**mzIdentML: exchange format for peptides and proteins identified from mass spectra**

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 1.2.1, draft April 2016.

# 

# 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.

**Contents**

[Abstract 1](#_Toc449341716)

[1. Introduction 3](#_Toc449341717)

[1.1 Background 3](#_Toc449341718)

[1.2 Document Structure 4](#_Toc449341719)

[2. Use Cases for mzIdentML 4](#_Toc449341720)

[3. Concepts and Terminology 5](#_Toc449341721)

[4. Relationship to Other Specifications 5](#_Toc449341722)

[4.1 The PSI Mass Spectrometry Controlled Vocabulary (CV) 6](#_Toc449341723)

[4.2 Validation of controlled vocabulary terms 6](#_Toc449341724)

[4.3 Changes from version 1.1.0 to version 1.2.0 6](#_Toc449341725)

[5. Format scope and specific use cases 7](#_Toc449341726)

[5.1.1 Handling updates to the controlled vocabulary 7](#_Toc449341727)

[5.1.2 Use of identifiers for input spectra to a search 7](#_Toc449341728)

[5.2 Comments on Specific Use Cases 9](#_Toc449341729)

[5.2.1 Protein grouping encoding 9](#_Toc449341730)

[5.2.2 Support for *de novo* sequencing approaches 12](#_Toc449341731)

[5.2.3 Spectral library searches 12](#_Toc449341732)

[5.2.4 Multiple database search engines 12](#_Toc449341733)

[5.2.5 Pre-fractionation of samples prior to MS and splitting of searches 13](#_Toc449341734)

[5.2.6 Peptide-level scores and statistical measures 13](#_Toc449341735)

[5.2.7 Encoding modification localisation scores 16](#_Toc449341736)

[5.2.8 Encoding results of crosslinking searches 19](#_Toc449341737)

[5.2.9 Encoding Proteogenomics annotation data 22](#_Toc449341738)

[5.3 Other supporting materials 22](#_Toc449341739)

[6. Model in XML Schema 23](#_Toc449341740)

[6.1 Element <MzIdentML> 23](#_Toc449341741)

[6.2 Element <AdditionalSearchParams> 25](#_Toc449341742)

[6.3 Element <Affiliation> 26](#_Toc449341743)

[6.4 Element <AmbiguousResidue> 26](#_Toc449341744)

[6.5 Element <AnalysisCollection> 27](#_Toc449341745)

[6.6 Element <AnalysisData> 27](#_Toc449341746)

[6.7 Element <AnalysisParams> 28](#_Toc449341747)

[6.8 Element <AnalysisProtocolCollection> 28](#_Toc449341748)

[6.9 Element <AnalysisSampleCollection> 29](#_Toc449341749)

[6.10 Element <AnalysisSoftware> 29](#_Toc449341750)

[6.11 Element <AnalysisSoftwareList> 29](#_Toc449341751)

[6.12 Element <AuditCollection> 30](#_Toc449341752)

[6.13 Element <BibliographicReference> 30](#_Toc449341753)

[6.14 Element <ContactRole> 31](#_Toc449341754)

[6.15 Element <Customizations> 31](#_Toc449341755)

[6.16 Element <DBSequence> 31](#_Toc449341756)

[6.17 Element <DataCollection> 32](#_Toc449341757)

[6.18 Element <DatabaseFilters> 33](#_Toc449341758)

[6.19 Element <DatabaseName> 34](#_Toc449341759)

[6.20 Element <DatabaseTranslation> 34](#_Toc449341760)

[6.21 Element <Enzyme> 35](#_Toc449341761)

[6.22 Element <EnzymeName> 35](#_Toc449341762)

[6.23 Element <Enzymes> 36](#_Toc449341763)

[6.24 Element <Exclude> 36](#_Toc449341764)

[6.25 Element <ExternalFormatDocumentation> 37](#_Toc449341765)

[6.26 Element <FileFormat> 37](#_Toc449341766)

[6.27 Element <Filter> 38](#_Toc449341767)

[6.28 Element <FilterType> 38](#_Toc449341768)

[6.29 Element <FragmentArray> 38](#_Toc449341769)

[6.30 Element <FragmentTolerance> 39](#_Toc449341770)

[6.31 Element <Fragmentation> 39](#_Toc449341771)

[6.32 Element <FragmentationTable> 39](#_Toc449341772)

[6.33 Element <Include> 40](#_Toc449341773)

[6.34 Element <InputSpectra> 40](#_Toc449341774)

[6.35 Element <InputSpectrumIdentifications> 40](#_Toc449341775)

[6.36 Element <Inputs> 41](#_Toc449341776)

[6.37 Element <IonType> 41](#_Toc449341777)

[6.38 Element <MassTable> 42](#_Toc449341778)

[6.39 Element <Measure> 42](#_Toc449341779)

[6.40 Element <Modification> 43](#_Toc449341780)

[6.41 Element <ModificationParams> 44](#_Toc449341781)

[6.42 Element <Organization> 44](#_Toc449341782)

[6.43 Element <Parent> 44](#_Toc449341783)

[6.44 Element <ParentTolerance> 45](#_Toc449341784)

[6.45 Element <Peptide> 45](#_Toc449341785)

[6.46 Element <PeptideEvidence> 46](#_Toc449341786)

[6.47 Element <PeptideEvidenceRef> 47](#_Toc449341787)

[6.48 Element <PeptideHypothesis> 48](#_Toc449341788)

[6.49 Element <PeptideSequence> 48](#_Toc449341789)

[6.50 Element <Person> 48](#_Toc449341790)

[6.51 Element <ProteinAmbiguityGroup> 49](#_Toc449341791)

[6.52 Element <ProteinDetection> 49](#_Toc449341792)

[6.53 Element <ProteinDetectionHypothesis> 50](#_Toc449341793)

[6.54 Element <ProteinDetectionList> 51](#_Toc449341794)

[6.55 Element <ProteinDetectionProtocol> 51](#_Toc449341795)

[6.56 Element <Provider> 52](#_Toc449341796)

[6.57 Element <Residue> 53](#_Toc449341797)

[6.58 Element <Role> 53](#_Toc449341798)

[6.59 Element <Sample> 53](#_Toc449341799)

[6.60 Element <SearchDatabase> 54](#_Toc449341800)

[6.61 Element <SearchDatabaseRef> 55](#_Toc449341801)

[6.62 Element <SearchModification> 55](#_Toc449341802)

[6.63 Element <SearchType> 56](#_Toc449341803)

[6.64 Element <Seq> 56](#_Toc449341804)

[6.65 Element <SequenceCollection> 57](#_Toc449341805)

[6.66 Element <SiteRegexp> 58](#_Toc449341806)

[6.67 Element <SoftwareName> 59](#_Toc449341807)

[6.68 Element <SourceFile> 59](#_Toc449341808)

[6.69 Element <SpecificityRules> 60](#_Toc449341809)

[6.70 Element <SpectraData> 60](#_Toc449341810)

[6.71 Element <SpectrumIDFormat> 61](#_Toc449341811)

[6.72 Element <SpectrumIdentification> 61](#_Toc449341812)

[6.73 Element <SpectrumIdentificationItem> 62](#_Toc449341813)

[6.74 Element <SpectrumIdentificationItemRef> 64](#_Toc449341814)

[6.75 Element <SpectrumIdentificationList> 64](#_Toc449341815)

[6.76 Element <SpectrumIdentificationProtocol> 65](#_Toc449341816)

[6.77 Element <SpectrumIdentificationResult> 68](#_Toc449341817)

[6.78 Element <SubSample> 69](#_Toc449341818)

[6.79 Element <SubstitutionModification> 69](#_Toc449341819)

[6.80 Element <Threshold> 69](#_Toc449341820)

[6.81 Element <TranslationTable> 70](#_Toc449341821)

[6.82 Element <cv> 71](#_Toc449341822)

[6.83 Element <cvList> 71](#_Toc449341823)

[6.84 Element <cvParam> 72](#_Toc449341824)

[6.85 Element <userParam> 72](#_Toc449341825)

[7. Specific Comments on schema 73](#_Toc449341826)

[7.1 File extension and compression 73](#_Toc449341827)

[7.2 Referencing elements within the document 73](#_Toc449341828)

[7.3 Searches against nucleotide sequences 73](#_Toc449341829)

[7.4 Reporting peptide and protein identifications passing a significance threshold 74](#_Toc449341830)

[7.5 Using decoy databases to set different thresholds of false discovery rate 74](#_Toc449341831)

[7.6 Database Filter 75](#_Toc449341832)

[7.7 Types of parameters and values 75](#_Toc449341833)

[7.8 Reporting fragmentation ions 75](#_Toc449341834)

[7.8.1 Internal fragments and immonium ions 76](#_Toc449341835)

[7.9 Enzyme definition 77](#_Toc449341836)

[7.10 Unknown modifications 78](#_Toc449341837)

[8. Conclusions 78](#_Toc449341838)

[9. Authors and Contributors 78](#_Toc449341839)

[10. References 79](#_Toc449341840)

[11. Intellectual Property Statement 79](#_Toc449341841)

[Copyright Notice 80](#_Toc449341842)

# 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. Associated tutorial material is available as outlined in Section X.

## 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 6 relates to other specifications, both those that it extends and those that it is intended to complement. Section **Error! Reference source not found.** discusses the reasoning behind several design decisions taken. 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 6.1. 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.

1. 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. However, there is no support for aggregating evidence from multiple MS levels. 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.
2. 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.
3. 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.
4. 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.
5. 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.
6. 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..
7. It should be possible to search a file to retrieve all molecules that have a specified modification.
8. It should be possible to store the results of a search of spectra against other spectra - i.e. a spectral library search.
9. It should be possible to store the results of a top down search i.e. analysis of complete proteins.
10. 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.
11. 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.
12. 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.
13. It should be possible to store
14. 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.
15. It should be possible to store the results of cross-linking approaches, whereby two peptides cross-linked using chemical reagents or biologically occurring modifications have been identified (newly added in mzIdentML 1.2).
16. It should be possible to represent statistical values or scores associated with the the positions of modifications on a peptide chain (newly added in mzIdentML 1.2).
17. 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).

There should be limited support for:

1. Support for 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.
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. A minor exception to this rule is support for multiple search engine approaches, as described in Section 5.2.4.

# Concepts and Terminology

This document assumes familiarity with XML Schema notation ([www.w3.org/XML/Schema](http://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 [RFC2119].

# 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/miape/msi/>). The Minimum Information About a Proteomics Experiment: Mass spectrometry Informatics document defines a checklist of information that should be reported about such a study. It is expected that mzIdentML will be used to support MIAPE:MSI compliant submissions to public repositories.
2. *mzML* (<http://www.psidev.info/mzml>). mzML is the PSI standard for capturing mass spectra / peak lists resulting from mass spectrometry in proteomics. It is RECOMMENDED that mzIdentML should be used in conjunction with mzML, although it will be 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 mass spectrometry (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 {Griss, 2014 #925}. mzTab results can represent a subset of the 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).

