpret the external format of the ExternalData instance. For example, XML Schema or static libraries (APIs) to access binary formats.
FileFormat	1	1	The format of the ExternalData file, for example "tiff" for image files.
SpectrumIDFormat	1	1	The format of the spectrum identifier within the source file

Example Context:	<SpectraData location="E:\Work\PSI\mzIdentML\ProteinInference\Rosetta2\Peaklistswithecoli\Rosetta2a_Ecoli_spectra *mgf" name="Rosetta2a_Ecoli_spectra.mgf" id="SID_1">* <FileFormat> <cvParam accession="MS:1001062" cvRef="PSI-MS" name="Mascot MGF format"/> </FileFormat> <SpectrumIDFormat> <cvParam accession="MS:1000774" cvRef="PSI-MS" name="multiple peak list nativeID format"/> </SpectrumIDFormat> ... </SpectraData>

Element <SpectrumIdentification>

Definition:	**An Analysis which tries to identify peptides in input spectra, referencing the database searched, the input spectra, the output results and the protocol that is run.**
Type:	SpectrumIdentificationType
Attributes:	

Attribute Name	Data Type	Use	Definition
activityDate	xsd:dateTime	optional	When the protocol was applied.
id	xsd:string	required	An identifier is an unambiguous string that is unique within the scope (i.e. a document, a set of related documents, or a repository) of its use.
name	xsd:string	optional	The potentially ambiguous common identifier, such as a human-readable name for the instance.
spectrumIdentificationList_ref	xsd:string	required	A reference to the SpectrumIdentificationList produced by this analysis in the DataCollection section.
spectrumIdentificationProtocol_ref	xsd:string	required	A reference to the search protocol used for this SpectrumIdentification.

Subelements:	

|**Subelement Name** |**minOccurs** |**maxOccurs** |**Definition** |InputSpectra |1 |unbounded |One of the spectra data sets used. |SearchDatabaseRef |1 |unbounded |One of the search databases used.

Example Context:	```<SpectrumIdentification spectrumIdentificationProtocol_ref="SearchProtocol_1_4299" spectrumIdentificationList_ref="SII_LIST_1_1_4299_120114_20_Orbi2_ZC_QC_220_HSAd0-d4-1to1-3_Din.raw" id="SpecIdent__4299_120114_20_Orbi2_ZC_QC_220_HSAd0-d4-1to1-3_Din.raw">```
	
	 ```<InputSpectra spectraData_ref="SD_4299_120114_20_Orbi2_ZC_QC_220_HSAd0-d4-1to1-3_Din.raw"></InputSpectra>```
	
	 ```<SearchDatabaseRef searchDatabase_ref="SDB_4299_203"></SearchDatabaseRef>```
	
	 ```</SpectrumIdentification>```

Element <SpectrumIdentificationItem>

Definition:	**An identification of a single (poly)peptide, resulting from querying an input spectra, along with the set of confidence values for that identification. PeptideEvidence elements should be given for all mappings of the corresponding Peptide sequence within protein sequences.**
Type:	SpectrumIdentificationItemType
Attributes:	

|**Attribute Name** |**Data Type** |**Use** |**Definition** |calculatedMassToCharge |xsd:double |optional |The theoretical mass-to-charge value calculated for the peptide in Daltons / charge. |calculatedPI |xsd:float |optional |The calculated isoelectric point of the (poly)peptide, with relevant modifications included. Do not supply this value if the PI cannot be calcuated properly. |chargeState |xsd:int |required |The charge state of the identified peptide. |experimentalMassToCharge |xsd:double |required |The mass-to-charge value measured in the experiment in Daltons / charge. |id |xsd:string |required |An identifier is an unambiguous string that is unique within the scope (i.e. a document, a set of related documents, or a repository) of its use. |massTable_ref |xsd:string |optional |A reference SHOULD be given to the MassTable used to calculate the sequenceMass only if more than one MassTable has been given. |name |xsd:string

|optional |The potentially ambiguous common identifier, such as a human-readable name for the instance. |passThreshold |xsd:boolean |required |Set to true if the producers of the file has deemed that the identification has passed a given threshold or been validated as correct. If no such threshold has been set, value of true should be given for all results. |peptide_ref |xsd:string |required |A reference to the identified (poly)peptide sequence in the Peptide element. |rank |xsd:int |required |For an MS/MS result set, this is the rank of the identification quality as scored by the search engine. 1 is the top rank. If multiple identifications have the same top score, they should all be assigned rank =1. For PMF data, the rank attribute may be meaningless and values of rank = 0 should be given. |sample_ref |xsd:string |optional |A reference should be provided to link the SpectrumIdentificationItem to a Sample if more than one sample has been described in the AnalysisSampleCollection.

Subelements:	

|**Subelement Name** |**minOccurs** |**maxOccurs** |**Definition** |PeptideEvidenceRef |0 |unbounded |Reference to the PeptideEvidence element identified. If a specific sequence can be assigned to multiple proteins and or positions in a protein all possible PeptideEvidence elements should be referenced here. |Fragmentation |0 |1 |The product ions identified in this result. |cvParam |0 |unbounded |A single entry from an ontology or a controlled vocabulary. |userParam |0 |unbounded |A single user-defined parameter.

Graphical Context:	

| Example Context: | <SpectrumIdentificationItem passThreshold="true" rank="1" peptide_ref="NMGGPYGGGNYGPGGSGGSGGYGGR_15.99491461956-ATAA-2" calculatedMassToCharge="1103.45377262

38118" experimentalMassToCharge="1103.454029541017" chargeState="2" id="SII_1920_1">

<PeptideEvidenceRef peptideEvidence_ref="PepEv_5164"/>

<Fragmentation>

<IonType charge="1" index="1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23">

<FragmentArray measure_ref="Measure_MZ" values="175.1193695 232.1403961 289.1618958

452.2268982 509.2429504 566.2701416 653.2999268 710.3218994 767.3411865 854.3770752 911.3968506

968.4257813 1065.472168 1122.492432 1285.553833 1399.612305 1456.62561 1513.645874 1570.669067

1733.721191 1830.793213 1887.808105 1944.829834"/>

<FragmentArray measure_ref="Measure_Int" values="5939.5844726563 4933.5014648438

13310.7265625 5077.6694335938 5685.9287109375 13253.552734375 7620.0947265625 7724.3696289063

16868.541015625 10552.126953125 11589.0576171875 7839.9741210938 47821.64453125 60335.71484375

21759.3984375 8742.5595703125 11512.0908203125 18130.890625 30577.375 3801.3923339844 8051.07421875

1954.5501708984 4844.9125976563"/>

<FragmentArray measure_ref="Measure_Error" values="4.173258879802688E-4 |

cvParam Mapping Rules:	Path /MzIdentML/DataCollection/AnalysisData/SpectrumIdentificationList/SpectrumIdentificationResult/Spect
	rumIdentificationItem
	MAY supply a **child** term of MS:1001405 (spectrum identification result details) one or more times
	e.g.: MS:1000016 (scan start time)
	e.g.: MS:1000796 (spectrum title)
	e.g.: MS:1000797 (peak list scans)
	e.g.: MS:1000798 (peak list raw scans)
	e.g.: MS:1000903 (product ion series ordinal)
	e.g.: MS:1000904 (product ion m/z delta)
	e.g.: MS:1000926 (product interpretation rank)
	e.g.: MS:1001030 (number of peptide seqs compared to each spectrum)
	e.g.: MS:1001035 (date / time search performed)
	e.g.: MS:1001036 (search time taken)
	et al.

Example cvParams:	`<cvParam accession="MS:1001330" cvRef="PSI-MS" value="2.5E-6" name="X\!Tandem:expect"/>`
	`<cvParam accession="MS:1001331" cvRef="PSI-MS" value="43.2" name="X\!Tandem:hyperscore"/>`
	`<cvParam accession="MS:1001250" cvRef="PSI-MS" value="0.0" name="local FDR"/>`
	`<cvParam accession="MS:1001868" cvRef="PSI-MS" value="0.0" name="distinct peptide-level q-value"/>`
	`<cvParam accession="MS:1001874" cvRef="PSI-MS" value="3.113325031133251E-4" name="FDRScore"/>`
	`<cvParam accession="MS:1000796" cvRef="PSI-MS" value="Locus:11.1.1.4652.4 File:"R1 p450 iTRAQ QS CEX11.wiff"" name="spectrum title"/>`
	`<cvParam accession="MS:1002049" cvRef="PSI-MS" value="129" name="MS-GF:RawScore"/>`
	`<cvParam accession="MS:1002050" cvRef="PSI-MS" value="143" name="MS-GF:DeNovoScore"/>`
	`<cvParam accession="MS:1002052" cvRef="PSI-MS" value="6.3746987E-25" name="MS-GF:SpecEValue"/>`
	`<cvParam accession="MS:1002053" cvRef="PSI-MS" value="1.3781529E-17" name="MS-GF:EValue"/>`
	`<cvParam accession="MS:1002054" cvRef="PSI-MS" value="0.0" name="MS-GF:QValue"/>`
	`<cvParam accession="MS:1002055" cvRef="PSI-MS" value="0.0" name="MS-GF:PepQValue"/>`
	`<cvParam accession="MS:1002351" cvRef="PSI-MS" value="0.0" name="PSM-level local FDR"/>`
	`<cvParam accession="MS:1002354" cvRef="PSI-MS" value="0.0" name="PSM-level q-value"/>`
	`<cvParam accession="MS:1002355" cvRef="PSI-MS" value="1.5603866050496166E-18" name="PSM-level FDRScore"/>`

Example userParams:	<userParam value="0" name="IsotopeError"/>
	<userParam value="HCD" name="AssumedDissociationMethod"/>

Example for peptide-level statistics:

```
<SpectrumIdentificationItem                                    chargeState="3"
experimentalMassToCharge="710.352539"
calculatedMassToCharge="710.352984"
peptide_ref="KMDLSDEGGGGVRYPGLHPK_##Oxidation(M):2"        rank="1"
passThreshold="false" id="SIR_3397_SII_1">

<PeptideEvidenceRef
peptideEvidence_ref="KMDLSDEGGGGVRYPGLHPK_generic|B_GENSCAN00
000016205_REVERSED|p:genscan_42_61"></PeptideEvidenceRef>

<cvParam    cvRef="PSI-MS"    accession="MS:1002356"    name="PSM-level
combined FDRScore" value="0.38641138028680944"></cvParam>

<userParam        name="search        engines        identifying        PSM"
value="2"></userParam>

<cvParam cvRef="PSI-MS" accession="MS:1002359" name="distinct peptide-
level local FDR" value="0.419570671870644"></cvParam>

<cvParam cvRef="PSI-MS" accession="MS:1001868" name="distinct peptide-
level q-value" value="0.4192650334075724"></cvParam>

<cvParam cvRef="PSI-MS" accession="MS:1002360" name="distinct peptide-
level FDRScore" value="0.41934590570107133"></cvParam>

<cvParam  cvRef="PSI-MS"  accession="MS:1002500"  name="peptide  passes
threshold" value="true"></cvParam>

<cvParam  cvRef="PSI-MS"  accession="MS:1002520"  name="peptide  group
ID" value="KMDLSDEGGGGVRYPGLHPK_##Oxidation(M):2"></cvParam>

</SpectrumIdentificationItem>
```

Example for crosslinking:

```
<SpectrumIdentificationItem                                    chargeState="4"
experimentalMassToCharge="0.0" peptide_ref="54600873_54605193_9_1_p1"
```

rank="1" passThreshold="false" id="SII_21_1">

<PeptideEvidenceRef
peptideEvidence_ref="pepevid_psm252621611_pep54605193_protP02768-A_target_137"></PeptideEvidenceRef>

<cvParam cvRef="PSI-MS" accession="MS:1002511" name="crosslink spectrum identification item" value="21"></cvParam>

<cvParam cvRef="PSI-MS" accession="MS:1002545" name="The xi result 'Score'." value="2.769918944845425"></cvParam>

</SpectrumIdentificationItem>

Element <SpectrumIdentificationItemRef>

Definition:	Reference(s) to the SpectrumIdentificationItem element(s) that support the given PeptideEvidence element. Using these references, it is possible to indicate which spectra were actually accepted as evidence for this peptide identification in the given protein.
Type:	SpectrumIdentificationItemRefType
Attributes:	

|Attribute Name |Data Type |Use |Definition |spectrumIdentificationItem_ref |xsd:string |required |A reference to the SpectrumIdentificationItem element(s).

Subelements:	none
Example Context:	<SpectrumIdentificationItemRef spectrumIdentificationItem_ref="SII_1000_1"></SpectrumIdentificationItemRef>

Example cvParams:	`<cvParam cvRef="PSI-MS" accession="MS:1001591" name="anchor protein"></cvParam>`
	
	`<cvParam cvRef="PSI-MS" accession="MS:1001593" name="group member with undefined relationship OR ortholog protein"></cvParam>`
	
	`<cvParam cvRef="PSI-MS" accession="MS:1002676" name="protein-pair-level global FDR" value="0.a:null:1.0:true"></cvParam>`
	
	`<cvParam cvRef="PSI-MS" accession="MS:1002677" name="residue-pair-level global FDR" value="0.a:58:0.04716981132075472:true"></cvParam>`
	
	`<cvParam cvRef="PSI-MS" accession="MS:1002415" name="protein group passes threshold" value="true"></cvParam>`
	
	`<cvParam cvRef="PSI-MS" accession="MS:1002404" name="count of identified protein" value="2"></cvParam>`

Element <SpectrumIdentificationList>

Definition:	Represents the set of all search results from SpectrumIdentification.
Type:	SpectrumIdentificationListType
Attributes:	

|Attribute Name |Data Type |Use |Definition |id |xsd:string |required |An identifier is an unambiguous string that is unique within the scope (i.e. a document, a set of related documents, or a repository) of its use. |name |xsd:string |optional |The potentially ambiguous common identifier, such as a human-readable name for the instance. |numSequencesSearched |xsd:long |optional |The number of database sequences searched against. This value should be provided unless a de novo search has been performed.

Subelements:	

|Subelement Name |minOccurs |maxOccurs |Definition |FragmentationTable |0 |1 |Contains the types of measures that will be reported in generic arrays for each SpectrumIdentificationItem e.g. product ion m/z, product ion intensity, product ion m/z error |SpectrumIdentificationResult |1

|unbounded |All identifications made from searching one spectrum. For PMF data, all peptide identifications will be listed underneath as SpectrumIdentificationItems. For MS/MS data, there will be ranked SpectrumIdentificationItems corresponding to possible different peptide IDs. |cvParam |0 |unbounded |A single entry from an ontology or a controlled vocabulary. |userParam |0 |unbounded |A single user-defined parameter.

Graphical Context:	Generated by XMLSpy www.altova.com

Example Context:	`<SpectrumIdentificationList xmlns="http://psidev.info/psi/pi/mzIdentML/1.2" id="SII_LIST_1">`
	
	`<SpectrumIdentificationResult spectrumID="index=6451" spectraData_ref="SD_COMBINED_SE_0" id="SIR_8947">`
	
	`<SpectrumIdentificationItem chargeState="2" experimentalMassToCharge="679.817322" calculatedMassToCharge="679.818488" peptide_ref="AVMDDFAAFVEK_##Oxidation(M):3" rank="1" passThreshold="false" id="SIR_8947_SII_1">`
	
	`<PeptideEvidenceRef peptideEvidence_ref="AVMDDFAAFVEK_generic`

A_ENSP00000401820	`p:known_378_389"></PeptideEvidenceRef>`
	
	`<PeptideEvidenceRef peptideEvidence_ref="AVMDDFAAFVEK_generic`

A_ENSP00000421027	`p:putative_420_431"></PeptideEvidenceRef>`
	
	`<PeptideEvidenceRef peptideEvidence_ref="AVMDDFAAFVEK_generic`

A_ENSP00000483421	p:known_357_368"></PeptideEvidenceRef>
	<PeptideEvidenceRef peptideEvidence_ref="AVMDDFAAFVEK_generic
A_ENSP00000480485	p:known_357_368"></PeptideEvidenceRef>
	... </SpectrumIdentificationList>

Element <SpectrumIdentificationProtocol>

Definition:	The parameters and settings of a SpectrumIdentification analysis.
Type:	SpectrumIdentificationProtocolType
Attributes:	

|**Attribute Name** |**Data Type** |**Use** |**Definition** |analysisSoftware_ref |xsd:string |required |The search algorithm used, given as a reference to the SoftwareCollection section. |id |xsd:string |required |An identifier is an unambiguous string that is unique within the scope (i.e. a document, a set of related documents, or a repository) of its use. |name |xsd:string |optional |The potentially ambiguous common identifier, such as a human-readable name for the instance.

Subelements:	

|**Subelement Name** |**minOccurs** |**maxOccurs** |**Definition** |SearchType |1 |1 |The type of search performed e.g. PMF, Tag searches, MS-MS |AdditionalSearchParams |0 |1 |The search parameters other than the modifications searched. |ModificationParams |0 |1 |The specification of static/variable modifications (e.g. Oxidation of Methionine) that are to be considered in the spectra search. |Enzymes |0 |1 |The list of enzymes used in experiment |MassTable |0 |unbounded |The masses of residues used in the search. |FragmentTolerance |0 |1 |The tolerance of the search given as a plus and minus value with units. |ParentTolerance |0 |1 |The tolerance of the search given as a plus and minus value with units. |Threshold |1 |1 |The threshold(s) applied to determine that a result is significant. If multiple terms are used it is assumed that all conditions are satisfied by the passing results. |DatabaseFilters |0 |1 |The specification of filters applied to the database searched. |DatabaseTranslation |0 |1 |A specification of how a nucleic acid sequence database was translated for searching.

Graphical Context:	image::img/mzidentml/image11.png[http://www.peptideatlas.org/PSI/schemas/mzIdentML/1.2/figures/SpectrumIdentificationProtocol.png,width=557,height=813]

Example Context:	\<SpectrumIdentificationProtocol id="SIP_1058955438233790425" analysisSoftware_ref="SOF_10581839310406754 333"> \<SearchType> \<cvParam accession="MS:1001083" cvRef="PSI-MS" name="ms-ms search"/> \</SearchType> \<AdditionalSearchParams> \<cvParam accession="MS:1002494" cvRef="PSI-MS" name="crosslinking search"/> \<userParam name="input_consensusXML" unitName="xsd:string" value="leitner004.consensusXML"/> ... \</SpectrumIdentificationProtocol>

Element \<SpectrumIdentificationResult>

Definition:	**All identifications made from searching one spectrum. For PMF data, all peptide identifications will be listed underneath as SpectrumIdentificationItems. For MS/MS data, there will be ranked SpectrumIdentificationItems corresponding to possible different peptide IDs.**
Type:	SpectrumIdentificationResultType
Attributes:	

|**Attribute Name** |**Data Type** |**Use** |**Definition** |id |xsd:string |required |An identifier is an unambiguous string that is unique within the scope (i.e. a document, a set of related documents, or a repository) of its use. |name |xsd:string |optional |The potentially ambiguous common identifier, such as a human-readable name for the instance. |spectraData_ref |xsd:string |required |A reference to a spectra data set (e.g. a spectra file). |spectrumID |xsd:string |required |The locally unique id for the spectrum in the spectra data set specified by SpectraData_ref. External guidelines are provided on the use of consistent identifiers for spectra in different external formats.

Subelements:	

|**Subelement Name** |**minOccurs** |**maxOccurs** |**Definition** |SpectrumIdentificationItem |1 |unbounded |An identification of a single (poly)peptide, resulting from querying an input spectra, along with the set of confidence values for that identification. PeptideEvidence elements should be

given for all mappings of the corresponding Peptide sequence within protein sequences. |cvParam |0 |unbounded |A single entry from an ontology or a controlled vocabulary. |userParam |0 |unbounded |A single user-defined parameter.

Example Context:	`<SpectrumIdentificationResult spectrumID="index=7665" spectraData_ref="SD_COMBINED_SE_0" id="SIR_7191">` `<SpectrumIdentificationItem chargeState="4" experimentalMassToCharge="1123.974121" calculatedMassToCharge="1123.968707" peptide_ref="CRCQYSGVNNLCH`	A_ENSP00000376692_REVERSED	p: novel_575_612"></PeptideEvidenceRef> `<PeptideEvidenceRef peptideEvidence_ref="CRCQYSGVNNLCHTSSHCPNQGSTCENVDTCLKPDEPK_generic`	B_GENSCAN000000366974_REVERSED	p: genscan_494_531"></PeptideEvidenceRef> `<PeptideEvidenceRef peptideEvidence_ref="CRCQYSGVNNLCHTSSHCPNQGSTCENVDTCLKPDEPK_generic`	A_ENSP00000471297_REVERSED	p: putative_641_678"></PeptideEvidenceRef> `<PeptideEvidenceRef peptideEvidence_ref="CRCQYSGVNNLCHTSSHCPNQGSTCENVDTCLKPDEPK_generic`	A_ENSP00000319883_REVERSED	p: known_633_670"></PeptideEvidenceRef> `<PeptideEvidenceRef peptideEvidence_ref="CRCQYSGVNNLCHTSSHCPNQGSTCENVDTCLKPDEPK_generic`	A_ENSP00000472280_REVERSED	p: putative_622_659"></PeptideEvidenceRef> ... </SpectrumIdentificationResult>

Element <SpectrumIDFormat>

Definition:	The format of the spectrum identifier within the source file
Type:	SpectrumIDFormatType
Attributes:	none
Subelements:	

|**Subelement Name** |**minOccurs** |**maxOccurs** |**Definition** |cvParam |1 |1 |A single entry from an ontology or a controlled vocabulary.

Example Context:	<SpectrumIDFormat> <cvParam accession="MS:1000774" cvRef="PSI-MS" name="multiple peak list nativeID format"/> </SpectrumIDFormat>

cvParam Mapping Rules:	Path /MzIdentML/DataCollection/Inputs/SpectraData/ SpectrumIDFormat MUST supply a **child** term of MS:1000767 (native spectrum identifier format) only once e.g.: MS:1000768 (Thermo nativeID format) e.g.: MS:1000769 (Waters nativeID format) e.g.: MS:1000770 (WIFF nativeID format) e.g.: MS:1000771 (Bruker/Agilent YEP nativeID format) e.g.: MS:1000772 (Bruker BAF nativeID format) e.g.: MS:1000773 (Bruker FID nativeID format) e.g.: MS:1000774 (multiple peak list nativeID format) e.g.: MS:1000775 (single peak list nativeID format) e.g.: MS:1000776 (scan number only nativeID format) e.g.: MS:1000777 (spectrum identifier nativeID format) et al. MUST supply a **child** term of MS:1001529 (spectra data details) only once e.g.: MS:1001530 (mzML unique identifier) e.g.: MS:1001531 (spectrum from ProteinScape database nativeID format) e.g.: MS:1001532 (spectrum from database string nativeID format)
Example cvParams:	\<cvParam cvRef="PSI-MS" accession="MS:1000774" name="multiple peak list nativeID format">\</cvParam>

Element <SubSample>

Definition:	References to the individual component samples within a mixed parent sample.
Type:	SubSampleType
Attributes:	

|**Attribute Name** |**Data Type** |**Use** |**Definition** |sample_ref |xsd:string |required |A reference to the child sample.

Subelements:	none
Example Context:	

Element <SubstitutionModification>

Definition:	A modification where one residue is substituted by another (amino acid change).
Type:	SubstitutionModificationType
Attributes:	

|**Attribute Name** |**Data Type** |**Use** |**Definition** |avgMassDelta |xsd:double |optional |Atomic mass delta considering the natural distribution of isotopes in Daltons. This should only be reported if the original amino acid is known i.e. it is not "X" |location |xsd:int |optional |Location of the modification within the peptide - position in peptide sequence, counted from the N-terminus residue, starting at position 1. Specific modifications to the N-terminus should be given the location 0. Modification to the C-terminus should be given as peptide length + 1. |monoisotopicMassDelta |xsd:double |optional |Atomic mass delta when assuming only the most common isotope of elements in Daltons. This should only be reported if the original amino acid is known i.e. it is not "X" |originalResidue a| xsd:string with restriction

[ABCDEFGHIJKLMNOPQRSTUVWXYZ?\-]{1}

|required |The original residue before replacement. |replacementResidue a| xsd:string with restriction

[ABCDEFGHIJKLMNOPQRSTUVWXYZ?\-]{1}

|required |The residue that replaced the originalResidue.

Subelements:	none
Example Context:	<SubstitutionModification originalResidue="X" replacementResidue="I" location="10" />

Element <Threshold>

Definition:	**Depending on context (SpectrumIdentificationProtocol or ProteinDetectionProtocol):** **1: The threshold(s) applied to determine that a spectrum identification is significant. If multiple terms are used it is assumed that all conditions are satisfied by the passing results.** **2: The threshold(s) applied to determine that a protein detection is significant. If multiple terms are used it is assumed that all conditions are satisfied by the passing results.**
Type:	ParamListType
Attributes:	none
Subelements:	

|**Subelement Name** |**minOccurs** |**maxOccurs** |**Definition** |cvParam |1 |unbounded |A single entry from an ontology or a controlled vocabulary. |userParam |1 |unbounded |A single user-defined parameter.