## The PSI Mass Spectrometry Controlled Vocabulary (CV)

The PSI-MS controlled vocabulary is intended to provide terms for annotation of mzML, mzIdentML 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 mass spectrometry 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 have 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](mailto: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.

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

* Unit Ontology (<http://www.obofoundry.org/cgi-bin/detail.cgi?id=unit>)
* ChEBI (<http://www.ebi.ac.uk/chebi/>)
* PSI Protein modifications workgroup - http://psidev.cvs.sourceforge.net/viewvc/psidev/psi/mod/data/PSI-MOD.obo (DEPRECATED - it is RECOMMENDED to use Unimod wherever possible).
* For describing sample types, any suitable and stable ontologies MAY be used.

## Validation of controlled vocabulary terms

The correct usage of controlled vocabulary terms within mzIdentML is governed by the use of a mapping file which defines each XML location (XPath) where a <cvParam> instance can be used, and the allowed terms from the PSI-MS, or other, controlled vocabularies. 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 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 XXXX.

## 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), and searches employing multiple search engines (Section 5.2.4). In addition, minor changes have been made to the recommended encoding of information, such as retention time (Section **Error! Reference source not found.**) and for encoding approaches from *de novo* sequencing (Section 5.2.2). No changes have been made to the XML Schema or mapping file from version 1.1.0, although several open issues have been identified that may be resolved in future major version updates (Section **Error! Reference source not found.**).

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

|  |  |  |
| --- | --- | --- |
| **Term name** | **Accession** | **Reference to section within this document** |
| peptide-level scoring | MS:1002490 | 5.2.6 |
| modification localization scoring | MS:1002491 | 5.2.7 |
| consensus scoring | MS:1002492 | 5.2.4 |
| sample pre-fractionation | MS:1002493 | 5.2.5 |
| cross-linking search | MS:1002494 | 5.2.8 |
| de novo search | MS:1001010 | XXX |
| spectral library search | MS:1001031 | XXX |
| proteogenomics search | MS:1002635 | XXX |
| 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> in mzIdentML 1.2.

# 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.

### Use of identifiers for input spectra to a search

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>.

**Update in version 1.2:**

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.

|  |  |  |  |
| --- | --- | --- | --- |
| **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. |
| 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. |
| 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. |
| MS:100XXX | multiple peak list nativeID format, combined spectra | index=xsd:nonNegativeInteger, index=xsd:nonNegativeInteger, index=xsd:nonNegativeInteger… | Comma separated list of spectra that have been merged prior to searching or used together in some other manner intrinsic to the search. |
| Duplicating all terms by this mechanism above, to allow for comma separated lists of spectral IDs |  |  |  |

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

In mzIdentML, the spectrumID attribute should be constructed following the data type specification in Table . 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 specifications. In version 1.1.0, CV terms had been proposed for representing set relationships between different proteins within groups, but there was 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 (3).

1. As in mzIdentML version 1.1, a single protein accession that has been cited by software is captured in mzIdentML in <ProteinDetectionHypothesis> (PDH).
   1. 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).
   1. 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”.
   1. 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).
   1. 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.
   2. 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).
   1. 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.
   2. Developers of software packages MAY propose additional terms for describing group membership of PDHs, which will be incorporated into the CV.
   3. The associated value for these CV terms MAY be used to annotate which PDH(s) are the super/same-set of the annotated PDH.
   4. 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. |
| 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. |
| 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 | 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. |
| 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. |
| 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 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 |
| 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 this mandatory requriements makes it difficult to represent results from *de novo* sequencing approaches where PSMs may not necessarily have originated from a protein database search. As such, in mzIdentML 1.2 <PeptideEvidenceRef> has a cardinality of 0..many. In all cases of database search, export software MUST include all <PeptideEvidenceRef> elements for every PSM. *De novo* sequencing approaches are therefore supported, but only in a relatively straightforward manner, where complete peptide sequences are identified. There is no specification for representing partial peptide sequences or sequence tags.

### 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 peptide-spectrum matches. 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, 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 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 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 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>. 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).

### 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.

|  |  |
| --- | --- |
| **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. | The final 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 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. | The final file SHOULD contain *1* <SpectrumIdentificationList>, *1* <SpectrumIdentificationProtocol>, *1* <SpectrumIdentification> referencing *n* <InputSpectra> sub-elements, 1 <ProteinDetection>, 1 <ProteinDetectionList>. |

### 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 be encompass one of the following concepts:

* A peptide sequence with a given set of modifications – located a specified positions, identified from a single chart state.
* A peptide sequence with a given set of modifications – located a 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 specifications has been defined, using CV terms as described in Figure 1. Three CV terms have currently been added to the 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 to be present in mzIdentML 1.2 (Figure 1).

First of all, an additional CV term “peptide-level scoring” (MS:1002490) is included 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 in the file 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.

Also, 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.



Figure . The mechanism for encoding peptide-level statistics in mzIdentML 1.2

### Encoding modification localisation scores

A new addition to mzIdentML 1.2 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 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 (Figure 2).

To ensure that downstream software is aware that a file contains modification scores, a CV term is added to the *<SpectrumIdenticationProtocol>* - “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 (mandatory 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 PSM referencing the peptide (“modification rescored by false localization rate, MS:1002380; 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"/>



Figure . The specification in mzIdentML 1.2 for encoding modification localization scores, using cvParams.

### Encoding results of crosslinking searches

A new mechanism has been added in mzIdentML 1.2 to encode results from search engines that support identification of crosslinked peptides. This presents a new challenge for mzIdentML, since more than one peptide can be identified from the same spectrum – here termed the alpha and beta peptides, along with crosslinker modifications, linked across the two chains (donor and receiver crosslinkers). The encoding, as exemplified in Figure 3, enables scores or statistical measures to be assigned jointly to identification of the alpha-beta pair, or independently to the alpha and beta peptide. To fulfil these requirements in mzIdentML 1.2, we made the following adaptations.

First of all, the <SpectrumIdentificationProtocol> must contain the CV term “cross-linking search” (MS:1002494) as shown in Figure 3a and Table 1. To represent the cross-linkers, we have added a mechanism for relating two different <Peptide> elements together, using the CV terms “cross-link donor” (MS:1002509) and “cross-link receiver” (MS:1002510), where an identical value indicates that they are grouped together (Figure 3b). The <Modification> element has an attribute *monoisotopicMassDelta*, and by convention we expect that the crosslink donor contains the complete mass delta introduced by the crosslinking reagent, and that the crosslink receiver reports a mass shift delta of zero. A convention has been introduced: within a given <SpectrumIdentificationResult>, a pair of crosslinked peptides are reported as two instances of <SpectrumIdentificationItem> linked together by sharing the same value for the rank attribute, and through having a shared local unique identifier as the value for the CV term “cross-link spectrum identification item” (MS:1002511), as shown in Figure 3d. If the search engine has produced a single score for the crosslinked pair, both <SpectrumIdentificationItem> elements must carry the identical score (CV term and value, as in Figure 3e), but the two chains may also have additional, independent scores if needed.



Figure . The encoding in mzIdentML 1.2 for annotating results of searches including cross-linked peptides.

**Additional parts discussed in Gent 2016:**

* In some methodologies, a consensus spectrum or collection of spectra is interpreted together to arrive at an identification. In this case, we should add:
  + New terms to Table 2 for “multiple peak list nativeID format**, combined spectra**” (for all suitable file format types) allowing a comma-separated list of identifiers e.g. “index=1001, index=1007”
  + This approach MAY be used for cases where light and heavy paired spectra are intrinsically searched and processed together; it may also be used for cases where ETD and HCD or MS3 spectra are generated and interpreted together.
  + In the case of light and heavy crosslinkers, we would expect to see 4 SIIs (light-alpha, light-beta, heavy-alpha, heavy-beta) within one SpectrumIdentificationResult – all sharing the same “cross-linker identifier”. The (same) appropriate expMassToCharge values should be carried by the two light linked peptides and a different value for the two heavy linked peptides
* The cross-linking approach MAY also be combined with mod position re-scoring as needed, following that part of the specification
* The cross-linking approach MAY also be combined with peptide-level re-scoring (with special terms for cross-linked peptides), for approaches that do not follow the consensus spectrum approach outlined above
* At the protein-level:
  + Optional CV terms to be added for demonstrating that particular proteins are cross-linked in the following style “file unique ID: score” (see Figure 4)
  + Pairs of proteins assumed to be interacting are then encoded in different ProteinAmbiguityGroups, sharing the same ID and score values
  + If evidence exists that the same protein forms a homo-dimer, then the same protein accession can be used (via PDH) in different PAGs, sharing the same cross-linker identifier
* A new CV for cross-linking modifications/reagents will be maintained on the mzIdentML GitHub page, XL-CV, in CSV format, with the following specs:
  + Each entry (table row) will have a single unique ID – XL:00001 and so on
  + Provisional decision to have different reagents as different modifications, if there is general agreement on this principle.
  + Dead-end (monolink) modifications will be separated entries (Table rows)



Figure Encoding of protein interaction scores or statistics from cross-linking in mzIdentML.

### Encoding Proteogenomics annotation data

TO BE INSERTED ONCE AGREED

## Other supporting materials

The following example instance documents are available and between them cover all the use cases supported.

All example files can be downloaded manually from:

<http://code.google.com/p/psi-pi/source/browse/#svn%2Ftrunk%2Fexamples%2F1_1examples>

1. 55merge\_mascot\_full.mzid - example MS-MS search results including decoy matches from Mascot.
2. 55merge\_omssa.mzid - example MS-MS search results including decoy matches from OMSSA.
3. 55merge\_tandem.mzid - example MS-MS search results including decoy matches from X!Tandem.
4. MPC\_example.mzid – an example of PSMs from different search engines, assembled into proteins using a third-party algorithm; false-discovery estimation using decoy database.
5. Mascot\_NA\_example.mzid - an example of a search against an EST database with Mascot.
6. Mascot\_top\_down\_example.mzid - a single MS/MS spectra from an intact protein, searched with Mascot.
7. Sequest\_example\_ver1.1.mzid - a simple example derived from a “.out” file produced by SEQUEST.
8. mascot\_pmf\_example.mzid - example Peptide Mass Fingerprint search with Mascot.

# Model in XML Schema

An overview of the schema is presented in Figure 5. The following documentation is automatically generated from the XML Schema.



Figure . A diagrammatic overview of the mzIdentML schema.

## 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](#cvList) | 1 | 1 | The list of controlled vocabularies used in the file. | | [AnalysisSoftwareList](#AnalysisSoftwareList) | 0 | 1 | The software packages used to perform the analyses. | | [Provider](#Provider) | 0 | 1 | The Provider of the mzIdentML record in terms of the contact and software. | | [AuditCollection](#AuditCollection) | 0 | 1 | The complete set of Contacts (people and organisations) for this file. | | [AnalysisSampleCollection](#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](#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](#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](#AnalysisProtocolCollection) | 1 | 1 | The collection of protocols which include the parameters and settings of the performed analyses. | | [DataCollection](#DataCollection) | 1 | 1 | The collection of input and output data sets of the analyses. | | [BibliographicReference](#BibliographicReference) | 0 | unbounded | Any bibliographic references associated with the file | |
| **Graphical Context:** |  |
| **Example Context:** | <MzIdentML id="" version="1.1.0"  xsi:schemaLocation="http://psidev.info/psi/pi/mzIdentML/1.1 ../../schema/mzIdentML1.1.0.xsd"  xmlns="http://psidev.info/psi/pi/mzIdentML/1.1"  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" creationDate="2011-03-25T13:16:49">  <cvList>  <cv id="PSI-MS" fullName="Proteomics Standards Initiative Mass Spectrometry Vocabularies"  uri="http://psidev.cvs.sourceforge.net/viewvc/\*checkout\*/psidev/psi/psi-ms/mzML/controlledVocabulary/psi-ms.obo"  ...  </MzIdentML> |

## Element <AdditionalSearchParams>

|  |  |
| --- | --- |
| **Definition:** | The search parameters other than the modifications searched. |
| **Type:** | ParamListType |
| **Attributes:** | none |
| **Subelements:** | |  |  |  |  | | --- | --- | --- | --- | | **Subelement Name** | **minOccurs** | **maxOccurs** | **Definition** | | [cvParam](#cvParam) | 1 | unbounded | A single entry from an ontology or a controlled vocabulary. | | [userParam](#userParam) | 1 | unbounded | A single user-defined parameter. | |
| **Example Context:** | <AdditionalSearchParams>  <userParam name="Mascot Instrument Name" value="Default"/>  <cvParam accession="MS:1001211" name="parent mass type mono" cvRef="PSI-MS"/>  <cvParam accession="MS:1001108" name="param: a ion" cvRef="PSI-MS"/>  <cvParam accession="MS:1001146" name="param: a ion-NH3" cvRef="PSI-MS"/>  <cvParam accession="MS:1001118" name="param: b ion" cvRef="PSI-MS"/>  <cvParam accession="MS:1001149" name="param: b ion-NH3" cvRef="PSI-MS"/>  ...  </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  e.g.: MS:1001108 (param: a ion)  e.g.: MS:1001118 (param: b ion)  e.g.: MS:1001119 (param: c ion)  e.g.: MS:1001146 (param: a ion-NH3)  e.g.: MS:1001148 (param: a ion-H2O)  e.g.: MS:1001149 (param: b ion-NH3)  e.g.: MS:1001150 (param: b ion-H2O)  e.g.: MS:1001151 (param: y ion-NH3)  e.g.: MS:1001152 (param: y ion-H2O)  e.g.: MS:1001257 (param: v ion)  et al.  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) |
| **Example cvParams:** | <cvParam accession="MS:1001211" name="parent mass type mono" cvRef="PSI-MS"/>  <cvParam accession="MS:1001108" name="param: a ion" cvRef="PSI-MS"/>  <cvParam accession="MS:1001146" name="param: a ion-NH3" cvRef="PSI-MS"/>  <cvParam accession="MS:1001118" name="param: b ion" cvRef="PSI-MS"/>  <cvParam accession="MS:1001149" name="param: b ion-NH3" cvRef="PSI-MS"/>  <cvParam accession="MS:1001262" name="param: y ion" cvRef="PSI-MS"/>  <cvParam accession="MS:1001151" name="param: y ion-NH3" cvRef="PSI-MS"/>  <cvParam accession="MS:1001256" cvRef="PSI-MS" name="fragment mass type mono"/> |
| **Example userParams:** | <userParam name="Mascot Instrument Name" value="Default"/> |

## 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"/> |

## 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](#cvParam) | 1 | unbounded | A single entry from an ontology or a controlled vocabulary. | | [userParam](#userParam) | 1 | unbounded | A single user-defined parameter. | |
| **Example Context:** | <AmbiguousResidue code="X">  <cvParam accession="MS:1001360" name="alternate single letter codes" 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"/>  </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" name="alternate single letter codes" cvRef="PSI-MS" |

## 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](#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](#ProteinDetection) | 0 | 1 | An Analysis which assembles a set of peptides (e.g. from a spectra search analysis) to proteins. | |
| **Example Context:** | <AnalysisCollection>  <SpectrumIdentification id="SI" spectrumIdentificationProtocol\_ref="SIP"  spectrumIdentificationList\_ref="SIL\_1" activityDate="2011-03-24T11:37:37">  <InputSpectra spectraData\_ref="SD\_1"/>  <SearchDatabaseRef searchDatabase\_ref="SDB\_NeoProt\_tripledecoy"/>  </SpectrumIdentification>  <ProteinDetection id="PD\_1" proteinDetectionProtocol\_ref="PDP\_MascotParser\_1"  ...  </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](#SpectrumIdentificationList) | 1 | unbounded | Represents the set of all search results from SpectrumIdentification. | | [ProteinDetectionList](#ProteinDetectionList) | 0 | 1 | The protein list resulting from a protein detection process. | |
| **Example Context:** | <AnalysisData>  <SpectrumIdentificationList id="SII\_LIST\_1" xmlns="http://psidev.info/psi/pi/mzIdentML/1.1">  <FragmentationTable>  <Measure id="Measure\_MZ">  <cvParam accession="MS:1001225" cvRef="PSI-MS" unitCvRef="PSI-MS" unitName="m/z" unitAccession="MS:1000040" name="product ion m/z"/>  </Measure>  <Measure id="Measure\_Int">  ...  </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](#cvParam) | 1 | unbounded | A single entry from an ontology or a controlled vocabulary. | | [userParam](#userParam) | 1 | unbounded | A single user-defined parameter. | |
| **Example Context:** | <AnalysisParams>  <cvParam accession="MS:1001316" name="mascot:SigThreshold" cvRef="PSI-MS" value="0.05"/>  <cvParam accession="MS:1001317" name="mascot:MaxProteinHits" cvRef="PSI-MS" value="Auto"/>  <cvParam accession="MS:1001318" name="mascot:ProteinScoringMethod" cvRef="PSI-MS"  value="MudPIT"/>  <cvParam accession="MS:1001319" name="mascot:MinMSMSThreshold" cvRef="PSI-MS" value="0"/>  <cvParam accession="MS:1001320" name="mascot:ShowHomologousProteinsWithSamePeptides"  ...  </AnalysisParams> |
| **cvParam Mapping Rules:** | Path /MzIdentML/AnalysisProtocolCollection/ProteinDetectionProtocol/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 accession="MS:1001316" name="mascot:SigThreshold" cvRef="PSI-MS" value="0.05"/>  <cvParam accession="MS:1001317" name="mascot:MaxProteinHits" cvRef="PSI-MS" value="Auto"/>  <cvParam accession="MS:1001318" name="mascot:ProteinScoringMethod" cvRef="PSI-MS"  <cvParam accession="MS:1001319" name="mascot:MinMSMSThreshold" cvRef="PSI-MS" value="0"/>  <cvParam accession="MS:1001320" name="mascot:ShowHomologousProteinsWithSamePeptides"  <cvParam accession="MS:1001321" name="mascot:ShowHomologousProteinsWithSubsetOfPeptides"  <cvParam accession="MS:1001322" name="mascot:RequireBoldRed" cvRef="PSI-MS" value="0"/>  <cvParam accession="MS:1001323" name="mascot:UseUnigeneClustering" cvRef="PSI-MS"  <cvParam accession="MS:1001324" name="mascot:IncludeErrorTolerantMatches" cvRef="PSI-MS"  <cvParam accession="MS:1001325" name="mascot:ShowDecoyMatches" cvRef="PSI-MS" value="0"/> |

## 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](#SpectrumIdentificationProtocol) | 1 | unbounded | The parameters and settings of a SpectrumIdentification analysis. | | [ProteinDetectionProtocol](#ProteinDetectionProtocol) | 0 | 1 | The parameters and settings of a ProteinDetection process. | |
| **Example Context:** | <AnalysisProtocolCollection xmlns="http://psidev.info/psi/pi/mzIdentML/1.1">  <SpectrumIdentificationProtocol analysisSoftware\_ref="ID\_software" id="SearchProtocol\_1">  <SearchType>  <cvParam accession="MS:1001083" cvRef="PSI-MS" 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](#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](#ContactRole) | 0 | 1 | The Contact that provided the document instance. | | [SoftwareName](#SoftwareName) | 1 | 1 | The name of the analysis software package, sourced from a CV if available. | | [Customizations](#Customizations) | 0 | 1 | Any customizations to the software, such as alternative scoring mechanisms implemented, should be documented here as free text. | |
| **Example Context:** | <AnalysisSoftware id="AS\_mascot\_server" name="Mascot Server" version="2.3.02"  uri="http://www.matrixscience.com/search\_form\_select.html">  <ContactRole contact\_ref="ORG\_MSL">  <Role>  <cvParam accession="MS:1001267" name="software vendor" cvRef="PSI-MS"/>  </Role>  </ContactRole>  ...  </AnalysisSoftware> |

## Element <AnalysisSoftwareList>

|  |  |
| --- | --- |
| **Definition:** | The software packages used to perform the analyses. |
| **Type:** | AnalysisSoftwareListType |
| **Attributes:** | none |
| **Subelements:** | |  |  |  |  | | --- | --- | --- | --- | | **Subelement Name** | **minOccurs** | **maxOccurs** | **Definition** | | [AnalysisSoftware](#AnalysisSoftware) | 1 | unbounded | The software used for performing the analyses. | |
| **Example Context:** | <AnalysisSoftwareList>  <AnalysisSoftware id="AS\_mascot\_server" name="Mascot Server" version="2.3.02"  uri="http://www.matrixscience.com/search\_form\_select.html">  <ContactRole contact\_ref="ORG\_MSL">  <Role>  <cvParam accession="MS:1001267" name="software vendor" cvRef="PSI-MS"/>  </Role>  ...  </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](#Person) | 1 | 1 | 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](#Organization) | 1 | 1 | 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/1.1">  <Person firstName="firstname" lastName="secondName" id="PERSON\_DOC\_OWNER">  <Affiliation organization\_ref="ORG\_DOC\_OWNER"/>  </Person>  <Organization name="myworkplace" id="ORG\_DOC\_OWNER"/>  </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  authors="David N. Perkins, Darryl J. C. Pappin, David M. Creasy, John S. Cottrell" editor=""  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"  issue="18" pages="3551-3567" publication="Electrophoresis" volume="20" year="1999"  publisher="Wiley VCH"  title="Probability-based protein identification by searching sequence databases using mass spectrometry data"  ...  </BibliographicReference> |