Example Context:	<Threshold> <cvParam cvRef="PSI-MS" accession="MS:1001364" name="distinct peptide-level global FDR" value="1.0"/> <cvParam cvRef="PSI-MS" accession="MS:1002350" name="PSM-level global FDR" value="1.0"/> <cvParam cvRef="PSI-MS" accession="MS:1002567" name="phosphoRS score threshold" value="95.0"/> <cvParam cvRef="PSI-MS" accession="MS:1002557" name="D-Score threshold" value="95.0"/> </Threshold>

cvParam Mapping Rules:	Path /MzIdentML/AnalysisProtocolCollection/SpectrumIdentificationProtocol/Threshold
	MAY supply a **child** term of MS:1001302 (search engine specific input parameter) one or more times
	e.g.: MS:1001005 (SEQUEST:CleavesAt)
	e.g.: MS:1001007 (SEQUEST:OutputLines)
	e.g.: MS:1001009 (SEQUEST:DescriptionLines)
	e.g.: MS:1001026 (SEQUEST:NormalizeXCorrValues)
	e.g.: MS:1001028 (SEQUEST:SequenceHeaderFilter)
	e.g.: MS:1001032 (SEQUEST:SequencePartialFilter)
	e.g.: MS:1001037 (SEQUEST:ShowFragmentIons)
	e.g.: MS:1001038 (SEQUEST:Consensus)
	e.g.: MS:1001042 (SEQUEST:LimitTo)
	e.g.: MS:1001046 (SEQUEST:sort by dCn)
	et al.
	MAY supply a **child** term of MS:1001153 (search engine specific score) one or more times
	e.g.: MS:1001154 (SEQUEST:probability)
	e.g.: MS:1001155 (SEQUEST:xcorr)
	e.g.: MS:1001156 (SEQUEST:deltacn)
	e.g.: MS:1001157 (SEQUEST:sp)
	e.g.: MS:1001158 (SEQUEST:Uniq)
	e.g.: MS:1001159 (SEQUEST:expectation value)
	e.g.: MS:1001160 (SEQUEST:sf)
	e.g.: MS:1001161 (SEQUEST:matched ions)

Example cvParams:	<cvParam cvRef="PSI-MS" accession="MS:1001494" name="no threshold"></cvParam> <cvParam accession="MS:1002351" cvRef="PSI-MS" value="0.01" name="PSM-level local FDR"/> <cvParam accession="MS:1001316" cvRef="PSI-MS" value="0.05" name="Mascot:SigThreshold"/> <cvParam cvRef="PSI-MS" accession="MS:1001364" name="distinct peptide-level global FDR" value="1.0"/> <cvParam cvRef="PSI-MS" accession="MS:1002350" name="PSM-level global FDR" value="1.0"/> <cvParam cvRef="PSI-MS" accession="MS:1002567" name="phosphoRS score threshold" value="95.0"/> <cvParam cvRef="PSI-MS" accession="MS:1002557" name="D-Score threshold" value="95.0"/> <cvParam cvRef="PSI-MS" accession="MS:1002369" name="protein group-level global FDR" value="0.01"/>

Element <TranslationTable>

Definition:	**The table used to translate codons into nucleic acids e.g. by reference to the NCBI translation table.**
Type:	TranslationTableType
Attributes:	

|**Attribute Name** |**Data Type** |**Use** |**Definition** |id |xsd:string |required |An identifier is an unambiguous string that is unique within the scope (i.e. a document, a set of related documents, or a repository) of its use. |name |xsd:string |optional |The potentially ambiguous common identifier, such as a human-readable name for the instance.

Subelements:	

|**Subelement Name** |**minOccurs** |**maxOccurs** |**Definition** |cvParam |0 |unbounded |A single entry from an ontology or a controlled vocabulary.

Example Context:	
cvParam Mapping Rules:	Path /MzIdentML/AnalysisProtocolCollection/SpectrumIdentificationProtocol/DatabaseTranslation/Translation Table MUST supply term MS:1001410 (translation start codons) only once MUST supply term MS:1001025 (translation table) only once MUST supply term MS:1001423 (translation table description) only once

Element \<userParam>

Definition:	**In case more information about the ions annotation has to be conveyed, that has no fit in FragmentArray. Note: It is suggested that the value attribute takes the form of a list of the same size as FragmentArray values. However, there is no formal encoding and it cannot be expeceted that other software will process or impart that information properly.**
Type:	UserParamType
Attributes:	

|Attribute Name |Data Type |Use |Definition |name |xsd:string |required |The name of the parameter. |type |xsd:string |optional |The datatype of the parameter, where appropriate (e.g.: xsd:float). |unitAccession |xsd:string |optional |An accession number identifying the unit within the OBO foundry Unit CV. |unitCvRef |xsd:string |optional |If a unit term is referenced, this attribute MUST refer to the CV 'id' attribute defined in the cvList in this file. |unitName |xsd:string |optional |The name of the unit. |value |xsd:string |optional |The user-entered value of the parameter.

Subelements:	none

Example Context:	<userParam value="VLENAEGDR; ASSGLNEDEIQK; MQELAQVSQK; KTAEDYLGEPVTEAVITVPAYFNDAQR; SLGQFNLDGINPAPR; MPMVQK; IIAADNGDAWVEVK; DVSIMPFK; KDVNPDEAVAIGAAVQGGVLTGDVK; KFEELVQTR; NDPLAMQR; VAEFFGK; QVEEAGDKLPADDK; MAPPQISAEVLKK; KQVEEAGDKLPADDK; LINYLVEEFK; MAPPQISAEVLK; QAVTNPQNTLFAIK; TFEVLATNGDTHLGGEDFDSR; VALQDAGLSVSDIDDVILVGGQTR; FQDEEVQR" name="unique peptides"/>

Specific Comments on schema

In this section, several points of documentation are elaborated beyond the core specification in Section 6..

File extension and compression

It is noted that standard file compression algorithms greatly reduce the mzIdentML file sizes, speeding up file transfers and uploads / downloads. It is also noted that software implementing mzIdentML import or export will be expected to benefit in performance from working with compressed mzIdentML, since the compression and decompression algorithms are expected to give significant performance gains over disk access times for non-compressed files. As such, it is RECOMMENDED that mzIdentML files are compressed using gzip from all software that exports mzIdentML and software that imports SHOULD be expected to read gzipped files, as well as native (non-compressed) mzIdentML files. The file extension for native mzIdentML files SHOULD be ".mzid" and for compressed files SHOULD be "mzid.gz".

Referencing elements within the document

A number of elements within the schema have an attribute which is used to reference an element elsewhere in the file using the unique identifier of the referenced element. These attributes are named following the convention: "[elementName]_ref". The uniqueness of the value in the "id" attribute of elements is validated using xsd:key, and the integrity of the reference is validated using xsd:keyref, defined within the schema.

Searches against nucleotide sequences

The "seq" attribute on <DBSequence> SHOULD contain the nucleic acid sequence if a nucleic acid database was searched (rather than up to six translated sequences). <Peptide> represents the identified amino acid sequence (including modifications) and, as such, the <peptideSequence> elements SHOULD store the translated amino acid sequences. <PeptideEvidence> contains the *DBSequence_Ref* together with the translation frame and a *TranslationTable_Ref* attribute (see below). The *Peptide_Ref* is done in <SpectrumIdentificationItem> as in the case for an amino acid database. If protein detection is performed, there are <PeptideHypothesis> elements referencing <PeptideEvidence> elements from <SpectrumIdentificationItem> sections. For clarification, see the example instance document for a nucleic acid search (Section 5.3).

In the <SpectrumIdentificationProtocol>, <TranslationTable> is used to specify how nucleic acid sequences are translated into amino acid sequences as follows:

<DatabaseTranslation frames="1 2 3 -1 -2 -3">

<TranslationTable id="TT_1" name="Standard">

<cvParam accession="MS:1001025" name="translation table" cvRef="PSI-MS" value="FFLLSSSSYY**CC*WLLLLPPPPHHQQRRRRIIIMTTTTNNKKSSRRVVVVAAAADDEEGGGG" />

<cvParam accession="MS:1001410" name="translation start codons" cvRef="PSI-MS" value="---M--------------M--------------M------------------------" />

<cvParam accession="MS:1001423" name="translation table description" cvRef="PSI-MS" value="http://www.ncbi.nlm.nih.gov/Taxonomy/taxonomyhome.html/index.cgi?chapter=cgencodes #SG1" />

</TranslationTable>

<TranslationTable id="TT_2" name="Vertebrate Mitochondrial">

<cvParam accession="MS:1001025" name="translation table" cvRef="PSI-MS" value="FFLLSSSSYY**CCWWLLLLPPPPHHQQRRRRIIMMTTTTNNKKSS**VVVVAAAADDEEGGGG" />

<cvParam accession="MS:1001410" name="translation start codons" cvRef="PSI-MS" value="-------------------------------MMMM--------------M------------" />

<cvParam accession="MS:1001423" name="translation table description" cvRef="PSI-MS" value="http://www.ncbi.nlm.nih.gov/Taxonomy/taxonomyhome.html/index.cgi?chapter=cgencodes #SG2" />

</TranslationTable>

The attribute "frames" specifies which frames are considered and one or more translation tables can be specified using CV parameters. The translation table is defined here: http://www.ncbi.nlm.nih.gov/IEB/ToolBox/SDKDOCS/SEQFEAT.HTML#_Genetic_Codes:

"The genetic codes themselves are arrays of 64 amino acid codes. The index to the position in the array of the amino acid is derived from the codon by the following method:

index = (base1 16) + (base2 4) + (base3 1)

where T=0, C=1, A=2, G=3"

The same encoding technique is used to specify start codons. Alphabet names are prefixed with "s" (e.g. sncbieaa) to indicate start codon arrays. Each cell of a start codon array contains either the gap code ("-" for ncbieaa) or an amino acid code if it is valid to use the codon as a start codon. Currently all starts are set to code for methionine, since it has never been convincingly demonstrated that a protein can start with any other amino acid. However, if other amino acids are shown to be used as starts, this structure can easily accommodate that information.

For each peptide, the frame and translation table should be specified in the <PeptideEvidence> element:

<PeptideEvidence id="1" TranslationTable_ref="TT_1" frame="1" />

Reporting peptide and protein identifications passing a significance threshold

The elements <SpectrumIdentificationItem> and <ProteinDetectionHypothesis> have a mandatory

Boolean attribute *passThreshold* that allows a file producer to indicate that an identification has passed a given threshold or that it has been manually validated. Depending on the intended purpose of the file, the file producer MAY wish to report a number of identifications that fall below the given significance threshold, for example to allow global statistical analyses to be performed which are not possible if only identifications passing the threshold are reported. Thresholds for peptide-spectrum matches or for protein identification should be encoded as instances of <cvParam> within <SpectrumDetectionProtocol> or <ProteinDetectionProtocol>, respectively, as follows. If the file producer does not want to indicate that a threshold has been set, all identifications MUST have passThreshold = "true" and the "no threshold" CV term should be given within the protocols.

<SpectrumIdentificationProtocol id="SIP" AnalysisSoftware_ref="AS_mascot_server">

...

<Threshold>
<cvParam accession="MS:1001316" name="mascot:SigThreshold" cvRef="PSI-MS" value="0.05"/>
</Threshold>

<ProteinDetectionProtocol id="PDP_MascotParser_1" AnalysisSoftware_ref="AS_mascot_parser">

...

<Threshold>

<cvParam accession="MS:1001316" name="mascot:SigThreshold" cvRef="PSI-MS" value="0.05"/>

</Threshold>

The reporting of significance thresholds at the PSM and peptide level (mzIdentML 1.2) is explained in section 5.2.7. Reporting of threshold for modification position (also mzIdentML 1.2) is explained in section 5.2.8.

Using decoy databases to set different thresholds of false discovery rate

mzIdentML supports the reporting of searches against decoy databases, constructed and searched using many of the currently known methods. A <SpectrumIdentificationItem> can be marked as matching a decoy peptide using the *isDecoy* attribute of the referenced <PeptideEvidence> element, thus allowing the false discovery rate to be calculated across an entire file. The *DBSequence_Ref* references the decoy protein record.

Implementers of the format SHOULD report the peptide identifications that pass the threshold they wish to communicate to a consumer of the data. For example, a threshold could be set by p-value, false discovery rate, by a native search engine score (or a more complex system documented with CV terms in <Threshold>), and those peptides reported (passing the threshold) are used to determine which proteins have been detected. It is not guaranteed that a consumer of an mzIdentML file will be able to calculate other results, or global false discovery rates, using different thresholds from the reported information, although in some circumstances they may be able to, for

example, if a user reports the complete output of a search against a target and decoy search.

<SequenceCollection>

<PeptideEvidence isDecoy="true" post="D" pre="K" end="404" start="392"

peptide_ref="HAVGGRYSSLLCK__57.0215@C$403;_"

dBSequence_ref="dbseq_REV_psu|NC_LIV_113200" id="PE6_2_4"/>

<SequenceCollection>

<DataCollection>

<Inputs>

<SearchDatabase location="/localdirectory/18.E_coli_K12_edit.fasta" id="K12_nosignal" name="K12" numDatabaseSequences="9376" releaseDate="01-2008-08-2008" version="1.0" >

<FileFormat>

<cvParam accession="MS:1001348" name="FASTA format" cvRef="PSI-MS"/>

</FileFormat>

<DatabaseName>

<userParam name="18.E_coli_K12_edit.fasta" />

</DatabaseName>

<cvParam accession="MS:1001197" name="**DB composition target+decoy**" cvRef="PSI-MS"/>

<cvParam accession="MS:1001283" name="**decoy DB accession regexp**" value="**Rnd**" cvRef="PSI-MS"/>

<cvParam accession="MS:1001195" name="**decoy DB type reverse**" cvRef="PSI-MS"/>

</SearchDatabase>

</Inputs>

<AnalysisData>

<SpectrumIdentificationList>

<SpectrumIdentificationResult>

<SpectrumIdentificationItem passThreshold="false" rank="1"
peptide_ref="HAVGGRYSSLLCK__57.0215@C$403;_"
experimentalMassToCharge="1448.756" chargeState="2" id="SII_6_1">
<PeptideEvidenceRef peptideEvidence_ref="PE6_2_4"/>