## Element <ContactRole>

|  |  |
| --- | --- |
| **Definition:** | 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](#Role) | 1 | 1 | The roles (lab equipment sales, contractor, etc.) the Contact fills. | |
| **Example Context:** | <ContactRole contact\_ref="PERSON\_DOC\_OWNER">  <Role>  <cvParam accession="MS:1001271" name="researcher" cvRef="PSI-MS"/>  </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 <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](#Seq) | 0 | 1 | The actual sequence of amino acids or nucleic acid. | | [cvParam](#cvParam) | 0 | unbounded | A choice of either a cvParam or userParam. | | [userParam](#userParam) | 0 | unbounded | A choice of either a cvParam or userParam. | |
| **Example Context:** | <DBSequence id="DBSeq\_1\_Rnd2psu|NC\_LIV\_080090" length="16207"  searchDatabase\_ref="SDB\_NeoProt\_tripledecoy" accession="Rnd2psu|NC\_LIV\_080090">  <Seq>DPARMTSRLSEAAPAEESMSAEATFHWYSPMLEYARAAPLWLPSRMLIEHAKPGKREGDSGHLESEATAGPPSPPAPPPEATSLPSYRGLLAFS  QASPPLCPVQLYPPRRRAHFELPLVSSDESQESRCLATCLRGVGLSWDYISPAGTSLVVAEPHGFSGPDLIQGPSADTARAELVPFFSAWSLEERVQSVSW  ...  AVIRTHQADALVHEDSRTALGWLASIYXGRSPSVGSDVSDSKFPKFAMKNSTRKKLKGDDSAITSAYVASAGGSSMGILSG</Seq>  <cvParam accession="MS:1001088" name="protein description" cvRef="PSI-MS"  value="Rnd2psu|NC\_LIV\_080090 Decoy sequence, was | organism=Neospora\_caninum | product=hypothetical protein | location=Neo\_chrVIIa:229175-282694(-) | length=16207"  />  </DBSequence> |
| **Example cvParams:** | <cvParam accession="MS:1001088" cvRef="PSI-MS" value="Rnd3psu|NC\_LIV\_083320 Rnd3psu|NC\_LIV\_083320 Decoy sequence, was | organism=Neospora\_caninum | product=zinc finger (CCCH type) protein, putative | location=Neo\_chrVIIa:3989308-3992771(+) | length=661" name="protein description"/> |

## 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](#Inputs) | 1 | 1 | The inputs to the analyses including the databases searched, the spectral data and the source file converted to mzIdentML. | | [AnalysisData](#AnalysisData) | 1 | 1 | Data sets generated by the analyses, including peptide and protein lists. | |
| **Graphical Context:** |  |
| **Example Context:** | <DataCollection>  <Inputs xmlns="http://psidev.info/psi/pi/mzIdentML/1.1">  <SourceFile location="build/classes/resources/55merge\_omssa.omx" id="SourceFile\_1">  <FileFormat>  <cvParam accession="MS:1001400" cvRef="PSI-MS" name="OMSSA xml file"/>  </FileFormat>  </SourceFile>  ...  </DataCollection> |

## Element <DatabaseFilters>

|  |  |
| --- | --- |
| **Definition:** | The specification of filters applied to the database searched. |
| **Type:** | DatabaseFiltersType |
| **Attributes:** | none |
| **Subelements:** | |  |  |  |  | | --- | --- | --- | --- | | **Subelement Name** | **minOccurs** | **maxOccurs** | **Definition** | | [Filter](#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" name="DB filter taxonomy" cvRef="PSI-MS"/>  </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](#cvParam) | 1 | 1 | A single entry from an ontology or a controlled vocabulary. | | [userParam](#userParam) | 1 | 1 | A single user-defined parameter. | |
| **Example Context:** | <DatabaseName>  <userParam name=" Neo\_rndTryp\_3times.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 SwissProt)  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) |
| **Example cvParams:** | <cvParam accession="MS:1001104" name="database UniProtKB/Swiss-Prot" cvRef="PSI-MS"/> |
| **Example userParams:** | <userParam name="Neo\_rndTryp\_3times.fasta"/>  <userParam name="D:/Software/Databases/Neospora\_3rndTryp/Neo\_rndTryp\_3times.fasta"/> |

## 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](#TranslationTable) | 1 | unbounded | The table used to translate codons into nucleic acids e.g. by reference to the NCBI translation table. | |
| **Example Context:** |  |

## 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 | 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 | 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](#SiteRegexp) | 0 | 1 | Regular expression for specifying the enzyme cleavage site. | | [EnzymeName](#EnzymeName) | 0 | 1 | The name of the enzyme from a CV. | |
| **Example Context:** | <Enzyme id="ENZ\_0" cTermGain="OH" nTermGain="H" semiSpecific="0">  <SiteRegexp><![CDATA[(?<=[KR])(?!P)]]></SiteRegexp>  <EnzymeName>  <cvParam accession="MS:1001251" name="Trypsin" cvRef="PSI-MS"/>  </EnzymeName>  </Enzyme> |

## Element <EnzymeName>

|  |  |
| --- | --- |
| **Definition:** | The name of the enzyme from a CV. |
| **Type:** | ParamListType |
| **Attributes:** | None |
| **Subelements:** | |  |  |  |  | | --- | --- | --- | --- | | **Subelement Name** | **minOccurs** | **maxOccurs** | **Definition** | | [cvParam](#cvParam) | 1 | unbounded | A single entry from an ontology or a controlled vocabulary. | | [userParam](#userParam) | 1 | unbounded | A single user-defined parameter. | |
| **Example Context:** | <EnzymeName>  <cvParam accession="MS:1001251" cvRef="PSI-MS" 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 accession="MS:1001251" name="Trypsin" cvRef="PSI-MS"/> |

## 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](#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 id="ENZ\_0" cTermGain="OH" nTermGain="H" semiSpecific="0">  <SiteRegexp><![CDATA[(?<=[KR])(?!P)]]></SiteRegexp>  <EnzymeName>  <cvParam accession="MS:1001251" name="Trypsin" cvRef="PSI-MS"/>  </EnzymeName>  </Enzyme>  ...  </Enzymes> |

## Element <Exclude>

|  |  |
| --- | --- |
| **Definition:** | All sequences fulfilling the specifed criteria are excluded. |
| **Type:** | ParamListType |
| **Attributes:** | none |
| **Subelements:** | |  |  |  |  | | --- | --- | --- | --- | | **Subelement Name** | **minOccurs** | **maxOccurs** | **Definition** | | [cvParam](#cvParam) | 1 | unbounded | A single entry from an ontology or a controlled vocabulary. | | [userParam](#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 |
| **Attributes:** | none |
| **Subelements:** | |  |  |  |  | | --- | --- | --- | --- | | **Subelement Name** | **minOccurs** | **maxOccurs** | **Definition** | | [cvParam](#cvParam) | 1 | 1 | A single entry from an ontology or a controlled vocabulary. | |
| **Example Context:** | <FileFormat>  <cvParam accession="MS:1001062" cvRef="PSI-MS" name="Mascot MGF file"/>  </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)  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 file)  e.g.: MS:1001107 (data stored in database)  e.g.: MS:1001199 (Mascot DAT file)  e.g.: MS:1001200 (Sequest out file)  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 file)  e.g.: MS:1001400 (OMSSA xml file)  et al.  Path /MzIdentML/DataCollection/Inputs/SpectraData/FileFormat  MUST supply a \*child\* term of MS:1000560 (mass spectrometer file format) one or more times  e.g.: MS:1000526 (Waters raw file)  e.g.: MS:1000562 (ABI WIFF file)  e.g.: MS:1000563 (Thermo RAW file)  e.g.: MS:1000564 (PSI mzData file)  e.g.: MS:1000565 (Micromass PKL file)  e.g.: MS:1000566 (ISB mzXML file)  e.g.: MS:1000567 (Bruker/Agilent YEP file)  e.g.: MS:1000584 (mzML file)  e.g.: MS:1000613 (DTA file)  e.g.: MS:1000614 (ProteinLynx Global Server mass spectrum XML file)  et al. |
| **Example cvParams:** | <cvParam accession="MS:1001199" name="Mascot DAT file" cvRef="PSI-MS"/>  <cvParam accession="MS:1001348" name="FASTA format" cvRef="PSI-MS"/>  <cvParam accession="MS:1001062" name="Mascot MGF file" cvRef="PSI-MS"/>  <cvParam accession="MS:1001400" cvRef="PSI-MS" name="OMSSA xml file"/> |

## 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](#FilterType) | 1 | 1 | The type of filter e.g. database taxonomy filter, pi filter, mw filter | | [Include](#Include) | 0 | 1 | All sequences fulfilling the specifed criteria are included. | | [Exclude](#Exclude) | 0 | 1 | All sequences fulfilling the specifed criteria are excluded. | |
| **Example Context:** | <Filter>  <FilterType>  <cvParam accession="MS:1001020" name="DB filter taxonomy" cvRef="PSI-MS"/>  </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](#cvParam) | 1 | 1 | A single entry from an ontology or a controlled vocabulary. | | [userParam](#userParam) | 1 | 1 | A single user-defined parameter. | |
| **Example Context:** | <FilterType>  <cvParam accession="MS:1001020" name="DB filter taxonomy" cvRef="PSI-MS"/>  </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" name="DB filter taxonomy" cvRef="PSI-MS"/> |

## 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:** |  |

## 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](#cvParam) | 1 | unbounded | A single entry from an ontology or a controlled vocabulary. | |
| **Example Context:** | <FragmentTolerance>  <cvParam accession="MS:1001412" name="search tolerance plus value" value="0.8"  cvRef="PSI-MS" unitAccession="UO:0000221" unitName="dalton" unitCvRef="UO"/>  <cvParam accession="MS:1001413" name="search tolerance minus value" value="0.8"  cvRef="PSI-MS" unitAccession="UO:0000221" unitName="dalton" unitCvRef="UO"/>  </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 accession="MS:1001412" name="search tolerance plus value" value="0.8"  <cvParam accession="MS:1001413" name="search tolerance minus value" value="0.8" |

## Element <Fragmentation>

|  |  |
| --- | --- |
| **Definition:** | The product ions identified in this result. |
| **Type:** | FragmentationType |
| **Attributes:** | none |
| **Subelements:** | |  |  |  |  | | --- | --- | --- | --- | | **Subelement Name** | **minOccurs** | **maxOccurs** | **Definition** | | [IonType](#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:** |  |

## 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](#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 accession="MS:1001225" cvRef="PSI-MS" unitCvRef="PSI-MS" unitName="m/z" unitAccession="MS:1000040" name="product ion m/z"/>  </Measure>  <Measure id="Measure\_Int">  <cvParam accession="MS:1001226" cvRef="PSI-MS" name="product ion intensity"/>  </Measure>  ...  </FragmentationTable> |

## Element <Include>

|  |  |
| --- | --- |
| **Definition:** | All sequences fulfilling the specifed criteria are included. |
| **Type:** | ParamListType |
| **Attributes:** | none |
| **Subelements:** | |  |  |  |  | | --- | --- | --- | --- | | **Subelement Name** | **minOccurs** | **maxOccurs** | **Definition** | | [cvParam](#cvParam) | 1 | unbounded | A single entry from an ontology or a controlled vocabulary. | | [userParam](#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 <InputSpectra>

|  |  |
| --- | --- |
| **Definition:** | One of the spectra data sets used. |
| **Type:** | InputSpectraType |
| **Attributes:** | |  |  |  |  | | --- | --- | --- | --- | | **Attribute Name** | **Data Type** | **Use** | **Definition** | | spectraData\_ref | xsd:string | optional | A reference to the SpectraData element which locates the input spectra to an external file. | |
| **Subelements:** | none |
| **Example Context:** | <InputSpectra spectraData\_ref="SID\_1"/> |

## 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="SIL\_1"/> |

## 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](#SourceFile) | 0 | unbounded | A file from which this mzIdentML instance was created. | | [SearchDatabase](#SearchDatabase) | 0 | unbounded | A database for searching mass spectra. Examples include a set of amino acid sequence entries, or annotated spectra libraries. | | [SpectraData](#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/1.1">  <SourceFile location="build/classes/resources/55merge\_omssa.omx" id="SourceFile\_1">  <FileFormat>  <cvParam accession="MS:1001400" cvRef="PSI-MS" name="OMSSA xml file"/>  </FileFormat>  </SourceFile>  <SearchDatabase numDatabaseSequences="22348" location="D:/Software/Databases/Neospora\_3rndTryp/Neo\_rndTryp\_3times.fasta" id="SearchDB\_1">  ...  </Inputs> |

## 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](#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. | | [cvParam](#cvParam) | 1 | 1 | A single entry from an ontology or a controlled vocabulary. | |
| **Example Context:** |  |
| **cvParam Mapping Rules:** | Path /MzIdentML/DataCollection/AnalysisData/SpectrumIdentificationList/SpectrumIdentificationResult/Spectrum  IdentificationItem/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:1001226 (product ion intensity)  e.g.: MS:1001227 (product ion m/z error)  et al. |

## 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](#Residue) | 0 | unbounded | The specification of a single residue within the mass table. | | [AmbiguousResidue](#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](#cvParam) | 0 | unbounded | A single entry from an ontology or a controlled vocabulary. | | [userParam](#userParam) | 0 | unbounded | A single user-defined parameter. | |
| **Example Context:** | <MassTable id="MT" msLevel="1 2">  <Residue code="A" mass="71.037114"/>  <Residue code="C" mass="103.009185"/>  <Residue code="D" mass="115.026943"/>  <Residue code="E" mass="129.042593"/>  <Residue code="F" mass="147.068414"/>  <Residue code="G" mass="57.021464"/>  ...  </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](#cvParam) | 1 | unbounded | A single entry from an ontology or a controlled vocabulary. | |
| **Example Context:** | <Measure id="Measure\_Error">  <cvParam accession="MS:1001227" cvRef="PSI-MS" unitCvRef="PSI-MS" unitName="m/z" unitAccession="MS:1000040" name="product ion m/z error"/>  </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 cvRef="PSI-MS" accession="MS:1001225" name="product ion m/z"/>  <cvParam cvRef="PSI-MS" accession="MS:1001226" name="product ion intensity"/>  <cvParam cvRef="PSI-MS" accession="MS:1001227" 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. |
| **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](#cvParam) | 1 | unbounded | A single entry from an ontology or a controlled vocabulary. | |
| **Example Context:** | <Modification location="11" residues="M" monoisotopicMassDelta="15.994915">  <cvParam accession="UNIMOD:35" name="Oxidation" cvRef="UNIMOD"/>  <cvParam accession="MS:1001524" name="fragment neutral loss" cvRef="PSI-MS"  value="63.998285" unitAccession="UO:0000221" unitName="dalton" unitCvRef="UO"/>  </Modification> |
| **Example cvParams:** | <cvParam accession="UNIMOD:4" name="Carbamidomethyl" cvRef="UNIMOD"/>  <cvParam accession="UNIMOD:35" name="Oxidation" cvRef="UNIMOD"/>  <cvParam accession="MS:1001524" name="fragment neutral loss" cvRef="PSI-MS" value="0" |

## 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](#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="C" massDelta="57.021465" fixedMod="true">  <cvParam accession="UNIMOD:4" cvRef="UNIMOD" name="Carbamidomethyl"/>  </SearchModification>  <SearchModification residues="M" massDelta="15.994915" fixedMod="false">  <cvParam accession="UNIMOD:35" cvRef="UNIMOD" name="Oxidation"/>  </SearchModification>  ...  </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](#cvParam) | 0 | unbounded | A choice of either a cvParam or userParam. | | [userParam](#userParam) | 0 | unbounded | A choice of either a cvParam or userParam. | | [Parent](#Parent) | 0 | 1 | The containing organization (the university or business which a lab belongs to, etc.) | |
| **Example Context:** | <Organization id="ORG\_MSL" name="Matrix Science Limited"/> |

## 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](#cvParam) | 1 | unbounded | A single entry from an ontology or a controlled vocabulary. | |
| **Example Context:** | <ParentTolerance>  <cvParam accession="MS:1001412" name="search tolerance plus value" value="1.5"  cvRef="PSI-MS" unitAccession="UO:0000221" unitName="dalton" unitCvRef="UO"/>  <cvParam accession="MS:1001413" name="search tolerance minus value" value="1.5"  cvRef="PSI-MS" unitAccession="UO:0000221" unitName="dalton" unitCvRef="UO"/>  </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 accession="MS:1001412" name="search tolerance plus value" value="1.5"  <cvParam accession="MS:1001413" name="search tolerance minus value" value="1.5" |

## 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](#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](#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](#SubstitutionModification) | 0 | unbounded | A modification where one residue is substituted by another (amino acid change). | | [cvParam](#cvParam) | 0 | unbounded | A choice of either a cvParam or userParam. | | [userParam](#userParam) | 0 | unbounded | A choice of either a cvParam or userParam. | |
| **Example Context:** | <Peptide id="TVDVGMGGVDLANLKACSGSGVSQELHIWGK\_000000100000000000000000000000000">  <PeptideSequence>TVDVGMGGVDLANLKACSGSGVSQELHIWGK</PeptideSequence>  <Modification location="6" residues="M" monoisotopicMassDelta="15.994915">  <cvParam accession="UNIMOD:35" name="Oxidation" cvRef="UNIMOD"/>  <cvParam accession="MS:1001524" name="fragment neutral loss" cvRef="PSI-MS"  value="63.998285" unitAccession="UO:0000221" unitName="dalton" unitCvRef="UO"/>  </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 | 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 | 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](#cvParam) | 0 | unbounded | A choice of either a cvParam or userParam. | | [userParam](#userParam) | 0 | unbounded | A choice of either a cvParam or userParam. | |
| **Example Context:** | <PeptideEvidence id="DDDHSDQGGEVQGGR\_00000000000000000\_1\_Rnd2psu|NC\_LIV\_123110\_2144\_2158"  start="2144" end="2158" pre="R" post="G" isDecoy="false"  dBSequence\_ref="DBSeq\_1\_Rnd2psu|NC\_LIV\_123110" peptide\_ref="DDDHSDQGGEVQGGR\_00000000000000000"/>  <PeptideEvidence id="SVAGKGLADEHTACR\_00000000000000000\_1\_Rnd3psu|NC\_LIV\_114310\_1396\_1410"  start="1396" end="1410" pre="R" post="E" isDecoy="false"  dBSequence\_ref="DBSeq\_1\_Rnd3psu|NC\_LIV\_114310" peptide\_ref="SVAGKGLADEHTACR\_00000000000000000"/>  <PeptideEvidence id="FASCCGEDDGEAPR\_0000000000000000\_1\_psu|NC\_LIV\_113540\_2484\_2497" start="2484"  ...  </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="SGALR\_0000000\_1\_Rnd1psu|NC\_LIV\_081600\_1089\_1093"/>  <PeptideEvidenceRef peptideEvidence\_ref="SGALR\_0000000\_1\_Rnd1psu|NC\_LIV\_020100\_977\_981"/>  <PeptideEvidenceRef peptideEvidence\_ref="SGALR\_0000000\_1\_Rnd1psu|NC\_LIV\_102910\_725\_729"/>  <PeptideEvidenceRef peptideEvidence\_ref="SGALR\_0000000\_1\_Rnd1psu|NC\_LIV\_122850\_402\_406"/>  <PeptideEvidenceRef peptideEvidence\_ref="SGALR\_0000000\_1\_Rnd1psu|NC\_LIV\_060960\_32\_36"/>  <PeptideEvidenceRef peptideEvidence\_ref="SGALR\_0000000\_1\_Rnd2psu|NC\_LIV\_145280\_820\_824"/>  ...  </PeptideEvidenceRef> |
| **Example cvParams:** | <cvParam accession="MS:1001171" name="mascot:score" cvRef="PSI-MS" value="13.49"/>  <cvParam accession="MS:1001172" name="mascot:expectation value" cvRef="PSI-MS"  <cvParam accession="MS:1001363" name="peptide unique to one protein" cvRef="PSI-MS"/>  <cvParam accession="MS:1001371" name="mascot:identity threshold" cvRef="PSI-MS" value="42"/>  <cvParam accession="MS:1001370" name="mascot:homology threshold" cvRef="PSI-MS" value="24"/>  <cvParam accession="MS:1001030" name="number of peptide seqs compared to each spectrum"  <cvParam accession="MS:1000796" name="spectrum title" cvRef="PSI-MS" |

## 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](#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="KDLYGNVVLSGGTTMYEGIGER\_000000000000000100000000\_1\_psu|NC\_LIV\_020800\_292\_313">  <SpectrumIdentificationItemRef spectrumIdentificationItem\_ref="SII\_303\_1"/>  <SpectrumIdentificationItemRef spectrumIdentificationItem\_ref="SII\_304\_1"/>  </PeptideHypothesis> |
| **Example cvParams:** | <cvParam accession="MS:1001171" name="mascot:score" cvRef="PSI-MS"  <cvParam accession="MS:1001093" name="sequence coverage" cvRef="PSI-MS" value="11"/>  <cvParam accession="MS:1001097" name="distinct peptide sequences" cvRef="PSI-MS" |

## 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>GELLGLGGVSGCPLRSGGTEAGGALEQPPLKPK</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](#cvParam) | 0 | unbounded | A choice of either a cvParam or userParam. | | [userParam](#userParam) | 0 | unbounded | A choice of either a cvParam or userParam. | | [Affiliation](#Affiliation) | 0 | unbounded | The organization a person belongs to. | |
| **Example Context:** | <Person firstName="firstname" lastName="secondName" id="PERSON\_DOC\_OWNER">  <Affiliation organization\_ref="ORG\_DOC\_OWNER"/>  </Person> |