```
<cvParam accession="MS:1001329" name="OMSSA:pvalue" cvRef="PSI-MS"

value="0.00073351" />

</SpectrumIdentificationItem>

<SpectrumIdentificationResult>

</SpectrumIdentificationList>

</AnalysisData>

</DataCollection>
```

Database Filter

The format can specify that a sequence database has been filtered, for example based on pI, protein mass, taxonomy or even a set of accession numbers for a second pass search. For example, all animals except mice would be encoded as (NCBI:33208 is metazoa, NCBI:10090 is *Mus musculus*):

```
<DatabaseFilters>

<Filter>

<FilterType>

<cvParam accession="MS:1001020" name="DB filter taxonomy" cvRef="PSI-MS" />

</FilterType>

<Include>

<cvParam accession="MS:1001467" name="taxonomy: NCBI TaxID" cvRef="PSI-MS" value="33208"/>

</Include>

<Exclude>

<cvParam accession="MS:1001467" name="taxonomy: NCBI TaxID" cvRef="PSI-MS" value="10090"/>

</Exclude>

</Filter>

</DatabaseFilters>
```

Types of parameters and values

There are several types for parameters that are used in the schema:

<ParamListType>: A list (i.e. unbounded number) of <ParamGroup> elements.

<ParamGroup>: A choice between <cvParam> or <userParam> elements.

<ParamType>: A single reference to <ParamGroup>, which allows a choice between either <cvParam> or <userParam> elements at the specified point in the schema.

<cvParamType>: A single entry from an ontology or a CV. Attributes: *accession, cvRef, name, value, unitAccession, unitName, unitCvRef.*

<userParamType>: A single user-defined parameter. Attributes: *name, value, unitAccession, unitName, unitCvRef.*

Reporting fragmentation ions

mzIdentML employs an array type structure to support the reporting of ion types identified in an MS/MS analysis, coupled with CV parameters to retain flexibility in the types of ions that can be reported.

A brief example is given here to explain how these structures should be used where *y11*, *y8* and *y7* have been identified with charge = 2+. First, the types of measures to be reported are given in the <FragmentationTable> using <cvParam> instances. Second, each <SpectrumIdentificationItem> contains an index of values (11, 8 and 7 for each y ion) and parallel arrays that reference back to each <Measure> defined in the <FragmentationTable>. In the example, the *y8* ion has a product ion *m/z* = 436.4, product ion intensity = 11 and product ion *m/z* error = 0.1284 (the second position in the index of each array).

<FragmentationTable>

<Measure id="m_mz">

<cvParam cvRef="PSI-MS" accession="MS:1001225" name="product ion m/z"/>

</Measure>

<Measure id="m_intensity">

<cvParam cvRef="PSI-MS" accession="MS:1001226" name="product ion intensity"/>

</Measure>

<Measure id="m_error">

<cvParam cvRef="PSI-MS" accession="MS:1001227" name="product ion m/z error" unitAccession="MS:1000040" unitName="m/z" unitCvRef="PSI-MS"/>

</Measure>

</FragmentationTable>

...

<IonType index="11 8 7" charge="2">

```
<cvParam cvRef="PSI-MS" accession="MS:1001220" name="frag: y ion"/>

<FragmentArray values="551.3 436.4 380.1 " measure_ref="m_mz"/>

<FragmentArray values="800 11 46" measure_ref="m_intensity"/>

<FragmentArray values="0.4752 0.1284 0.3704" measure_ref="m_error"/>

</IonType>
```

Internal fragments and immonium ions

mzIdentML supports the reporting of internal fragment ions, of which an immonium ion is a special case comprising a single side chain (http://www.matrixscience.com/help/fragmentation_help.html). For internal and immonium ions, the index is used in two different ways. Internal fragments are reported using the index structure to identify the start and end of the ion within the sequence. The example shows how the index performs this different role, as it identifies pairs of internal ions: *ya2-5, ya3-7, ya3-8, ya4-8, ya5-8, ya5-11, ya8-11*.

```
<IonType index="2 5 3 7 3 8 4 8 5 8 5 11 8 11" charge="1">

<FragmentArray values="315.2 388.1 501.4 444.1 342.8 669.901495 412.4 " measure_ref="m_mz"/>

<FragmentArray values="44 63 10430 75 48 6420 31" measure_ref="m_intensity"/>

<FragmentArray values="-0.0027 -0.1191 0.0969 -0.1817 -0.4340 0.4721 0.1082" measure_ref="m_error"/>

<cvParam cvRef="PSI-MS" accession="MS:1001366" name="frag: internal ya ion"/>

</IonType>
```

For immonium ions, the index is the position of the identified ion within the peptide sequence. If the peptide contains the same amino acid in multiple positions that cannot be distinguished, all positions should be given. Example, where immonium ions have been found matching T and G in the following peptide sequence FGGEENTY (positions 2 or 3, and position 7):