## 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 |
| **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](#ProteinDetectionHypothesis) | 1 | unbounded | A single result of the ProteinDetection analysis (i.e. a protein). | | [cvParam](#cvParam) | 0 | unbounded | A single entry from an ontology or a controlled vocabulary. | | [userParam](#userParam) | 0 | unbounded | A single user-defined parameter. | |
| **Example Context:** | <ProteinAmbiguityGroup id="PAG\_hit\_2">  <ProteinDetectionHypothesis id="PDH\_psu|NC\_LIV\_105380\_0"  dBSequence\_ref="DBSeq\_1\_psu|NC\_LIV\_105380" passThreshold="true">  <PeptideHypothesis  peptideEvidence\_ref="VIDENFGLVEGLMTTVHAATGTQK\_00000000000001000000000000\_1\_psu|NC\_LIV\_105380\_842\_865">  <SpectrumIdentificationItemRef spectrumIdentificationItem\_ref="SII\_307\_1"/>  </PeptideHypothesis>  ...  </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](#InputSpectrumIdentifications) | 1 | unbounded | The lists of spectrum identifications that are input to the protein detection process. | |
| **Example Context:** | <ProteinDetection id="PD\_1" proteinDetectionProtocol\_ref="PDP\_MascotParser\_1"  proteinDetectionList\_ref="PDL\_1" activityDate="2011-03-25T13:33:51">  <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 | optional | A reference to the corresponding DBSequence entry. This optional and redundant, because the PeptideEvidence elements referenced from here also map to the DBSequence. | | 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](#PeptideHypothesis) | 1 | unbounded | Peptide evidence on which this ProteinHypothesis is based by reference to a PeptideEvidence element. | | [cvParam](#cvParam) | 0 | unbounded | A single entry from an ontology or a controlled vocabulary. | | [userParam](#userParam) | 0 | unbounded | A single user-defined parameter. | |
| **Example Context:** | <ProteinDetectionHypothesis id="PDH\_psu|NC\_LIV\_105380\_0"  dBSequence\_ref="DBSeq\_1\_psu|NC\_LIV\_105380" passThreshold="true">  <PeptideHypothesis  peptideEvidence\_ref="VIDENFGLVEGLMTTVHAATGTQK\_00000000000001000000000000\_1\_psu|NC\_LIV\_105380\_842\_865">  <SpectrumIdentificationItemRef spectrumIdentificationItem\_ref="SII\_307\_1"/>  </PeptideHypothesis>  <cvParam accession="MS:1001171" name="mascot:score" cvRef="PSI-MS" value="99.72"/>  ...  </ProteinDetectionHypothesis> |
| **cvParam Mapping Rules:** | Path /MzIdentML/DataCollection/AnalysisData/ProteinDetectionList/ProteinAmbiguityGroup/ProteinDetectionHypo  thesis  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)  e.g.: MS:1001162 (Sequest:total ions)  e.g.: MS:1001163 (Sequest:consensus score)  et al.  MAY supply a \*child\* term of MS:1001060 (quality estimation method details) one or more times  e.g.: MS:1001058 (quality estimation by manual validation)  e.g.: MS:1001194 (quality estimation with decoy database)  e.g.: MS:1001447 (prot:FDR threshold)  e.g.: MS:1001448 (pep:FDR threshold)  e.g.: MS:1001454 (quality estimation with implicite decoy sequences)  e.g.: MS:1001494 (no threshold)  e.g.: MS:1001574 (report only spectra assigned to identified proteins)  MAY supply a \*child\* term of MS:1001085 (protein result details) one or more times  e.g.: MS:1001088 (protein description)  e.g.: MS:1001090 (taxonomy nomenclature)  e.g.: MS:1001093 (sequence coverage)  e.g.: MS:1001097 (distinct peptide sequences)  e.g.: MS:1001098 (confident distinct peptide sequences)  e.g.: MS:1001099 (confident peptide qualification)  e.g.: MS:1001100 (confident peptide)  e.g.: MS:1001125 (manual validation)  e.g.: MS:1001157 (Sequest:sp)  e.g.: MS:1001158 (Sequest:Uniq)  et al. |

## 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](#ProteinAmbiguityGroup) | 0 | unbounded | A set of logically related results from a protein detection, for example to represent conflicting assignments of peptides to proteins. | | [cvParam](#cvParam) | 0 | unbounded | A single entry from an ontology or a controlled vocabulary. | | [userParam](#userParam) | 0 | unbounded | A single user-defined parameter. | |
| **Example Context:** | <ProteinDetectionList id="PDL\_1">  <ProteinAmbiguityGroup id="PAG\_hit\_1">  <ProteinDetectionHypothesis id="PDH\_psu|NC\_LIV\_020800\_0"  dBSequence\_ref="DBSeq\_1\_psu|NC\_LIV\_020800" passThreshold="true">  <PeptideHypothesis  peptideEvidence\_ref="LCYIALDFDEEMKAAEDSSDIEK\_0000000000001000000000000\_1\_psu|NC\_LIV\_020800\_217\_239">  <SpectrumIdentificationItemRef spectrumIdentificationItem\_ref="SII\_308\_1"/>  ...  </ProteinDetectionList> |
| **cvParam Mapping Rules:** | Path /MzIdentML/DataCollection/AnalysisData/ProteinDetectionList  MAY supply a \*child\* term of MS:1001184 (search statistics) one or more times  e.g.: MS:1001035 (date / time search performed)  e.g.: MS:1001036 (search time taken)  e.g.: MS:1001177 (number of molecular hypothesis considered) |

## 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](#AnalysisParams) | 0 | 1 | The parameters and settings for the protein detection given as CV terms. | | [Threshold](#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:** |  |
| **Example Context:** | <ProteinDetectionProtocol id="PDP\_MascotParser\_1" analysisSoftware\_ref="AS\_mascot\_parser">  <AnalysisParams>  <cvParam accession="MS:1001316" name="mascot:SigThreshold" cvRef="PSI-MS" value="0.05"/>  <cvParam accession="MS:1001317" name="mascot:MaxProteinHits" cvRef="PSI-MS" value="Auto"/>  <cvParam accession="MS:1001318" name="mascot:ProteinScoringMethod" cvRef="PSI-MS"  value="MudPIT"/>  <cvParam accession="MS:1001319" name="mascot:MinMSMSThreshold" cvRef="PSI-MS" value="0"/>  ...  </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](#ContactRole) | 0 | 1 | The Contact that provided the document instance. | |
| **Example Context:** | <Provider id="PROVIDER">  <ContactRole contact\_ref="PERSON\_DOC\_OWNER">  <Role>  <cvParam accession="MS:1001271" name="researcher" cvRef="PSI-MS"/>  </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="103.009185"/> |

## Element <Role>

|  |  |
| --- | --- |
| **Definition:** | The roles (lab equipment sales, contractor, etc.) the Contact fills. |
| **Type:** | RoleType |
| **Attributes:** | none |
| **Subelements:** | |  |  |  |  | | --- | --- | --- | --- | | **Subelement Name** | **minOccurs** | **maxOccurs** | **Definition** | | [cvParam](#cvParam) | 1 | 1 | A single entry from an ontology or a controlled vocabulary. | |
| **Example Context:** | <Role>  <cvParam accession="MS:1001267" name="software vendor" cvRef="PSI-MS"/>  </Role> |
| **Example cvParams:** | <cvParam accession="MS:1001267" name="software vendor" cvRef="PSI-MS"/>  <cvParam accession="MS:1001271" name="researcher" cvRef="PSI-MS"/> |

## 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](#ContactRole) | 0 | unbounded | The Contact that provided the document instance. | | [SubSample](#SubSample) | 0 | unbounded | References to the individual component samples within a mixed parent sample. | | [cvParam](#cvParam) | 0 | unbounded | A choice of either a cvParam or userParam. | | [userParam](#userParam) | 0 | unbounded | A choice of either a cvParam or userParam. | |
| **Example Context:** |  |

## Element <SearchDatabase>

|  |  |
| --- | --- |
| **Definition:** | A database for searching mass spectra. Examples include a set of amino acid sequence entries, or annotated spectra libraries. |
| **Type:** | SearchDatabaseType |
| **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](#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](#FileFormat) | 0 | 1 | The format of the ExternalData file, for example "tiff" for image files. | | [DatabaseName](#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](#cvParam) | 0 | unbounded | A single entry from an ontology or a controlled vocabulary. | |
| **Example Context:** | <SearchDatabase numDatabaseSequences="22348" location="D:/Software/Databases/Neospora\_3rndTryp/Neo\_rndTryp\_3times.fasta" id="SearchDB\_1">  <FileFormat>  <cvParam accession="MS:1001348" cvRef="PSI-MS" name="FASTA format"/>  </FileFormat>  <DatabaseName>  <userParam name="Neo\_rndTryp\_3times.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 | optional | A reference to the database searched. | |
| **Subelements:** | none |
| **Example Context:** | <SearchDatabaseRef searchDatabase\_ref="SDB\_NeoProt\_tripledecoy"/> |

## 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. |
| **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](#SpecificityRules) | 0 | 1 | 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](#cvParam) | 1 | unbounded | A single entry from an ontology or a controlled vocabulary. | |
| **Example Context:** | <SearchModification residues="C" massDelta="57.021465" fixedMod="true">  <cvParam accession="UNIMOD:4" cvRef="UNIMOD" name="Carbamidomethyl"/>  </SearchModification> |
| **cvParam Mapping Rules:** | Path /MzIdentML/AnalysisProtocolCollection/SpectrumIdentificationProtocol/ModificationParams/SearchModifi  cation  MUST supply a \*child\* term of UNIMOD:0 (UNIMOD root) one or more times  MUST 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)  MUST supply a \*child\* term of MOD:00000 (protein modification) one or more times |
| **Example cvParams:** | <cvParam accession="UNIMOD:4" name="Carbamidomethyl" cvRef="UNIMOD"/>  <cvParam accession="UNIMOD:35" name="Oxidation" 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](#cvParam) | 1 | 1 | A single entry from an ontology or a controlled vocabulary. | | [userParam](#userParam) | 1 | 1 | A single user-defined parameter. | |
| **Example Context:** | <SearchType>  <cvParam accession="MS:1001083" name="ms-ms search" cvRef="PSI-MS" value=""/>  </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) |
| **Example cvParams:** | <cvParam accession="MS:1001083" name="ms-ms search" cvRef="PSI-MS" value=""/> |

## Element <Seq>

|  |  |
| --- | --- |
| **Definition:** | The actual sequence of amino acids or nucleic acid. |
| **Type:** | sequence |
| **Attributes:** | none |
| **Subelements:** | none |
| **Example Context:** | <Seq>MPSRSNSRGASADPASDALSDADAASSAVPGSASERSFPFVHPSRDQLTADKPAKRDADQEAFAMTEDTLPPVPPAPPPGEEGVPSSRFTSSEAFH  DPPASPACASPPRRRCAAASPELEALGAFFARYACCLERVAVVDGAAECPGSLFGCALLPHVEASPAFAVSPAAWTSRWEADPFAWSGQGETRHGGALASR  ...  RVERRRPDHRRRAGDCGEKGPKRGARRGRKHGGARAPSRAGPETEPAAASPAASARQRLVQAMALPACAFLDLQAQQPSSFSVSPDGTPDPVYPQDLVSLD  ALRQGFPCGAPTGKSLGRIQDWSSTGATFWSRRVRAALDAFVLLPSWYGGIENRLLLEEAAVLLANTATCALLEAYAVHCLKRQAAAPIPRMYAAGHAAVS  DASLRSVQIAENQTVDDRLPTASTCFFMIKLPKYSSKEVLRKKLKLAIMSCVDIDLDALHHDLDFAQFE</Seq> |
| **Example cvParams:** | <cvParam accession="MS:1001088" name="protein description" cvRef="PSI-MS" |

## 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 |
| **Attributes:** | none |
| **Subelements:** | |  |  |  |  | | --- | --- | --- | --- | | **Subelement Name** | **minOccurs** | **maxOccurs** | **Definition** | | [DBSequence](#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](#Peptide) | 0 | unbounded | One (poly)peptide (a sequence with modifications). The combination of Peptide sequence and modifications MUST be unique in the file. | | [PeptideEvidence](#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:** |  |
| **Example Context:** | <SequenceCollection xmlns="http://psidev.info/psi/pi/mzIdentML/1.1">  <DBSequence accession="Rnd3psu|NC\_LIV\_083320" searchDatabase\_ref="SearchDB\_1" length="661" id="dbseq\_Rnd3psu|NC\_LIV\_083320">  <cvParam accession="MS:1001088" cvRef="PSI-MS" value="Rnd3psu|NC\_LIV\_083320 Rnd3psu|NC\_LIV\_083320 Decoy sequence, was | organism=Neospora\_caninum | product=zinc finger (CCCH type) protein, putative | location=Neo\_chrVIIa:3989308-3992771(+) | length=661" name="protein description"/>  </DBSequence>  <DBSequence accession="Rnd1psu|NC\_LIV\_123020" searchDatabase\_ref="SearchDB\_1" length="2986" id="dbseq\_Rnd1psu|NC\_LIV\_123020">  <cvParam accession="MS:1001088" cvRef="PSI-MS" value="Rnd1psu|NC\_LIV\_123020 Rnd1psu|NC\_LIV\_123020 Decoy sequence, was | organism=Neospora\_caninum | product=hypothetical protein | location=Neo\_chrX:3202583-3213218(-) | length=2986" name="protein description"/>  </DBSequence>  ...  </SequenceCollection> |

## Element <SiteRegexp>

|  |  |
| --- | --- |
| **Definition:** | Regular expression for specifying the enzyme cleavage site. |
| **Type:** | xsd:string |
| **Attributes:** | none |
| **Subelements:** | none |
| **Example Context:** | <SiteRegexp><![CDATA[(?<=[KR])(?!P)]]></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](#cvParam) | 1 | 1 | A choice of either a cvParam or userParam. | | [userParam](#userParam) | 1 | 1 | A choice of either a cvParam or userParam. | |
| **Example Context:** | <SoftwareName>  <cvParam accession="MS:1001478" name="Mascot Parser" cvRef="PSI-MS"/>  </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 accession="MS:1001207" name="Mascot" cvRef="PSI-MS"/>  <cvParam accession="MS:1001478" name="Mascot Parser" cvRef="PSI-MS"/>  <cvParam accession="MS:1001475" cvRef="PSI-MS" name="OMSSA"/> |

## Element <SourceFile>

|  |  |
| --- | --- |
| **Definition:** | A file from which this mzIdentML instance was created. |
| **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](#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](#FileFormat) | 0 | 1 | The format of the ExternalData file, for example "tiff" for image files. | | [cvParam](#cvParam) | 0 | unbounded | A single entry from an ontology or a controlled vocabulary. | | [userParam](#userParam) | 0 | unbounded | A single user-defined parameter. | |
| **Example Context:** | <SourceFile location="file:///D:/TestSpace/NeoTestMarch2011/55merge\_mascot.dat" id="SF\_1">  <FileFormat>  <cvParam accession="MS:1001199" name="Mascot DAT file" cvRef="PSI-MS"/>  </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](#cvParam) | 1 | unbounded | A single entry from an ontology or a controlled vocabulary. | |
| **Example Context:** |  |
| **cvParam Mapping Rules:** | Path /MzIdentML/AnalysisProtocolCollection/SpectrumIdentificationProtocol/ModificationParams/SearchModifi  cation/SpecificityRules  MUST supply a \*child\* term of MS:1001056 (modification specificity rule) one or more times  e.g.: MS:1001189 (modification specificity N-term)  e.g.: MS:1001190 (modification specificity C-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](#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](#FileFormat) | 0 | 1 | The format of the ExternalData file, for example "tiff" for image files. | | [SpectrumIDFormat](#SpectrumIDFormat) | 1 | 1 | The format of the spectrum identifier within the source file | |
| **Example Context:** | <SpectraData location="D:/TestSpace/NeoTestMarch2011/55merge.mgf" id="SID\_1">  <FileFormat>  <cvParam accession="MS:1001062" cvRef="PSI-MS" name="Mascot MGF file"/>  </FileFormat>  <SpectrumIDFormat>  <cvParam accession="MS:1000774" cvRef="PSI-MS" name="multiple peak list nativeID format"/>  </SpectrumIDFormat>  ...  </SpectraData> |

## Element <SpectrumIDFormat>

|  |  |
| --- | --- |
| **Definition:** | The format of the spectrum identifier within the source file |
| **Type:** | SpectrumIDFormatType |
| **Attributes:** | none |
| **Subelements:** | |  |  |  |  | | --- | --- | --- | --- | | **Subelement Name** | **minOccurs** | **maxOccurs** | **Definition** | | [cvParam](#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 nativeID format) |
| **Example cvParams:** | <cvParam accession="MS:1000774" name="multiple peak list nativeID format" cvRef="PSI-MS"/> |

## 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](#InputSpectra) | 1 | unbounded | One of the spectra data sets used. | | [SearchDatabaseRef](#SearchDatabaseRef) | 1 | unbounded | One of the search databases used. | |
| **Example Context:** | <SpectrumIdentification id="SI" spectrumIdentificationProtocol\_ref="SIP"  spectrumIdentificationList\_ref="SIL\_1" activityDate="2011-03-24T11:37:37">  <InputSpectra spectraData\_ref="SD\_1"/>  <SearchDatabaseRef searchDatabase\_ref="SDB\_NeoProt\_tripledecoy"/>  </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 | optional | 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](#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](#Fragmentation) | 0 | 1 | The product ions identified in this result. | | [cvParam](#cvParam) | 0 | unbounded | A single entry from an ontology or a controlled vocabulary. | | [userParam](#userParam) | 0 | unbounded | A single user-defined parameter. | |
| **Graphical Context:** |  |
| **Example Context:** | <SpectrumIdentificationItem id="SII\_69\_10" calculatedMassToCharge="515.329999"  chargeState="1" experimentalMassToCharge="514.242" peptide\_ref="GIGLR\_0000000" rank="10"  passThreshold="false">  <PeptideEvidenceRef peptideEvidence\_ref="GIGLR\_0000000\_1\_psu|NC\_LIV\_020600\_1026\_1030"/>  <PeptideEvidenceRef peptideEvidence\_ref="GIGLR\_0000000\_1\_Rnd1psu|NC\_LIV\_145070\_441\_445"/>  <PeptideEvidenceRef peptideEvidence\_ref="GIGLR\_0000000\_1\_Rnd1psu|NC\_LIV\_120130\_148\_152"/>  <PeptideEvidenceRef peptideEvidence\_ref="GIGLR\_0000000\_1\_Rnd3psu|NC\_LIV\_072540\_650\_654"/>  ...  </SpectrumIdentificationItem> |
| **cvParam Mapping Rules:** | Path /MzIdentML/DataCollection/AnalysisData/SpectrumIdentificationList/SpectrumIdentificationResult/Spectrum  IdentificationItem  MAY supply a \*child\* term of MS:1001405 (spectrum identification result details) one or more times  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)  e.g.: MS:1001088 (protein description)  et al. |
| **Example cvParams:** | <cvParam accession="MS:1001171" name="mascot:score" cvRef="PSI-MS" value="13.49"/>  <cvParam accession="MS:1001172" name="mascot:expectation value" cvRef="PSI-MS"  <cvParam accession="MS:1001363" name="peptide unique to one protein" cvRef="PSI-MS"/>  <cvParam accession="MS:1001371" name="mascot:identity threshold" cvRef="PSI-MS" value="43"/>  <cvParam accession="MS:1001370" name="mascot:homology threshold" cvRef="PSI-MS" value="26"/>  <cvParam accession="MS:1001030" name="number of peptide seqs compared to each spectrum"  <cvParam accession="MS:1000796" name="spectrum title" cvRef="PSI-MS"  <cvParam accession="MS:1001328" cvRef="PSI-MS" value="0.866331351956052" name="OMSSA:evalue"/>  <cvParam accession="MS:1001329" cvRef="PSI-MS" value="2.0805267818349E-4" name="OMSSA:pvalue"/> |

## 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\_308\_1"/> |

## 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](#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](#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](#cvParam) | 0 | unbounded | A single entry from an ontology or a controlled vocabulary. | | [userParam](#userParam) | 0 | unbounded | A single user-defined parameter. | |
| **Graphical Context:** |  |
| **Example Context:** | <SpectrumIdentificationList id="SII\_LIST\_1" xmlns="http://psidev.info/psi/pi/mzIdentML/1.1">  <FragmentationTable>  <Measure id="Measure\_MZ">  <cvParam accession="MS:1001225" cvRef="PSI-MS" unitCvRef="PSI-MS" unitName="m/z" unitAccession="MS:1000040" name="product ion m/z"/>  </Measure>  <Measure id="Measure\_Int">  <cvParam accession="MS:1001226" cvRef="PSI-MS" name="product ion intensity"/>  ...  </SpectrumIdentificationList> |
| **cvParam Mapping Rules:** | Path /MzIdentML/DataCollection/AnalysisData/SpectrumIdentificationList  MAY supply a \*child\* term of MS:1001184 (search statistics) one or more times  e.g.: MS:1001035 (date / time search performed)  e.g.: MS:1001036 (search time taken)  e.g.: MS:1001177 (number of molecular hypothesis considered) |