```
<IonType charge="1" index="2 3 7">

<FragmentArray values="288.2 286.1 387.2 371.127841 " measure_ref="m_mz"/>

<FragmentArray values="2137 83 656 1663" measure_ref="m_intensity"/>

<FragmentArray values="0.0260 -0.1125 -0.0602 -0.1011" measure_ref="m_error"/>

<cvParam cvRef="PSI-MS" accession="MS:1001239" name="frag: immonium ion"/>

</IonType>
```

Encoding Neutral loss fragment ions

The encoding of the identification of neutral loss fragment ions has changed from version 1.1 to

version 1.2.0. The CV previously contained an attempt to enumerate all possible neutral losses from all types of fragment ions, leading to a long and incomplete list of possible terms. As such, pairs of CV terms are now allowed in version 1.2.0 to describe both the ion type and the type of neutral loss, as follows:

<IonType charge="2" index="3">

<FragmentArray measure_ref="Measure_MZ" values="165.0899353"/>

<FragmentArray measure_ref="Measure_Int" values="2058.2841796875"/>

<FragmentArray measure_ref="Measure_Error" values="0.007899949187986977"/>

<cvParam cvRef="PSI-MS" accession="MS:1001220" name="frag: y ion"/>

<cvParam cvRef="PSI-MS" accession="MS:1000336" name="neutral loss" value="H3N"/>

</IonType>

Enzyme definition

The <SpectrumIdentificationProtocol> SHOULD contain a specification of which enzyme (if any) was applied in the search. The element <Enzyme> has optional sub-elements for specifying the <EnzymeName> using a CV term and the cleavage site, using a regular expression. Regular expressions should be encoded following the notation of Perl Compatible Regular Expressions (PCRE regex, http://www.pcre.org, matching the syntax and semantics of Perl version 5). The PSI-MS CV contains terms for the most common enzymes with pre-defined regular expressions (Table 6). If the enzyme used is present in the PSI-MS CV, the term MUST be encoded under <EnzymeName> unless the rule given in the CV does not match that used by the software or if the enzyme used is not present in the CV, in which case the regular expression used MUST be given in the element <SiteRegexp>. If the <EnzymeName> element is used, the regular expression MAY also be provided additionally. For a no enzyme search, (i.e. one where there may be a cleavage at any residue), the CV term MS:1001091 'NoEnzyme' MUST be specified, and the missedCleavages and semiSpecific attributes SHOULD NOT be specified. If two or more enzymes are used, multiple <Enzyme> elements SHOULD be provided rather than trying to build a regular expression covering all cleavage sites. If the software uses a name for an enzyme other than the one specified in the CV, a user param term MAY also be given.

The following guidelines SHOULD be followed when generating regular expressions in an instance document for enzymes not present in the CV: 1) use the PCRE supplied negation syntax for look-ahead and look-behind assertions and 2) use the most compact representation possible for a regex. The start of a match specifies the cleavage point. For example the enzyme trypsin, which cleaves following a K or R residue unless the next residue is P, has the regular expression:

(?⇐[KR])(?!P)

The ?⇐ is a "zero-width positive look-behind assertion", and [] means one of this character set. So, this rule is to look behind for a K or R. ?! is a zero-width positive look-ahead assertion, and ?!P means any character that is not P. An example of an "N-term" enzyme is Asp-N which cleaves before D or B. This can be described using the PCRE:

(?=[BD])

The ?= is a "zero-width positive look-ahead assertion."

A simple 3 line perl program can be written to test a regular expression:

$protein = "ABCDKPEFGHIJKLMNOPQRSTUVWXYZ";

@peptides = split(/(?<=[KR])(?!P)/, $protein);

print join "\n", @peptides;

The program returns:

ABCDKPEFGHIJK

LMNOPQR

STUVWXYZ

Enzyme Name	Regular expression
Trypsin	(?<=[KR])(?!P)
Arg-C	(?<=R)(?!P)
Asp-N	(?=[BD])
Asp-N_ambic	(?=[DE])
Chymotrypsin	(?<=[FYWL])(?!P)
CNBr	(?<=M)
Formic_acid	?<=D
?=D	Lys-C
(?<=K)(?!P)	Lys-C/P
(?<=K)	PepsinA
(?<=[FL])	TrypChymo
(?<=[FYWLKR])(?!P)	Trypsin/P
(?<=[KR])	V8-DE
(?<=[BDEZ])(?!P)	V8-E
(?<=[EZ])(?!P)	Leukocyte elastase
(?<=[ALIV])(?!P)	Proline endopeptidase
(?<=[HKR]P)(?!P)	Glutamyl endopeptidase
(?<=[^E]E)	2-iodobenzoate

Table 6. Common enzymes and the cleavage site specified as regular expressions as represented in the PSI-MS CV.

Unknown modifications

In version 1.1.0 onwards of mzIdentML there has been a change with respect to how "unknown modifications" (i.e. those not present in an allowed CV) are reported on peptides. In version 1.0, <userParam> elements were allowed on <Peptide> to capture these modifications. In version 1.1.0 onwards, only <cvParam> elements can be given on <Peptide> and a term "unknown modification" has been added to the PSI-MS CV. This term MUST only be used if the identified modification is not present in Unimod (or other allowed CV such as PSI-MOD, although PSI-MOD is now deprecated), according to the identity of the residue modified and the delta mass, within the parent tolerance specified in the search. The semantic validator will check any uses of the "unknown modification" term (MS:1001460) and reject files if the modification is present in Unimod.

Identifications based on multiple mass spectra

This Section has been added to version 1.3.0 of the specification. Some analysis workflows utilise multiple spectra to arrive at a given identification, for instance, the following crosslinking search strategies:

> (i) "light" and "heavy" isotopes of the crosslinker are used as a pair, combined together and searched once;
>
> (ii) multiple spectra of the same precursor are acquired, *e.g.* using different fragmentation techniques like EThcD;
>
> (iii) when using a cleavable crosslinker and both MS3 spectra of the cleaved peptides and the MS2 spectrum of the crosslinked peptide pair are considered in the identification process.

mzIdentML 1.2.0 included a method for encoding such cases using the "combined spectra" type of input file format. This essentially associates a single <SpectrumIdentificationResult> element with a comma separated list of spectrum identifiers. This would work for crosslinking search strategy (i) if all the spectra contributing to an identification share the same acquisition settings, but it presents a problem for cases (ii) and (iii). The <SpectrumIdentificationResult> element with the comma separated list of identifiers is inside a single <SpectrumIdentificationList> element and this can only be associated with a single <SpectrumAnalysisProtocol> element. But in cases (ii) and (iii) the spectra referenced have different acquisition settings, which would require the use of distinct <SpectrumAnalysisProtocol> elements to be encoded correctly. Many of the workflows using cleavable crosslinkers fall into categories (ii) and (iii) above.

The consequence of the use of the "combined spectra" type of input file format is that the different spectra cannot be associated with different <SpectrumAnalysisProtocol> elements. But doing so is necessary to correctly encode cases (ii) and (iii) above (to capture, for example, different acquisition settings for different fragmentation modes).

mzIdentML 1.3.0 now advises the use of the new CV term "identification based on multiple spectra" (MS:1003332) to encode identifications based on multiple spectra. This CV term goes inside

<SpectrumIdentificationItem> elements. The "combined spectra" type of input file format from version 1.2.0 has been retired and is not part of the 1.3.0 specification.

The values of the "identification based on multiple spectra" CV term (MS:1003332) take the form: [identifier string]:[P or C]. For example:

<cvParam accession="MS:1003332" cvRef="PSI-MS" value="1234:P" name="identification based on multiple spectra"/>

The letters 'P' and 'C' refer to 'parent' and 'child'. <SpectrumIdentificationItem> elements marked 'parent' cover the entire identification (in the case of crosslinking, the crosslinked peptide pair) and those marked 'child' only identify a constituent part of the whole identification (typically a single peptide in an MS3 scan). Identifications based on multiple spectra MUST have both 'P' and 'C' marked constituent spectra identifications for cases that include at least one child element, or neither 'P' nor 'C' in the case that there is no parent/child relationship. There is no limit on the number of constituent spectra that are marked either 'P' or 'C'.

The unique identifier string ("1234" in the above example) can associate <SpectrumIdentificationItem> elements across different <SpectrumIdentificationList> elements. These <SpectrumIdentificationList> elements can then be associated with different <SpectrumIdentificationProtocol> elements.

There may not be a parent-child relationship between the scan events supporting an identification since one scan may not represent a constituent part of the other. To describe such a sibling relationship between scans, the 'P' and 'C' can be omitted, as is the case in Figure 6.

The associated example file multiple_spectra_per_id_1.3.0_draft.mzid shows a common workflow for cleavable crosslinkers which uses two levels of MS2 (one using HCD as the fragmentation mode, one using ETD as the fragmentation mode) followed by MS3 scans. To encode this, three distinct <SpectrumIdentificationProtocol> elements are needed: one for HCD MS2 scans, one for ETD MS2 scans, and one for MS3 scans. Three distinct <SpectrumIdentification> elements associate each of the <SpectrumIdentificationProtocol> elements with different <SpectrumIdentificationList> elements. Figure 5 shows an excerpt from this example file to illustrate the use of the new CV terms.

Figure 6 shows this approach used in the context of encoding identifications of glycopeptides.

<AnalysisData>

<!-- for the ETD to be correctly encoded it needs to go into a separate list -→

<SpectrumIdentificationList id="sil_ETD">

<SpectrumIdentificationResult spectrumID="index=2" spectraData_ref="pk_id" id="SIR_2">

<SpectrumIdentificationItem chargeState="3" experimentalMassToCharge="100" peptide_ref="p1"

rank="1" passThreshold="false" id="ETD_SII_0">

<PeptideEvidenceRef peptideEvidence_ref="pepevid_p1"/>

<cvParam cvRef="PSI-MS" accession="MS:1002511"

name="crosslink spectrum identification item" value="1"/>

<!-- this flags it as part of the crosslinked identification '1234' -→

<cvParam accession="MS:1003332" cvRef="PSI-MS" value="**1234:P**"

name="identification based on multiple spectra"/>

        <cvParam cvRef="PSI-MS" accession="MS:1003336"

name="Posterior Error Probability from multiple spectra identification"

value="**1234:1E-08**"/>

</SpectrumIdentificationItem>

<SpectrumIdentificationItem chargeState="3" experimentalMassToCharge="100" peptide_ref="p2"

rank="1" passThreshold="false" id="ETD_SII_1">

<PeptideEvidenceRef peptideEvidence_ref="pepevid_p2"/>

<cvParam cvRef="PSI-MS" accession="MS:1002511"

name="crosslink spectrum identification item" value="1"/>

<!-- this flags it as part of the crosslinked identification '1234' -→

<cvParam accession="MS:1003332" cvRef="PSI-MS" value="**1234:P**"

name="identification based on multiple spectra"/>

      <cvParam cvRef="PSI-MS" accession="MS:1003336"

name="Posterior Error Probability from multiple spectra identification"

value="**1234:1E-08**"/>

</SpectrumIdentificationItem>

</SpectrumIdentificationResult>

</SpectrumIdentificationList>

<!-- as the MS3 may have different search params, e.g. mass tolerance, they need to go into a separate list -→

<SpectrumIdentificationList id="sil_MS3">

<!-- HCD MS3 match peptide 1 A-→

<SpectrumIdentificationResult spectrumID="index=3" spectraData_ref="pk_id" id="SIR_3">

<SpectrumIdentificationItem chargeState="3" experimentalMassToCharge="50"

```
peptide_ref="p1_a" rank="1" passThreshold="false" id="MS3_SII_0">

<PeptideEvidenceRef peptideEvidence_ref="pepevid_p1_a"/>

<!-- this flags it as part of the crosslinked identification '1234' -→

<cvParam accession="MS:1003332" cvRef="PSI-MS" value="1234:C"

name="identification based on multiple spectra"/>

        <cvParam cvRef="PSI-MS" accession="MS:1003336"

name="Posterior Error Probability from multiple spectra identification"

value="1234:1E-08"/>

</SpectrumIdentificationItem>

</SpectrumIdentificationResult>

<!-- HCD MS3 match peptide 2 T-→

<SpectrumIdentificationResult spectrumID="index=4" spectraData_ref="pk_id" id="SIR_4">

<SpectrumIdentificationItem chargeState="3" experimentalMassToCharge="50"

peptide_ref="p2_t" rank="1" passThreshold="false" id="MS3_SII_1">

<PeptideEvidenceRef peptideEvidence_ref="pepevid_p2_t"/>

<!-- this flags it as part of the crosslinked identification '1234' -→

<cvParam accession="MS:1003332" cvRef="PSI-MS" value="1234:C"

name="identification based on multiple spectra"/>

        <cvParam cvRef="PSI-MS" accession="MS:1003336"

name="Posterior Error Probability from multiple spectra identification"

value="1234:1E-08"/>

</SpectrumIdentificationItem>

</SpectrumIdentificationResult>

<!-- HCD MS3 match peptide 1 T-→

<SpectrumIdentificationResult spectrumID="index=3" spectraData_ref="pk_id" id="SIR_5">

<SpectrumIdentificationItem chargeState="3" experimentalMassToCharge="51"

peptide_ref="p1_t" rank="1" passThreshold="false" id="MS3_SII_2">
```

```xml
<PeptideEvidenceRef peptideEvidence_ref="pepevid_p1_t"/>
<!-- this flags it as part of the crosslinked identification '1234' →
<cvParam accession="MS:1003332" cvRef="PSI-MS" value="1234:C"
name="identification based on multiple spectra"/>
        <cvParam cvRef="PSI-MS" accession="MS:1003336"
name="Posterior Error Probability from multiple spectra identification"
value="1234:1E-08"/>
</SpectrumIdentificationItem>
</SpectrumIdentificationResult>
<!-- HCD MS3 match peptide 2 A-→
<SpectrumIdentificationResult spectrumID="index=4" spectraData_ref="pk_id" id="SIR_6">
<SpectrumIdentificationItem chargeState="3" experimentalMassToCharge="49"
peptide_ref="p2_a" rank="1" passThreshold="false" id="MS3_SII_3">
<PeptideEvidenceRef peptideEvidence_ref="pepevid_p2_a"/>
<!-- this flags it as part of the crosslinked identification '1234' -→
<cvParam accession="MS:1003332" cvRef="PSI-MS" value="1234:C"
name="identification based on multiple spectra"/>
        <cvParam cvRef="PSI-MS" accession="MS:1003336"
name="Posterior Error Probability from multiple spectra identification"
value="1234:1E-08"/>
</SpectrumIdentificationItem>
</SpectrumIdentificationResult>
</SpectrumIdentificationList>
</AnalysisData>
```

Figure 5. XML snippet showing the new CV terms "identification based on multiple spectra" (MS:1003332) and "Posterior Error Probability from multiple spectra identification" (MS:1003336) used in the context of crosslinking.

To encode peptide level scores for multiple spectra identifications different CV terms MUST be used. The values of these new terms take the form: [multiple spectra identification identifier]: [score].

The identifier before the colon MUST be an identifier used for an identification based on multiple spectra. For example:

<cvParam accession="MS:1003332" cvRef="PSI-MS" value="**1234:P**" name="identification based on multiple spectra"/>

<cvParam cvRef="PSI-MS" accession="MS:1003336"

name="Posterior Error Probability for an identification based on multiple spectra" value="**1234:1E-08**"/>

A new CV term for "Posterior Error Probability from multiple spectra identification" (MS:1003336) has also been introduced in this context. For search specific, match level scores for identifications based on multiple spectra, new CV terms must be created, these should be children of MS:1003334 ("Parent term for PSM-level scores for identifications based on multiple spectra").

<AnalysisData>

<!-- HCD Spectrum List -→

<SpectrumIdentificationList id="SII_LIST_4_1" >

<SpectrumIdentificationResult spectrumID="controllerType=0 controllerNumber=1 scan=3832"

spectraData_ref="SD_17022_recal_B210619_02_Lumos_ZC_CO_190_D2I_SDA-

WT12019_09_19_OPRmix_35trig_EThcD35.raw"

id="SIR_4">

<SpectrumIdentificationItem chargeState="3" experimentalMassToCharge="821.6863"

calculatedMassToCharge="821.68451"

peptide_ref="T_O-Glycosylation_EAQT_O-Glycosylation_T_O-

Glycosylation_PLAA_Hex_4_HexNAc_4" rank="1"

passThreshold="true" id="SII_4_1">

<PeptideEvidenceRef peptideEvidence_ref="T_O-Glycosylation_EAQT_O-Glycosylation_T_O-

Glycosylation_PLAA_Hex_4_HexNAc_4_pe1"/>

<cvParam cvRef="PSI-MS" accession="MS:1002520" value="T_O-Glycosylation_EAQT_O-

Glycosylation_T_O-Glycosylation_PLAA_Hex_4_HexNAc_4:256"

name="peptide group ID"/>

<cvParam cvRef="PSI-MS" accession="MS:1002354" value="0.005" name="PSM-level q-value"/>

<cvParam cvRef="PSI-MS" accession="MS:1003332" value="**256**"

name="identification based on multiple spectra" />

<cvParam cvRef="PSI-MS" accession="MS:1003336"

name="Posterior Error Probability from multiple spectra identification"

value="256:1E-08"/>

</SpectrumIdentificationItem>

</SpectrumIdentificationResult>

</SpectrumIdentificationList>

<!-- EThcD Spectrum List -→

<SpectrumIdentificationList id="SII_LIST_5_1" >

<SpectrumIdentificationResult spectrumID="controllerType=0 controllerNumber=1 scan=3836"

spectraData_ref="SD_17022_recal_B210619_02_Lumos_ZC_CO_190_D2I_SDA-

WT12019_09_19_OPRmix_35trig_EThcD35.raw"

id="SIR_5">

<SpectrumIdentificationItem chargeState="3" experimentalMassToCharge="821.6863"

calculatedMassToCharge="821.68451"

peptide_ref="T_O-Glycosylation_EAQT_O-Glycosylation_T_O-

Glycosylation_PLAA_Hex_4_HexNAc_4" rank="1"

passThreshold="true" id="SII_5_1">

<PeptideEvidenceRef

peptideEvidence_ref="T_Hex_1_HexNAc_1_EAQT_Hex_2_HexNAc_2_T_Hex_1_HexNAc_1_PLAA_pe1"
/>

<cvParam cvRef="PSI-MS" accession="MS:1002520"

value="T_O-Glycosylation_EAQT_O-Glycosylation_T_O-

Glycosylation_PLAA_Hex_4_HexNAc_4:256"

name="peptide group ID"/>

<cvParam cvRef="PSI-MS" accession="MS:1003147" value="1:0.97:1:true"

name="PTMProphet probability"/>

<cvParam cvRef="PSI-MS" accession="MS:1003147" value="2:0.83:5:true"

name="PTMProphet probability"/>

<cvParam cvRef="PSI-MS" accession="MS:1003147" value="3:0.89:6:true"

name="PTMProphet probability"/>

<cvParam cvRef="PSI-MS" accession="MS:1002354" value="0.001" name="PSM-level q-value"/>

<cvParam cvRef="PSI-MS" accession="MS:1003332"

name="identification based on multiple spectra" value="**256**" />

<cvParam cvRef="PSI-MS" accession="MS:1003336"

name="Posterior Error Probability from multiple spectra identification"

value="256:1E-08"/>

</SpectrumIdentificationItem>

</SpectrumIdentificationResult>

</SpectrumIdentificationList>

</AnalysisData>

Figure 6. XML snippet showing the new CV terms "identification based on multiple spectra" (MS:1003332) and "Posterior Error Probability from multiple spectra identification" (MS:1003336) used in the context of encoding glycopeptides, using HCD-EThCD spectrum pairs for glycan localization (taken from dataset PXD020077).

Linking SearchModification elements to Modification elements

mzIdentML version 1.3.0 introduces two new CV terms to link <SearchModification> elements and <Modification> elements - "search modification id" (MS:1003392) which goes inside <SearchModification> elements, and "search modification id ref" (MS:1003393) which goes inside <Modification> elements. This allows for more detailed information on modifications to be provided without redundant repetition of this information throughout the file. Making this link is optional but recommended where possible. In the case of open modification searches, such a link cannot be made.

The values of "search modification id" (MS:1003392) MUST be unique within the <SpectrumIdentificationProtocol> element.

However, if there are multiple <SpectrumIdentificationProtocol> elements within the file, then the values of "search modification id" SHOULD be identical for identical modifications across different <SpectrumIdentificationProtocol> elements. This is to avoid the duplication of <Peptide> elements with the same <Modification> but identified with a different <SpectrumIdentificationProtocol>).

Conclusions

This document contains the specifications for using the mzIdentML format to represent results from peptide and protein identification pipelines, in the context of a proteomics investigation. This specification, in conjunction with the XML Schema, mapping file and CV constitute a proposal for a standard from the Proteomics Standards Initiative.

Authors and Contributors

Colin W. Combe, University of Edinburgh

Lars Kolbowski, Technical University Berlin

Lutz Fischer, University of Edinburgh

Ville Koskinen, Matrix Science, London

Joshua Klein, Boston University

Alexander Leitner, ETH Zurich

Juan Antonio Vizcaíno, European Informatics Institute

Andy Jones, University of Liverpool

Juri Rappsilber, University of Edinburg & Technical University Berlin

The mzIdentML version 1.2.0 authors were as follows:

Juan Antonio Vizcaíno, European Bioinformatics Institute

Gerhard Mayer, Medizinisches Proteom-Center, Ruhr-Universität Bochum

Martin Eisenacher, Medizinisches Proteom-Center, Ruhr-Universität Bochum

Sean Seymour, SCIEX / Applied Biosystems

Terry Farrah, Institute for Systems Biology

Eric Deutsch, Institute for Systems Biology

Andrew R Jones, University of Liverpool

The mzIdentML version 1.1.0 authors were as follows:

David Creasy, Matrix Science

Florian Reisinger, European Bioinformatics Institute

Johannes Griss, European Bioinformatics Institute

Juan Antonio Vizcaíno, European Bioinformatics Institute

Matthew Chambers, Vanderbilt University Medical Center

Gerhard Mayer, Medizinisches Proteom-Center, Ruhr-Universität Bochum

Martin Eisenacher, Medizinisches Proteom-Center, Ruhr-Universität Bochum

Andrew Jones, University of Liverpool

The mzIdentML version 1.0 authors were as follows:

Angel Pizarro, Center for Bioinformatics, University of Pennsylvania

David Creasy, MatrixScience

Phil Jones, European Bioinformatics Institute

Andreas Bertsch, Eberhard Karls University Tübingen

Jenny Siepen, University of Manchester

Martin Eisenacher, Medizinisches Proteom-Center, Ruhr-Universität Bochum

Andrew Jones, University of Liverpool

In addition to the authors indicated above, the following people contributed to the model development, gave feedback or tested mzIdentML:

- Simon Hubbard, University of Manchester
- Julian Selley, University of Manchester
- Zsuzsanna Bencsath-Makkai, Biomedical Engineering, McGill University
- Randy Julian, IndigoBio
- Pierre-Alain Binz, GeneBio Geneva
- Alex Masselot, GeneBio Geneva
- Lennart Martens, European Bioinformatics Institute
- Henning Hermjakob, European Bioinformatics Institute
- Luisa Montecchi, European Bioinformatics Institute
- Richard Côté, European Bioinformatics Institute
- Marc Sturm, Eberhard Karls University, Tübingen
- Jim Shofstahl, Thermo Fisher
- David Horn, Agilent
- Jimmy Eng, Fred Hutchinson Cancer Research
- Brian Searle, Proteome Software
- Phillip Young, Waters
- Michael Kohl, Medizinisches Proteom-Center, Ruhr-Universität Bochum, Germany
- Christian Stephan, Medizinisches Proteom-Center, Ruhr-Universität Bochum, Germany
- Eugene Kapp, Ludwig Institute for Cancer Research
- Michael Coleman, Stowers Institute
- Julian Uszkoreit, Medizinisches Proteom-Center, Ruhr-Universität Bochum, Germany
- Oliver Kohlbacher, Eberhard Karls University, Tübingen

- Mathias Walzer, Eberhard Karls University, Tübingen
- David Ovelleiro, European Bioinformatics Institute
- Alberto Medina, ProteoRed Consortium, Spain
- Salvador Martínez-Bartolomé, ProteoRed Consortium, Spain
- Laurent Gatto, University of Cambridge
- Simon Perkins, University of Liverpool
- Harald Barsnes, University of Bergen
- Marc Vaudel, University of Bergen
- Yasset Perez-Riverol, European Bioinformatics Institute
- Tobias Ternent, European Bioinformatics Institute
- Eugen Netz, Max Planck Institute for Developmental Biology, Tübingen
- Robert J. Chalkley, University of California San Franscisco
- Fawaz Ghali, University of Liverpool
- Timo Sachsenberg, Eberhard Karls University, Tübingen
- Andrea Sinz, Martin Luther University Halle-Wittenberg

References

1. Martens, L., Chambers, M., Sturm, M., Kessner, D., Levander, F., Shofstahl, J., Tang, W. H., Römpp, A., Neumann, S., Pizarro, A. D., Montecchi-Palazzi, L., Tasman, N., Coleman, M., Reisinger, F., Souda, P., Hermjakob, H., Binz, P.-A., and Deutsch, E. W. (2011) mzML—a Community Standard for Mass Spectrometry Data. *Molecular & Cellular Proteomics* 10, R110.0001332. Walzer, M., Qi, D., Mayer, G., Uszkoreit, J., Eisenacher, M., Sachsenberg, T., Gonzalez-Galarza, F. F., Fan, J., Bessant, C., Deutsch, E. W., Reisinger, F., Vizcaino, J. A., Medina-Aunon, J. A., Albar, J. P., Kohlbacher, O., and Jones, A. R. (2013) The mzQuantML data standard for mass spectrometry-based quantitative studies in proteomics. *Molecular & cellular proteomics : MCP* 12, 2332-23403. Griss, J., Jones, A. R., Sachsenberg, T., Walzer, M., Gatto, L., Hartler, J., Thallinger, G. G., Salek, R. M., Steinbeck, C., Neuhauser, N., Cox, J., Neumann, S., Fan, J., Reisinger, F., Xu, Q. W., Del Toro, N., Perez-Riverol, Y., Ghali, F., Bandeira, N., Xenarios, I., Kohlbacher, O., Vizcaino, J. A., and Hermjakob, H. (2014) The mzTab data exchange format: communicating mass-spectrometry-based proteomics and metabolomics experimental results to a wider audience. *Molecular & cellular proteomics : MCP* 13, 2765-27754. Kerrien, S., Orchard, S., Montecchi-Palazzi, L., Aranda, B., Quinn, A., Vinod, N., Bader, G., Xenarios, I., Wojcik, J., Sherman, D., Tyers, M., Salama, J., Moore, S., Ceol, A., Chatr-aryamontri, A., Oesterheld, M., Stumpflen, V., Salwinski, L., Nerothin, J., Cerami, E., Cusick, M., Vidal, M., Gilson, M., Armstrong, J., Woollard, P., Hogue, C., Eisenberg, D., Cesareni, G., Apweiler, R., and Hermjakob, H. (2007) Broadening the horizon - level 2.5 of the HUPO-PSI format for molecular interactions. *BMC Biology* 5, 445. Mayer, G., Montecchi-Palazzi, L., Ovelleiro, D., Jones, A. R., Binz, P.-A., Deutsch, E. W., Chambers, M., Kallhardt, M., Levander, F., Shofstahl, J., Orchard, S., Antonio Vizcaíno, J., Hermjakob, H., Stephan, C., Meyer, H. E., and Eisenacher, M. (2013) The HUPO proteomics standards initiative- mass spectrometry controlled vocabulary. *Database* 20136. Seymour, S. L., Farrah, T., Binz, P. A., Chalkley, R. J., Cottrell, J. S., Searle, B. C., Tabb, D. L., Vizcaino, J. A., Prieto, G., Uszkoreit, J., Eisenacher, M., Martinez-Bartolome, S., Ghali, F., and Jones, A. R. (2014) A standardized framing for reporting protein identifications in mzIdentML 1.2. *Proteomics* 14, 2389-23997. Li, H., Handsaker, B., Wysoker, A., Fennell, T., Ruan, J., Homer, N., Marth, G., Abecasis, G., and Durbin, R. (2009) The Sequence Alignment/Map format and SAMtools. *Bioinformatics (Oxford, England)* 25, 2078-2079

Intellectual Property Statement

The PSI takes no position regarding the validity or scope of any intellectual property or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; neither does it represent that it has made any effort to identify any such rights. Copies of claims of rights made available for publication and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this specification can be obtained from the PSI Chair.

The PSI invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights that may cover technology that may be required to practice this recommendation. Please address the information to the PSI Chair (see contacts information at PSI website).

Copyright Notice