## 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](#SearchType) | 1 | 1 | The type of search performed e.g. PMF, Tag searches, MS-MS | | [AdditionalSearchParams](#AdditionalSearchParams) | 0 | 1 | The search parameters other than the modifications searched. | | [ModificationParams](#ModificationParams) | 0 | 1 | The specification of static/variable modifications (e.g. Oxidation of Methionine) that are to be considered in the spectra search. | | [Enzymes](#Enzymes) | 0 | 1 | The list of enzymes used in experiment | | [MassTable](#MassTable) | 0 | unbounded | The masses of residues used in the search. | | [FragmentTolerance](#FragmentTolerance) | 0 | 1 | The tolerance of the search given as a plus and minus value with units. | | [ParentTolerance](#ParentTolerance) | 0 | 1 | The tolerance of the search given as a plus and minus value with units. | | [Threshold](#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](#DatabaseFilters) | 0 | 1 | The specification of filters applied to the database searched. | | [DatabaseTranslation](#DatabaseTranslation) | 0 | 1 | A specification of how a nucleic acid sequence database was translated for searching. | |
| **Graphical Context:** |  |
| **Example Context:** | <SpectrumIdentificationProtocol analysisSoftware\_ref="ID\_software" id="SearchProtocol\_1">  <SearchType>  <cvParam accession="MS:1001083" cvRef="PSI-MS" name="ms-ms search"/>  </SearchType>  <AdditionalSearchParams>  <cvParam accession="MS:1001211" cvRef="PSI-MS" name="parent mass type mono"/>  <cvParam accession="MS:1001256" cvRef="PSI-MS" name="fragment mass type mono"/>  ...  </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](#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](#cvParam) | 0 | unbounded | A single entry from an ontology or a controlled vocabulary. | | [userParam](#userParam) | 0 | unbounded | A single user-defined parameter. | |
| **Example Context:** | <SpectrumIdentificationResult spectraData\_ref="SID\_1" spectrumID="index=21" id="SIR\_32">  <SpectrumIdentificationItem passThreshold="false" rank="1" peptide\_ref="VIDENFGLVEGLMTTVHAATGTQK\_1@12" calculatedMassToCharge="2546268.0" experimentalMassToCharge="2547212.0" chargeState="3" id="SII\_32\_1">  <PeptideEvidenceRef peptideEvidence\_ref="PE32\_2\_53"/>  <cvParam accession="MS:1001328" cvRef="PSI-MS" value="7.40729329987533E-8" name="OMSSA:evalue"/>  <cvParam accession="MS:1001329" cvRef="PSI-MS" value="3.18593260209692E-11" name="OMSSA:pvalue"/>  </SpectrumIdentificationItem>  <SpectrumIdentificationItem passThreshold="false" rank="2" peptide\_ref="APCSGSAVTGVDSPGCDGVGDLNVTR" calculatedMassToCharge="2547129.0" experimentalMassToCharge="2547212.0" chargeState="3" id="SII\_32\_2">  ...  </SpectrumIdentificationResult> |
| **cvParam Mapping Rules:** | Path /MzIdentML/DataCollection/AnalysisData/SpectrumIdentificationList/SpectrumIdentificationResult  MAY supply a \*child\* term of MS:1001405 (spectrum identification result details) one or more times  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)  e.g.: MS:1001088 (protein description)  et al. |

## 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 | xsd:string with restriction  [ABCDEFGHIJKLMNOPQRSTUVWXYZ?\-]{1} | required | The original residue before replacement. | | replacementResidue | xsd:string with restriction  [ABCDEFGHIJKLMNOPQRSTUVWXYZ?\-]{1} | required | The residue that replaced the originalResidue. | |
| **Subelements:** | none |
| **Example Context:** | <SubstitutionModification location="15" originalResidue="X" replacementResidue="P"/> |

## Element <Threshold>

|  |  |
| --- | --- |
| **Definition:** | 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. |
| **Type:** | ParamListType |
| **Attributes:** | none |
| **Subelements:** | |  |  |  |  | | --- | --- | --- | --- | | **Subelement Name** | **minOccurs** | **maxOccurs** | **Definition** | | [cvParam](#cvParam) | 1 | unbounded | A single entry from an ontology or a controlled vocabulary. | | [userParam](#userParam) | 1 | unbounded | A single user-defined parameter. | |
| **Example Context:** | <Threshold>  <cvParam accession="MS:1001316" name="mascot:SigThreshold" cvRef="PSI-MS" value="0.05"/>  </Threshold> |
| **cvParam Mapping Rules:** | Path /MzIdentML/AnalysisProtocolCollection/SpectrumIdentificationProtocol/Threshold  MUST 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.  MUST 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)  e.g.: MS:1001162 (Sequest:total ions)  e.g.: MS:1001163 (Sequest:consensus score)  et al.  MUST supply term MS:1001494 (no threshold) only once  MUST supply term MS:1001448 (pep:FDR threshold) only once  Path /MzIdentML/AnalysisProtocolCollection/ProteinDetectionProtocol/Threshold  MUST 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.  MUST 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)  e.g.: MS:1001162 (Sequest:total ions)  e.g.: MS:1001163 (Sequest:consensus score)  et al.  MUST supply term MS:1001447 (prot:FDR threshold) only once  MUST supply term MS:1001494 (no threshold) only once |
| **Example cvParams:** | <cvParam accession="MS:1001316" name="mascot:SigThreshold" cvRef="PSI-MS" value="0.05"/>  <cvParam accession="MS:1001494" name="no threshold" cvRef="PSI-MS"/> |

## 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](#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 <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 id="PSI-MS" fullName="Proteomics Standards Initiative Mass Spectrometry Vocabularies"  uri="http://psidev.cvs.sourceforge.net/viewvc/\*checkout\*/psidev/psi/psi-ms/mzML/controlledVocabulary/psi-ms.obo"  version="2.32.0"/>  <cv id="UNIMOD" fullName="UNIMOD" uri="http://www.unimod.org/obo/unimod.obo"/>  <cv id="UO" fullName="UNIT-ONTOLOGY"  uri="http://obo.cvs.sourceforge.net/\*checkout\*/obo/obo/ontology/phenotype/unit.obo"/>  </cvList>  ...  </cv> |

## Element <cvList>

|  |  |
| --- | --- |
| **Definition:** | The list of controlled vocabularies used in the file. |
| **Type:** | CVListType |
| **Attributes:** | none |
| **Subelements:** | |  |  |  |  | | --- | --- | --- | --- | | **Subelement Name** | **minOccurs** | **maxOccurs** | **Definition** | | [cv](#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="http://psidev.cvs.sourceforge.net/viewvc/\*checkout\*/psidev/psi/psi-ms/mzML/controlledVocabulary/psi-ms.obo"  version="2.32.0"/>  <cv id="UNIMOD" fullName="UNIMOD" uri="http://www.unimod.org/obo/unimod.obo"/>  <cv id="UO" fullName="UNIT-ONTOLOGY"  uri="http://obo.cvs.sourceforge.net/\*checkout\*/obo/obo/ontology/phenotype/unit.obo"/>  ...  </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 accession="MS:1001088" name="protein description" cvRef="PSI-MS"  value="Rnd3psu|NC\_LIV\_111720 Decoy sequence, was | organism=Neospora\_caninum | product=forkhead-aasociated (FHA) domain-containing protein | location=Neo\_chrIX:1702443-1710709(-) | length=1155"  />  </DBSequence>  <DBSequence id="DBSeq\_1\_Rnd1psu|NC\_LIV\_080090" length="16207"  searchDatabase\_ref="SDB\_NeoProt\_tripledecoy" accession="Rnd1psu|NC\_LIV\_080090">  <Seq>TPLRAFARSEISNPSLIVLDDVLSQSLSSPGSHLERLLPEAEPGRGAEIDTKPHKRTTAFFSLEHEAEAWPPAPPTPPPGNQYAPIDRVGSATGAD  DPPTAPFALVPPRRRNLGLAPELFALGSQQTRAGIYSGRVGATQNTGVGPSAQLAAQANPAELDTPSQLAGPGSLVARALWAPVTFFLENSSRSLSSNSSR  ...  GGGLRAELAVNELLVEDRGPDNSNYVVAVKFPKAGAKECSRKKDKLDLHGSCDICAGMGTATSMSCTYA</Seq>  ...  </cvParam> |

## Element <userParam>

|  |  |
| --- | --- |
| **Definition:** | A single user-defined parameter. |
| **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 name="D:/Software/Databases/Neospora\_3rndTryp/Neo\_rndTryp\_3times.fasta"/> |

# 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

*Searches of Nucleic acid databases* - 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 a protein dectection 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:

<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> for example 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 5.3.3. Reporting of threshold for modification position (also mzIdentML 1.2) is explained in section 5.3.4.

## 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.

<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>

<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"/>

<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"/>

<cvParam accession="MS:1001329" name="OMSSA:pvalue" cvRef="PSI-MS" value="0.00073351" />

…

</SpectrumIdentificationItem>

## 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>.

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

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

<cvParamType>: A single entry from an ontology or a controlled vocabulary. 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 ion 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">

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

<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"/>

</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 cvRef="PSI-PI" accession=" MS:1001239" name="frag: immonium ion" index="2 3 7" charge="1">

<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"/>

</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 ). 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 cvTerm 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 userParam 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 | (?<=W) |

Table 5. 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), 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.

# 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. These artefacts are currently undergoing the PSI document process standardization process, which will result in a standard officially sanctioned by PSI.

# Authors and Contributors

Authors of this specification:

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

Sean Seymour, AB SCIEX

Terry Farrah, Institute for Systems Biology

Andrew Jones, University of Liverpool

Correspondence - [andrew.jones@liv.ac.uk](mailto:andrew.jones@liv.ac.uk)

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, the following people contributed to the model development, gave feedback or tested mzIdentML:

* Eric Deutsch, Institute for Systems Biology
* Simon Hubbard, University of Manchester
* Julian Selley, University of Manchester
* Zsuzsanna Bencsath-Makkai, Biomedical Engineering, McGill University
* Sean Seymour, Applied Biosystems
* 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 Martinez, ProteoRed Consortium, Spain
* Laurent Gatto, University of Cambridge

# References

* [RFC2119] Bradner, S. (1997). "Key words for use in RFCs to Indicate Requirement Levels, Internet Engineering Task Force, RFC 2119, <http://www.ietf.org/rfc/rfc2119.txt>.
* [Jones 07] Jones AR, Miller M, Spellman P and Pizarro A. Specification documentation for the Functional Genomics Experiment (FuGE) model: user guide. Version 1 (final): <http://fuge.sourceforge.net/dev/V1Final/FuGE-v1-SpecDoc.doc>.
* [Deutsch08] Deutsch EW, Martens L, Montecchi-Palazzi L, Binz P-A, Kessner D, Souda P mzML: Mass Spectrometry Markup Language, <http://www.psidev.info/index.php?q=node/303>.

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, R110.000133

2. 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, 2332-2340

3. 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, 2389-2399

# 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

Copyright (C) Proteomics Standards Initiative (2016). All Rights Reserved.

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this paragraph are included on all such copies and derivative works. However, this document itself may not be modified in any way, such as by removing the copyright notice or references to the PSI or other organizations, except as needed for the purpose of developing Proteomics Recommendations in which case the procedures for copyrights defined in the PSI Document process must be followed, or as required to translate it into languages other than English.

The limited permissions granted above are perpetual and will not be revoked by the PSI or its successors or assigns.

This document and the information contained herein is provided on an "AS IS" basis and THE PROTEOMICS STANDARDS INITIATIVE DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE."