

Johannes Griss, European Bioinformatics Institute  
Timo Sachsenberg, University of Tübingen  
Mathias Walzer, University of Tübingen  
Oliver Kohlbacher, University of Tübingen  
Andrew R. Jones, University of Liverpool  
Henning Hermjakob, European Bioinformatics Institute  
Juan Antonio Vizcaíno, European Bioinformatics Institute

June 1, 2012  
Updated: June 20, 2014

## **mzTab: exchange format for proteomics and metabolomics results**

### Status of This Document

This document presents the final specification of the mzTab data format developed by members of the Human Proteome Organisation (HUPO) Proteomics Standards Initiative (PSI) Proteomics Informatics (PI) Working Group. Distribution is unlimited.

### Version of This Document

The current version of this document is: version 1.0.0, June 20 2014.

## **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 quantification processes for proteins, peptides, small molecules and protein modifications from mass spectrometry. This document defines a tab delimited text file format to report proteomics and metabolomics results.

## **Contents**

Abstract.....	1
1. Introduction .....	2
1.1 Background .....	2
1.2 Document Structure .....	3
2. Use Cases for mzTab.....	3
3. Notational Conventions .....	4
4. Relationship to Other Specifications.....	5
4.1 The PSI Mass Spectrometry Controlled Vocabulary (CV) .....	5
5. Resolved Design and scope issues.....	6
5.1 Handling updates to the controlled vocabulary.....	6
5.2 Use of identifiers for input spectra to a search .....	6
5.3 Recommendations for reporting replicates within experimental designs .....	8

5.4	mzTab types 'Identification' and 'Quantification' .....	9
5.5	mzTab modes 'Summary' and 'Complete' .....	10
5.6	Recommendations for reporting protein inference.....	13
5.7	Recommendations for reporting quantification results.....	14
5.8	Reporting modifications and amino acid substitutions.....	15
5.9	Encoding missing values, zeroes, nulls, infinity and calculation errors.....	16
5.10	Number of peptides reported .....	17
5.11	Reliability score.....	17
5.12	Comments on Specific Use Cases.....	18
5.13	Other supporting materials.....	19
6.	Format specification .....	20
6.1	Sections .....	21
6.2	Metadata Section .....	22
6.3	Protein Section.....	35
6.4	Peptide Section .....	41
6.5	PSM Section.....	46
6.6	Small Molecule Section .....	51
7.	Non-supported use cases .....	57
8.	Conclusions.....	57
9.	Authors.....	57
10.	Contributors.....	57
11.	References.....	58
12.	Intellectual Property Statement .....	58
	TradeMark Section .....	59
	Copyright Notice .....	59

## 1. Introduction

### 1.1 Background

This document addresses the systematic description of peptide, protein, and small molecule identification and quantification data retrieved from mass spectrometry (MS)-based experiments. A large number of software tools are available that analyze MS data and produce a variety of different output data formats. The HUPO Proteomics Standards Initiative (PSI) has developed several vendor-neutral data formats to overcome this heterogeneity of data formats for MS data. Currently, the PSI promotes the usage of three file formats to report an experiment's data: mzML to store the pure MS data (i.e. the spectra and chromatograms), mzIdentML to store (poly)peptide identifications and potentially inferred protein identifications, and mzQuantML to store quantitative data associated with these results. All three of these formats are XML-based and require sophisticated software to access the stored data.

While full, detailed representation of MS data including provenance is essential for researchers in the field, many downstream analysis use cases are only concerned with the *results* of the experiment in an easily accessible format. In addition, there is a trend for performing more integrated experimental workflows involving both proteomics and metabolomics data. Thus, the current lack of standardization in the field of metabolomics was taken into account in the development of the format presented here, and structures were developed that can report protein, peptide, and small molecule MS based data.

mzTab is intended as a lightweight supplement to the already existing standard file formats, providing a summary, similar to the supplementary table of results of a scientific publication.

mzTab files can contain protein, peptide, and small molecule identifications together with basic quantitative information. mzTab is not intended to store an experiment's complete data / evidence but only its final reported results. This format is also intended to provide local LIMS systems as well as MS proteomics repositories a simple way to share and combine basic information.

mzTab has been developed with a view to support the following general tasks (more specific use cases are provided in Section 2):

- T1. *Facilitate the sharing of final experimental results*, especially with researchers outside the field of proteomics that i) lack specialized software to parse the existing PSI's XML-based standard file formats, and ii) are only interested in the final reported results and not in all the details related to the data processing due to the inherent complexity of MS proteomics data. Furthermore, this should encourage the development of small innovative tools without the requirement of parsing huge XML files, which might be outside the scope of many bioinformaticians.
- T2. *Export of results to external software*, that is not able to parse proteomics/metabolomics specific data formats but can handle simple tab-delimited file formats. As a guideline the file format is designed to be viewable by programs such as Microsoft Excel® and Open Office Spreadsheet.
- T3. *Contain the results of an experiment in a single file*, and thus not require linking two files to retrieve identification and quantification results to again simplify the processing of the data.
- T4. *Act as an output format of (web-) services* that report MS-based results and thus can produce standardized result pages.
- T5. *Allow the combination of MS-based proteomics and metabolomics experimental results* within a single file.
- T6. *Be able to link to the external experimental evidence* (i.e. the mass spectra in different formats), following the same approach used in mzIdentML and mzQuantML.

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. It is anticipated that tutorial material will be developed independently of this specification.

## 1.2 Document Structure

The remainder of this document is structured as follows. Section 2 lists use cases mzTab 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 5 discusses the reasoning behind several design decisions taken. Section 6 contains the documentation of the file. Section 7 lists use cases that are currently not supported. Conclusions are presented in Section 8.

## 2. Use Cases for mzTab

The following cases of usage have driven the development of the mzTab data model, and are used to define the scope of the format in version 1.0.

1. mzTab files should be simple enough to make proteomics/metabolomics results accessible to people outside the respective fields. This should facilitate the sharing of data beyond the borders of the fields and make it accessible to non-experts.
2. mzTab files should contain sufficient information to provide an electronic summary of all findings in a proteomics/metabolomics study to permit its use as a standard documentation format for 'supplementary material' sections of publications in proteomics and metabolomics. It should thus be able to replace PDF tables as a way of reporting peptides and proteins and make published identification and quantification information more accessible.
3. mzTab files should enable reporting at different levels of detail: ranging from a simple summary of the final results to a detailed reporting including the experimental design. In practise, when different samples and assays (including replicates) are reported in a single mzTab file, this file can be generated in two ways: 'Summary' mode, and 'Complete' mode. In 'Summary' full results per assay/replicate need not be included, only the final data for the experimental conditions analysed must be present. In 'Complete' mode, all the results per assay/replicate need to be detailed.
4. It should be possible to open mzTab files with "standard" software such as Microsoft Excel® or Open Office Spreadsheet. This should furthermore improve the usability of the format to people outside the fields of proteomics/metabolomics.
5. It should be possible to export proteomics data from, for example, mzIdentML/mzQuantML files into mzTab to then load this data into, for example, statistical tools such as those provided through the R programming language. With the current formats, complex conversion software would be needed to make proteomics results available to such environments.
6. mzTab files should make MS derived results easily accessible to scripting languages allowing bioinformaticians to develop software without the overhead of developing sophisticated parsing code. Since mzTab files will be comparatively small, the data from multiple experiments can be processed at once without requiring special resource management techniques.
7. It should be possible to contain the complete final results of an MS-based proteomics/metabolomics experiment in a single file. This should furthermore reduce the complexity of sharing and processing an experiment's final results. mzTab files should be able to store quantitative values for protein, peptide, and small molecule identifications. Furthermore, mzTab files should contain basic protein inference information and modification position ambiguity information. Additionally, mzTab files should be able to report merged results from multiple search engines.
8. It should be useful as an output format by web-services that can then be readily accessed by tools supporting mzTab.
9. As mzTab files only contain an experiment's core results, all entries should link back to their source. Furthermore, it should be possible to directly link a given peptide / small molecule identification to its source spectrum in an external MS data file. The same referencing system as in mzIdentML/mzQuantML should be used.

### 3. Notational Conventions

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 (Bradner 1997).

## 4. Relationship to Other Specifications

The specification described in this document has not been 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. *mzML* (<http://www.psdev.info/mzml>). *mzML* is the PSI standard for capturing mass spectra / peak lists resulting from mass spectrometry in proteomics (Martens, L., *et al.* 2011). *mzTab* files MAY be used in conjunction with *mzML*, although it will be possible to use *mzTab* with other formats of mass spectra. This document does not assume familiarity with *mzML*.
2. *mzIdentML* (<http://www.psdev.info/mzidentml>). *mzIdentML* is the PSI standard for capturing of peptide and protein identification data (Jones, A. R., *et al.* 2012). *mzTab* files MAY reference *mzIdentML* files that then contain the detailed evidence of the reported identifications.
3. *mzQuantML* (<http://www.psdev.info/mzquantml>). *mzQuantML* is the PSI standard for capturing quantitative proteomics data from mass spectrometry (Walzer, M. *et al.* 2013). *mzTab* files that report quantitative data MAY reference *mzQuantML* files for detailed evidence of the reported values.

### 4.1 The PSI Mass Spectrometry Controlled Vocabulary (CV)

The PSI-MS controlled vocabulary is intended to provide terms for annotation of *mzML*, *mzIdentML*, and *mzQuantML* files. The CV has been generated with a 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 with a “has\_units” key in the CV itself (relationship: has\_units: UO:0000221 ! dalton).

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. If there is no obvious consensus, the CV coordinators committee should vote and make a decision. A new *psi-ms.obo* should then be released by updating the file on the CVS server without changing the name of the file (this would alter the propagation of the file to the OBO website and to other ontology services that rely on file stable URI). For this reason an internal version number with two decimals (x.y.z) should be increased:

- x should be increased when a first level term is renamed, added, deleted or rearranged in the structure. Such rearrangement will be rare and is very likely to have repercussion on the mapping.
- y should be increased when any other term except the first level one is altered.
- z should be increased when there is no term addition or deletion but just editing on the definitions or other minor changes.

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/>)
- OBI (Ontology of Biological Investigations - <http://obi.sourceforge.net/>)
- PSI Protein modifications workgroup - <http://psidev.cvs.sourceforge.net/psidev/psi/mod/data/PSI-MOD.obo>
- Unimod modifications database - <http://www.unimod.org/obo/unimod.obo>
- PRIDE Controlled Vocabulary ([http://ebi-pride.googlecode.com/svn/trunk/pride-core/schema/pride\\_cv.obo](http://ebi-pride.googlecode.com/svn/trunk/pride-core/schema/pride_cv.obo))
- NEWT UniProt Taxonomy Database (<http://www.ebi.ac.uk/ontology-lookup/browse.do?ontName=NEWT>)
- BRENDA tissue/ enzyme source ([http://www.brenda-enzymes.info/ontology/tissue/tree/update/update\\_files/BrendaTissueOBO](http://www.brenda-enzymes.info/ontology/tissue/tree/update/update_files/BrendaTissueOBO)).
- Cell Type ontology ([http://obo.cvs.sourceforge.net/obo/obo/ontology/anatomy/cell\\_type/cell.obo](http://obo.cvs.sourceforge.net/obo/obo/ontology/anatomy/cell_type/cell.obo)).

## 5. Resolved Design and scope issues

There were several issues regarding the design of the format that were not clear cut, and a design choice was made that was not completely agreeable to everyone. So that these issues do not keep coming up, we document the issues here and why the decision that is implemented was made.

### 5.1 Handling updates to the controlled vocabulary

There is a difficult issue with respect to how software should encode CV terms, such that changes to the core can be accommodated. This issue is discussed at length in the mzML specification document (Martens, L *et al.* 2011), and mzTab follows the same convention. In brief, when a new term is required, the file producers must contact the CV working group (via the mailing list [psidev-ms-vocab@lists.sourceforge.net](mailto: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.

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

PSMs and small molecules **MUST** be linked to an identifier of the source spectrum (in an external file) from which the identifications are made by way of a reference in the spectra\_ref attribute and via the ms\_run 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. This is the exact same approach followed in mzIdentML and mzQuantML. *Note, this table shows examples from the CV but will be extended. The CV holds the definite specification for legal encodings of spectrumID values.*

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
MS:1000769	Waters nativeID format	function=xsd:positiveInteger process=xsd:nonNegativeInteger scan=xsd:nonNegativeInteger	
MS:1000770	WIFF nativeID format	sample=xsd:nonNegativeInteger period=xsd:nonNegativeInteger cycle=xsd:nonNegativeInteger experiment=xsd:nonNegativeInteger	
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 conversion from mzXML, or a DTA folder where native scan numbers can be derived.
MS:1000777	spectrum identifier nativeID format	spectrum=xsd:nonNegativeInteger	Used for conversion from mzData. The spectrum id attribute is referenced.
MS:1001530	mzML unique identifier	xsd:string	Used for referencing mzML. The value of the spectrum id attribute is referenced directly.

**Table 1.** Controlled vocabulary terms and rules implemented in the PSI-MS CV for formulating the “nativeID” to identify spectra in different file formats.

In mzTab, the spectra\_ref attribute should be constructed following the data type specification in Table 1. As an example, to reference the third spectrum (index = 2) in an MGF (Mascot Generic Format) file:

```

MTD  ms_run[1]-format      [MS, MS:1001062, Mascot MGF file, ]
MTD  ms_run[1]-id_format  [MS, MS:1000774, multiple peak list nativeID format, ]

...

PSH  sequence    ...    spectra_ref          ...
PSM  NILNELFQR   ...    ms_run[1]:index=2    ...

```

Example: Reference the spectrum with identifier “scan=11665” in an mzML file.

```

MTD  ms_run[1]-format      [MS, MS:1000584, mzML file, ]
MTD  ms_run[1]-id_format   [MS, MS:1001530, mzML unique identifier, ]

...

PSH  sequence      ...   spectra_ref      ...
PSM  NILNELFQR     ...   ms_run[1]:scan=11665  ...

```

### 5.3 Recommendations for reporting replicates within experimental designs

Modeling the correct reporting of technical/biological replicates within experimental designs is supported in mzTab using an adaptation of the system originally developed for mzQuantML comprising four components described below (Figure 1). These components have various cross-references and MUST be used in different types of mzTab files, as described in Section 5.4:

- **Study variable** – The variables about which the final results of a study are reported, which may have been derived following averaging across a group of replicate measurements (assays). In files where assays are reported, study variables have references to assays. The same concept has been defined by others as “experimental factor”.
- **MS run** – An MS run is effectively one run (or set of runs on pre-fractionated samples) on an MS instrument, and is referenced from assay in different contexts.
- **Assay** – The application of a measurement about the sample (in this case through MS) – producing values about small molecules, peptides or proteins. One assay is typically mapped to one MS run in the case of label-free MS analysis or multiple assays are mapped to one MS run for multiplexed techniques, along with a description of the label or tag applied.
- **Sample** – a biological material that has been analysed, to which descriptors of species, cell/tissue type etc. can be attached. In all of types of mzTab file, these MAY be reported in the metadata section as sample[1-n]-description. Samples are NOT MANDATORY in mzTab, since many software packages cannot determine what type of sample was analysed (e.g. whether biological or technical replication was performed).

Clear definitions of biological and technical replicates are difficult to provide as these are somewhat dependent upon the biological domain. However, we use the following general definitions in mzTab.

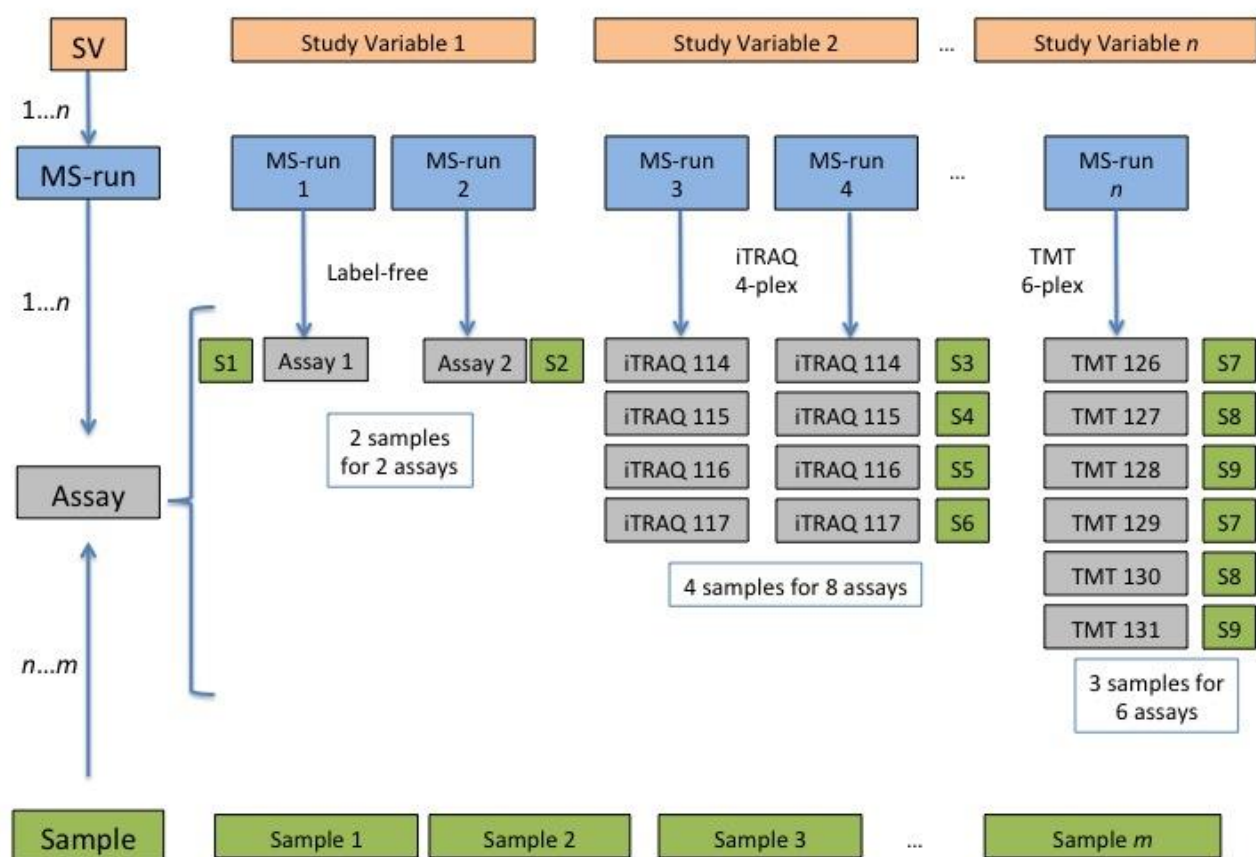
- Biological replicates are where different samples have been analysed by MS.
- Technical replicates are where same samples are analysed multiple times by (LC)-MS.

*Note: there is deliberately no attempt to define the boundary of the term “sample”.*

If sample level information is provided optimally, it is expected that  $n$  biological replicates can be mapped to sample[1-n];  $m$  technical replicate measurements of sample 1 SHOULD be mapped to assay[1-m] referencing sample[1] (for example). However, an open challenge



remains since analysis software is often not aware of whether replicates (multiple MS runs) are originally biological or technical in nature. As such, the default behavior for mzTab exporters from quantitative software is to exclude sample level information and report quantitative data for assay[1-n] and/or study\_variable[1-n] depending on whether it is a 'Complete' or 'Summary' file. Additional annotation software would typically be required to add the sample-level information, as provided (often manually) by the user.



**Figure 1.** Diagram summarizing the relation between Study Variables (SVs), MS runs, assays and samples.

## 5.4 mzTab types 'Identification' and 'Quantification'

There are two types of mzTab files which MUST be specified using the mandatory metadata field 'mzTab-type' ('Identification' or 'Quantification'). 'Identification' MUST be used to report raw peptide, protein and small molecule identifications. The type 'Quantification' MUST be used for quantification results (which optionally might contain identification results about the quantified protein/peptide or small molecules). 'Quantification' files MUST always report quantification data on the level of study variables and MAY report quantification data on the level of assays. In contrast, 'Identification' files MAY contain neither study variables nor assays but only report identifications on the level of MS runs. Of course, 'Identification' files SHOULD include information about study variables and assays if this information is available.

Providing metadata on samples is not mandatory in both mzTab types as most software for quantification and identification can't readily export this information.

## 5.5 mzTab modes 'Summary' and 'Complete'

There are two modes of reporting data in mzTab files: as 'Identification' and 'Quantification' type results. The type MUST be specified by the mandatory metadata field 'mzTab-mode' ('Summary' and 'Complete'). The 'Summary' mode is used to report final results (e.g. quantification data at the level of study variables). The 'Complete' mode is used if all quantification data is provided (e.g. quantification on the assay level and on the study variable level).

The MANDATORY fields in the Metadata Section 'mzTab-mode' and 'mzTab-type' MUST therefore be present to indicate which type of file it is. In general, "null" values SHOULD not be given within any column of a "Complete" file if the information is available. Tables 2-6 indicate which metadata or columns are mandatory for a specific mzTab-mode ('Summary' and 'Complete') and mzTab-type ('Identification' and 'Quantification') in the different sections.

In general, "null" values SHOULD not be used within any column of a "Complete" file if the information is available. This is the nomenclature used in these tables:

**S** ... required in summary file

**s** ... optional in summary file

**C** ... required in complete file

**c** ... optional in complete file

**SV** ... study variable

### Metadata Section

Field Name	Identification	Quantification
mzTab-version	<b>SC</b>	<b>SC</b>
mzTab-mode	<b>SC</b>	<b>SC</b>
mzTab-type	<b>SC</b>	<b>SC</b>
description	<b>SC</b>	<b>SC</b>
ms_run[1-n]-location	<b>SC</b>	<b>SC</b>
protein_search_engine_score[1-n]	<b>SC</b> (if protein section present)	<b>SC</b> (if protein section present)
peptide_search_engine_score[1-n]	<b>SC</b> (if peptide section present)	<b>SC</b> (if peptide section present)
psm_search_engine_score[1-n]	<b>SC</b> (if PSM section present)	<b>SC</b> (if PSM section present)
smallmolecule_search_engine_score[1-n]	<b>SC</b> (if small molecule section present)	<b>SC</b> (if small molecule section present)
fixed_mod[1-n]	<b>SC</b>	<b>SC</b>
variable_mod[1-n]	<b>SC</b>	<b>SC</b>
protein-quantification-unit	(not used)	<b>SC</b> (if protein section present)
peptide-quantification-unit	(not used)	<b>SC</b> (if peptide section present)
smallmolecule-quantification-unit	(not used)	<b>SC</b> (if small molecule section present)
study_variable[1-n]-description	(not used)	<b>SC</b>
software[1-n]	<b>sC</b>	<b>sC</b>
quantification_method	(not used)	<b>sC</b>
assay[1-n]-ms_run_ref	<b>sc</b> (required if assays reported)	<b>sC</b> (required if assays reported)
assay[1-n]-quantification_reagent	(not used)	<b>sC</b>
mzTab-ID	<b>sc</b>	<b>sc</b>
title	<b>sc</b>	<b>sc</b>
sample_processing[1-n]	<b>sc</b>	<b>sc</b>
instrument[1-n]-name	<b>sc</b>	<b>sc</b>

instrument[1-n]-source	SC	SC
instrument[1-n]-analyzer[1-n]	SC	SC
instrument[1-n]-detector	SC	SC
software[1-n]-setting[1-n]	SC	SC
false_discovery_rate	SC	SC
publication[1-n]	SC	SC
contact[1-n]-name	SC	SC
contact[1-n]-affiliation	SC	SC
contact[1-n]-email	SC	SC
uri[1-n]	SC	SC
fixed_mod[1-n]-site	SC	SC
fixed_mod[1-n]-position	SC	SC
variable_mod[1-n]-site	SC	SC
variable_mod[1-n]-position	SC	SC
ms_run[1-n]-format	SC	SC
ms_run[1-n]-id_format	SC (required if ms_run[1-n]-format reported)	SC (required if ms_run[1-n]-format reported)
ms_run[1-n]-fragmentation_method	SC	SC
ms_run[1-n]-hash	SC	SC
ms_run[1-n]-hash_method	SC (required if ms_run[1-n]-hash reported)	SC (required if ms_run[1-n]-hash reported)
custom[1-n]	SC	SC
sample[1-n]-species[1-n]	SC	SC
sample[1-n]-tissue[1-n]	SC	SC
sample[1-n]-cell_type[1-n]	SC	SC
sample[1-n]-disease[1-n]	SC	SC
sample[1-n]-description	SC	SC
sample[1-n]-custom[1-n]	SC	SC
study_variable[1-n]-description	SC (required if SV reported)	SC (required if SV reported)
study_variable[1-n]-sample_refs	SC	SC
study_variable[1-n]-assay_ref	SC	SC
assay[1-n]-quantification_mod[1-n]	(not used)	SC
assay[1-n]-quantification_mod[1-n]-position	(not used)	SC
assay[1-n]-quantification_mod[1-n]-site	(not used)	SC
assay[1-n]-sample_refs	(not used)	SC
cv[1-n]-label	SC	SC
cv[1-n]-full_name	SC	SC
cv[1-n]-version	SC	SC
cv[1-n]-url	SC	SC
colunit_protein	SC	SC
colunit_peptide	SC	SC
colunit_psm	SC	SC
colunit_small_molecule	SC	SC

**Table 2.** Mandatory and optional metadata in the Metadata section**Protein Section**

Field Name	Identification	Quantification
accession	SC	SC
description	SC	SC
taxid	SC	SC
species	SC	SC
database	SC	SC
database_version	SC	SC

search_engine	SC	SC
best_search_engine_score[1-n]	SC	SC
ambiguity_members	SC	SC
modifications	SC	SC
protein_coverage	sC	sC
protein_abundance_study_variable[1-n]	(not used)	SC
protein_abundance_stdev_study_variable[1-n]	(not used)	SC
protein_abundance_std_error_study_variable[1-n]	(not used)	SC
search_engine_score[1-n]_ms_run[1-n]	sC	sC
num_psms_ms_run[1-n]	sC	sc
num_peptides_distinct_ms_run[1-n]	sC	sc
num_peptide_unique_ms_run[1-n]	sC	sc
protein_abundance_assay[1-n]	(not used)	sC
opt_{identifier}_*	sc	sc
go_terms	sc	sc
reliability	sc	sc
uri	sc	sc

**Table 3.** Mandatory and optional columns in the Protein section

### Peptide Section (not recommended in 'Identification' files)

Field Name	Identification	Quantification
sequence	(not used)	SC
accession	(not used)	SC
unique	(not used)	SC
database	(not used)	SC
database_version	(not used)	SC
search_engine	(not used)	SC
best_search_engine_score[1-n]	(not used)	SC
modifications	(not used)	SC
retention_time	(not used)	SC
retention_time_window	(not used)	SC
charge	(not used)	SC
mass_to_charge	(not used)	SC
peptide_abundance_study_variable[1-n]	(not used)	SC
peptide_abundance_stdev_study_variable[1-n]	(not used)	SC
peptide_abundance_std_error_study_variable[1-n]	(not used)	SC
search_engine_score[1-n]_ms_run[1-n]	(not used)	sC
peptide_abundance_assay[1-n]	(not used)	sC
spectra_ref	(not used)	sC (if MS2 based quantification is used)
opt_{identifier}_*	(not used)	sc
reliability	(not used)	sc
uri	(not used)	sc

**Table 4.** Mandatory and optional columns in the Peptide section

### PSM Section

Field Name	Identification	Quantification
sequence	SC	SC
PSM_ID	SC	SC
accession	SC	SC
unique	SC	SC
database	SC	SC
database_version	SC	SC

search_engine	SC	SC
search_engine_score[1-n]	SC	SC
modifications	SC	SC
spectra_ref	SC	SC
retention_time	SC	SC
charge	SC	SC
exp_mass_to_charge	SC	SC
calc_mass_to_charge	SC	SC
pre	SC	SC
post	SC	SC
start	SC	SC
end	SC	SC
opt_{identifier}_*	sc	sc
reliability	sc	sc
uri	sc	sc

**Table 5.** Mandatory and optional columns in the PSM section

### Small Molecule Section

Field Name	Identification	Quantification
identifier	SC	SC
chemical_formula	SC	SC
smiles	SC	SC
inchi_key	SC	SC
description	SC	SC
exp_mass_to_charge	SC	SC
calc_mass_to_charge	SC	SC
charge	SC	SC
retention time	SC	SC
taxid	SC	SC
species	SC	SC
database	SC	SC
database_version	SC	SC
spectra_ref	SC	SC
search_engine	SC	SC
best_search_engine_score[1-n]	SC	SC
modifications	SC	SC
smallmolecule_abundance_assay[1-n]	(not used)	SC (if assays reported)
smallmolecule_abundance_study_variable[1-n]	(not used)	SC (if study vars. reported)
smallmolecule_stdev_study_variable[1-n]	(not used)	SC (if study vars. reported)
smallmolecule_std_error_study_variable[1-n]	(not used)	SC (if study vars. reported)
search_engine_score[1-n]_ms_run[1-n]	(not used)	sC
opt_{identifier}_*	sc	sc
reliability	sc	sc
uri	sc	sc

**Table 6.** Mandatory and optional columns in the Small Molecule section

## 5.6 Recommendations for reporting protein inference

There are multiple approaches to how protein inference can be reported. mzTab is designed to only hold experimental results, which in proteomics experiments can be very complex. At the same time, for downstream statistical analysis there is a need to simplify this problem. It is not possible to model detailed protein inference data without a significant level of complexity at the file format level. Therefore, it was decided to have only limited support for protein

inference/grouping reporting in mzTab files. Protein entries in mzTab files contain the field `ambiguity_members`. The protein accessions listed in this field should identify proteins that were also identified through the same set of peptides or spectra, or proteins supported by a largely overlapping set of evidence, and could also be a viable candidate for the “true” identification of the entity reported. It is RECOMMENDED that “subset proteins” that are unlikely to have been identified SHOULD NOT be reported here. The mapping of a single peptide-spectrum match (PSM) to multiple accessions is supported through the reporting of the same PSM on multiple rows of the PSM section, as exemplified below.

COM Example of how protein inference is reported. Other sections and several columns are omitted.

```
...
PRH  accession  ...  ambiguity_members  ...
PRT  P14602     ...  Q340U4, P16627  ...
...
PSH          sequence  PSM_ID  accession  unique  ...
PSM          DWYPAHSR   4       P14602     0       ...
PSM          DWYPAHSR   4       Q340U4     0       ...
PSM          DWYPAHSR   4       P16627     0       ...
```

## 5.7 Recommendations for reporting quantification results

Quantitative technologies generally result in some kind of abundance measurement of the identified analyte. Furthermore, several of the available techniques, furthermore, allow/require multiple samples to be multiplexed and analyzed in a single MS run – for example in label-based techniques, such as SILAC/N<sup>15</sup> where quantification occurs on MS<sup>1</sup> data or in tag-based techniques, such as iTRAQ/TMT where quantification occurs in MS<sup>2</sup> data.

One measurement of a small molecule, peptide or protein is mapped to the concept of assay in multiplexed techniques and label-free approaches in complete files. Each assay MUST have a reference to the quantification reagent/label used (“unlabelled” in the label-free case and the “light” channel in SILAC/N<sup>15</sup>) and each assay MUST have a reference to the `ms_run[1_n]` from which it originated. As such, in multiplexed techniques where *n* reagents are used within one analysis, `assay[1-n]` MUST reference the same `ms_run`.

If the data exporter wishes to report only final results for ‘Summary’ files (i.e. following averaging over replicates), then these MUST be reported as quantitative values in the columns associated with the `study_variable[1-n]` (e.g. `protein_abundance_study_variable[1]`). mzTab allows the reporting of abundance, standard deviation, and standard error for any `study_variable`. The unit of values in the abundance column MUST be specified in the metadata section of the mzTab file. The reported values SHOULD represent the final result of the performed data analysis. The exact meaning of the values will thus depend on the used analysis pipeline and quantitation method and is not expected to be comparable across multiple mzTab files.

Ratios can be generated by the file consumers based on the abundance values of the relevant conditions.

See coding examples for SILAC, iTRAQ and label free approaches from the relevant example files (listed in Section 5.13).



## 5.8 Reporting modifications and amino acid substitutions

Modifications are defined in the meta-data section and reported in the modification columns of the protein, peptide or PSM section.

### Defining modifications in the meta-data section:

The meta values “fixed\_mod[1-n]” and “variable\_mod[1-n]” describe all search modifications used to identify peptides and proteins of the mzTab file (e.g. carbamidomethylation, oxidation, labels/tags). This is the minimal information that **MUST** be provided for Complete Identification or Quantification files.

In addition, for each assay the optional meta-data assay[1-n]-quantification\_mod\* **MAY** be specified that allows to define details of modifications associated with the quantification reagent (e.g. SILAC label).

If no fixed or variable modifications are reported, then the following CV parameters **MUST** be used:

MS:1002453 (No fixed modifications searched)

MS:1002454 (No variable modifications searched)

### Reporting of modifications in columns of the protein, peptide and PSM sections:

Fixed modifications or modifications specified as quantification\_modification in the metadata Section **SHOULD NOT** be reported in protein (PRT) and peptide rows (PEP). In contrast, all variable modifications plus fixed modifications like those induced by the quantification reagents **MUST** be reported in peptide spectrum match rows (PSM).

Modifications or substitutions are modelled using a specific modification object with the following format:

**{position}{parameter}-{[modification or substitution identifier]}{[neutral loss]}**

The number of modification (or substitution) objects **MUST** correspond to the number of identified modifications (or substitutions) on a given peptide or PSM. It is also expected that modifications **SHOULD** be reported for proteins using the same format. However, it is recognised that some export software may not be able to do this. If software cannot determine protein-level modifications, “null” **MUST** be used. If the software has determined that there are no modifications to a given protein “0” **MUST** be used.

**{position}** is mandatory. However, if it is not known (e.g. MS1 Peptide Mass Fingerprinting), ‘null’ must be used. Terminal modifications in proteins and peptides **MUST** be reported with the position set to 0 (N-terminal) or the amino acid length +1 (C-terminal) respectively. This object allows modifications to be assigned to ambiguous locations, but only at the PSM and Peptide level. Ambiguity of modification position **MUST NOT** be reported at the Protein level. In that case, the modification element can be left empty. Ambiguous positions can be reported by separating the {position} and (optional) {cvParam} by an ‘|’ from the next position. Thereby, it is possible to report reliabilities / scores / probabilities etc. for every potential location.

Here only the modification field is given:

3-MOD:00412, 8-MOD:00412

3|4-MOD:00412, 8-MOD:00412

3|4|8-MOD:00412, 3|4|8-MOD:00412

TESTPEPTIDES with two known phosphorylation sites

First phosphorylation site can be either on S or T

Three possible positions for two phosphorylation sites

**{parameter}** is optional. It MAY be used to report a numerical value e.g. a probability score associated with the modification or location.

Reporting the first two possible sites for the phosphorylation with given probability score  
Here only the modification field is given:

```
3[MS,MS:1001876, modification probability, 0.8]|4[MS,MS:1001876, modification probability, 0.2]
MOD:00412, 8-MOD:00412
```

This option is not allowed though:

```
(3|4)[MS,MS:1001876, modification probability, 0.8]|7[MS,MS:1001876, modification probability, 0.2]-
MOD:00412
```

**{modification or substitution identifier}** for proteins and peptides modifications SHOULD be reported using either UNIMOD or PSI-MOD accessions. As these two ontologies are not applicable to small molecules, so-called CHEMMODs can also be defined. Two types of CHEMMODs are allowed: specifying a chemical formula or specifying a given *m/z* delta. The list of allowed Modification or Substitution identifiers therefore is:

```
CHEMMOD:+NH4
CHEMMOD:-18.0913
UNIMOD:18
MOD:00815
```

CHEMMODs SHOULD NOT be used for protein/peptide modifications if the respective entry is present in either the PSI-MOD or the UNIMOD ontology. Furthermore, mass deltas SHOULD NOT be reported if the given delta can be expressed through a known and unambiguous chemical formula. In the exceptional case that the modification cannot be reported with a term in PSI-MOD or UNIMOD (e.g. “unknown modification” MS: 1001460) but the delta mass is available, CHEMMODs MUST be used. Terms in PSI-MOD and UNIMOD not describing specific modifications MUST NOT be used.

All (identified) variable modifications as well as fixed modifications MUST be reported for every identification.

**{neutral loss}** is optional. Neutral losses are reported as cvParams. They are reported in the same way that modification objects are (as separate, comma-separated objects in the modification column). The position for a neutral loss MAY be reported.

```
PEH sequence ... modifications ...
COM Phosphorylation with a neutral loss:
PEP EISILACEIR ... 3-UNIMOD:21,3-[MS, MS:1001524, fragment neutral loss, 63.998285],7-UNIMOD:4
...
COM Neutral loss without an associated modification:
PEP EISILACEIR ... [MS, MS:1001524, fragment neutral loss, 63.998285],7-UNIMOD:4 ...
```

## 5.9 Encoding missing values, zeroes, nulls, infinity and calculation errors

In the table-based sections (protein, peptide, and small molecule) there MUST NOT be any empty cells. In case a given property is not available “null” MUST be used. This is, for example, the case when a URI is not available for a given protein (*i.e.* the table cell MUST NOT be empty but “null” has to be reported). If ratios are included and the denominator is zero, the “INF” value MUST be used. If the result leads to calculation errors (for example 0/0), this MUST be reported as “not a number” (“NaN”). In some cases, there is ambiguity with respect to these cases: e.g. in spectral counting if no peptide spectrum matches are observed

for a given protein, it is open for debate as to whether its abundance is zero or missing (“null”).

### 5.10 Number of peptides reported

There are columns allowed in the protein section to report the number of peptides supporting a given protein identification, which are MANDATORY for Complete Identification files.

- num\_psms\_ms\_run[1\_n]
  - The count of the total significant PSMs that can be mapped to the reported protein
- num\_peptides\_distinct\_ms\_run[1\_n]
  - The count of the number of different peptide sequences that have been identified above the significance threshold. Different modifications or charge states of the same peptide are not counted.
- num\_peptides\_unique\_ms\_run[1\_n]
  - The number of peptides that can be mapped uniquely to the protein reported. If ambiguity members have been reported, the count MUST be derived from the number of peptides that can be uniquely mapped to the group of accessions, since the assumption is that these accessions are supported by the same evidence.

The idea of these three columns is to give the researcher a quick overview of how well a given protein identification is supported by peptide identifications for a given ms\_run reported. The num\_psms column also provides the opportunity for reporting pseudo-quantitative (label-free) values from approaches in which no explicit quantification has been performed.

### 5.11 Reliability score

All protein, peptide, psm and small molecule identifications reported in an mzTab file MAY be assigned a reliability score (column “reliability” in all tables). This reliability only applies to the identification reliability but not to modification position and or quantification reliabilities. The idea is to provide a way for researchers and/or MS proteomics or metabolomics repositories to score the reported identifications based on their own criteria. This score is completely resource-dependent and MUST NOT be interpreted as a comparable score between mzTab files generated from different resources. The criteria used to generate this score SHOULD be documented by the data providers. If this information is not provided by the producers of mzTab files, “null” MUST be provided as the value for each of the protein, peptide or small molecule identification.

The reliability value, if provided, MUST be an integer between 1-3 in all but the *small molecule* section (see below) and SHOULD be interpreted as follows:

- 1: high reliability
- 2: medium reliability
- 3: poor reliability

For metabolomics (*small molecule* section), according to current MSI agreement, it should be reported as an integer between 1-4 and should be interpreted as follows:

- 1: identified metabolites

- 2: putatively annotated compounds
- 3: putatively characterized compound classes
- 4: unknown compounds

The idea behind this score was to mimic the general concept of “resource based trust”. For example, if one resource reports identifications with a given reliability this would be interpreted differently as an identification reported from another resource – depending on who is responsible for the given resource and how it is built. If resources now report their reliabilities using this metric and document how this metric is generated, a user can base his own interpretation of the results based on his trust in the resource. Furthermore, approaches to make various search engine scores comparable have failed so far. To prevent the notion that the reported scores represent comparable probabilities this very abstract metric was chosen. Resources **MUST** explicitly specify how these reliability scores are calculated and what metric they represent.

## 5.12 Comments on Specific Use Cases

Many special use cases for mzTab were considered during its development. Each of these use cases has a corresponding example file that exercises the relevant part of the format and provides a reference implementation example (see supporting documentation). Authors of software that create mzTab are encouraged to examine the examples that accompany this format release before implementing the writer.

### 5.12.1 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 mzTab as follows: mzTab files **SHOULD** only contain the “final” protein list generated by any such workflow. Any protein, peptide, and small molecule can be associated with any number of search engines as well as multiple search engine scores. Thus, it is possible to report which element was identified by which search engine together with the resulting scores.

### 5.12.2 Adding optional columns

Additional columns **MAY** be added to the end of rows in all the table-based sections (protein, peptide, PSM and small molecule). These columns represent information not included by default in the currently defined fields and differ from the specification of optionality with regards to columns that **MUST** be present in Summary or Complete files (Tables 2 and 3).

These column headers **MUST** start with the prefix “opt\_” followed by the identifier of the object they reference: assay, study variable, MS run or “global” (if the value relates to all replicates). Column names **MUST** only contain the following characters: ‘A’-‘Z’, ‘a’-‘z’, ‘0’-‘9’, ‘\_’, ‘-’, ‘[’, ‘]’, and ‘.’. CV parameter accessions **MAY** be used for optional columns following the format: opt\_{OBJECT\_ID}\_cv\_{accession}\_{parameter name}. Spaces within the parameter’s name **MUST** be replaced by ‘\_’.

The information stored within an optional column is completely up to the resource that generates the file. It **MUST** not be assumed that optional columns having the same name in different mzTab files contain the same type of information. CV parameter accessions **MAY** be used as optional column names according to the following convention:

opt\_{OBJECT\_ID}\_cv\_{accession}\_{parameter name}. Spaces within the parameter's name **MUST** be replaced by '\_'.

COM Example showing how emPAI values are reported in an additional column from MS run 1 using  
COM MS CV parameter "emPAI value" (MS:1001905)

```
...
PRH  accession  ...  opt_ms_run[1]_cv_MS:1001905_emPAI_value
PRT  P12345      ...  0.658
```

### 5.12.3 Referencing external resources (i.e. mzIdentML or mzQuantML files)

In mzTab all identifications MAY reference external resources that contain detailed evidence for the identification. This link is stored in the "uri" column of the respective table. This field **MUST NOT** be used to reference an external MS data file. MS data files should be referenced using the method described in Section 5.2.

Where these URIs point to depends on the resource that generated the mzTab file. If, for example, PeptideAtlas was exporting data in the mzTab format the URI would be expected to point to the identification's entry within the respective PeptideAtlas build. mzTab files originating from an mzIdentML file MAY reference the mzIdentML file using the URI column. In case quantitative values are reported coming from an mzQuantML file, the mzQuantML file **SHOULD** be referenced as it contains the reference to the underlying mzIdentML file.

### 5.12.4 Reporting sequence ambiguity

In MS based proteomics approaches, some amino acids cannot be unambiguously identified. To report such ambiguous amino acid identifications, the following symbols **SHOULD** be used:

Asparagine or aspartic acid	B
Glutamine or glutamic acid	Z
Leucine or Isoleucine	J
Unspecified or unknown amino acid	X

### 5.12.5 Reporting decoy peptide identifications

To report the results of a target-decoy search, decoy identifications MAY be labeled using the optional column "opt\_cv\_MS:1002217\_decoy\_peptide". The value of this column **MUST** be a Boolean (1/0).

## 5.13 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 from:

<http://code.google.com/p/mztab/wiki/ExampleFiles>.

- SILAC\_SQ – (hand crafted) Minimal "Summary Quantification report", SILAC experiment, quantification on 2 study variables (control/treatment).
- iTRAQ\_SQI – (hand crafted) Minimal "Summary Quantification report", iTRAQ experiment, quantification on 4 study variables (t=0, t=1, t=2, t=3), identifications reported.
- labelfree\_SQI – (hand crafted) Minimal "Summary Quantification report", labelfree experiment, quantification on 2 study variables (control/treatment), identifications reported.
- SILAC\_CQI.mzTab - (hand crafted) "Complete Quantification report" SILAC experiment, quantification on 2 study variables (control/treatment), 3+3 assays (replicates) reported, identifications reported.
- iTRAQ\_CQI.mzTab - (hand crafted) "Complete Quantification report" iTRAQ experiment, quantification on 4 study variables (t=0, t=1, t=2, t=3), 4\*4 assays (4 replicate experiments) reported, identifications reported.

- f) labelfree\_CQI.mzTab – (hand crafted) "Complete Quantification report" label free experiment, quantification on 2 study variables (control/treatment), 3+3 assays (replicates) reported, identifications reported.
- g) PRIDE\_Exp\_Complete\_Ac\_16649.xml-mztab.txt – file generated using the mztab-exporter (converted PRIDE experiment accession 16649) containing iTRAQ data.
- h) PRIDE\_Exp\_Complete\_Ac\_1643.xml-mztab.txt – file generated using the mztab-exporter (converted PRIDE experiment accession 1643) containing peptide and protein identification data.
- i) lipidomics-HFD-LD-study-TG.mzTab – File generated by the LipidDataAnalyzer (LDA) mzTab export for small molecules. Report of a "Complete Quantification report" lipidomics experiment for the lipid class TG. Quantification on 3 study variables (HFD/FED/FAS), 6+6+6 assays (biological replicates) reported, identifications reported.
- j) lipidomics-HFD-LD-study-PL-DG-SM.mzTab – File generated by the LDA mzTab export for small molecules. Report of a "Complete Quantification report" lipidomics experiment for the lipid classes SM, PE, PC, LPC, DG, PS. Quantification on 3 study variables (HFD/FED/FAS), 6+6+6 assays (biological replicates) reported, identifications reported.
- k) Cytidine.mzTab – File generated manually. It describes the identification of cytidine.
- l) MTBLS2.mzTab – mzTab generated from the metabolites identified from comparative LC/MS-based profiling of silver nitrate-treated *Arabidopsis thaliana* leaves of wild-type and *cyp79B2 cyp79B3* double knockout plants.

## 6. Format specification

This section describes the structure of an mzTab file.

- **Field separator**  
The column delimiter is the Unicode Horizontal Tab character (Unicode codepoint 0009).
- **File encoding**  
The UTF-8 encoding of the Unicode character set is the preferred encoding for mzTab files. However, parsers should be able to recognize commonly used encodings.
- **Case sensitivity**  
All column labels and field names are case-sensitive.
- **Line prefix**  
Every line in an mzTab file MUST start with a three letter code identifying the type of line delimited by a Tab character. The three letter codes are as follows:
  - MTD for metadata
  - PRH for the protein table header line (the column labels)
  - PRT for rows of the protein table
  - PEH for the peptide table header line (the column labels)
  - PEP for rows of the peptide table
  - PSH for the PSM table header (the column labels)
  - PSM for rows of the PSM table
  - SMH for small molecule table header line (the column labels)
  - SML for rows of the small molecule table
  - COM for comment lines
- **Header lines**  
Each table based section (protein, peptide, PSM and small molecule) MUST start with the corresponding header line. These header lines MUST only occur once in the document since each section also MUST only occur once.



- **Dates**  
Dates and times MUST be supplied in the ISO 8601 format (“YYYY-MM-DD”, “YYYY-MM-DDTHH:MMZ” respectively).
- **Decimal separator**  
In mzTab files the dot (“.”) MUST be used as decimal separator. Thousand separators MUST NOT be used in mzTab files.
- **Comment lines and empty lines**  
Comment lines can be placed anywhere in an mzTab file. These lines must start with the three-letter code COM and are ignored by most parsers. Empty lines can also occur anywhere in an mzTab file and are ignored.
- **Params**  
mzTab makes use of CV parameters. As mzTab is expected to be used in several experimental environments where parameters might not yet be available for the generated scores etc. all parameters can either report CV parameters or user parameters that only contain a name and a value.  
Parameters are always reported as [CV label, accession, name, value]. Any field that is not available MUST be left empty.

```
[MS, MS:1001207, Mascot,]
[, ,A user parameter, The value]
```

In case, the name of the param contains commas, quotes MUST be added to avoid problems with the parsing: [label, accession, “first part of the param name, second part of the name”, value].

```
[MOD, MOD:00648, "N,O-diacetylated L-serine",]
```

- **Sample IDs**  
To be able to supply metadata specific to each sample, ids in the format sample[1-n] are used.

```
MTD    sample[1]-species[1]    [NEWT, 9606, Homo sapiens (Human), ]
```

- **Assay IDs**  
To be able to supply metadata specific to each assay, ids in the format assay[1-n] are used.

```
MTD    assay[1]-quantification_reagent    [MS,MS:1002038,unlabeled sample,]
```

- **Study variable IDs**  
To be able to supply metadata specific to each study variable (grouping of assays), ids in the format study\_variable[1-n] are used.

```
MTD    study_variable[1]-description Group B (spike-in 0.74 fmol/uL)
```

## 6.1 Sections

mzTab files can contain five different sections. The MANDATORY metadata section is made up of key-value pairs. The other four sections are OPTIONAL: protein, peptide, PSM and small molecule section are table-based.

Every section in an mzTab file MUST only occur once if present. If the PSM, Peptide and Protein Sections are present, the information MUST be consistent between these sections.

Field names with indices in square brackets **MUST** be numbered sequentially and non-decreasing (starting at the first value indicated in the bracket; single integer steps).

## 6.2 Metadata Section

The metadata section can provide additional information about the dataset(s) reported in the mzTab file. All fields in the metadata section are optional apart from several exceptions:

- “mzTab-version” **MUST** always be reported.
- “mzTab-mode” **MUST** always be reported. Two modes are possible: ‘Summary’ and ‘Complete’.
- “mzTab-type” **MUST** always be reported. Two types are possible: ‘Quantification’ or ‘Identification’. Any analyses generating both quantification and identification results **MUST** be flagged as ‘Quantification’.
- “description” **MUST** always be reported.
- “ms\_run-location[1-n]” **MUST** always be reported.
- “protein\_search\_engine\_score[1-n]”, “peptide\_search\_engine\_score[1-n]”, “psm\_search\_engine\_score[1-n]” and “smallmolecule\_search\_engine\_score[1-n]” **MUST** be reported for every search engine score reported in the corresponding section.
- “fixed\_mod[1-n]” and “variable\_mod [1-n]” **MUST** be reported. If no modifications were searched, specific CV parameters need to be used (see Section 5.8).

In addition, various other metadata parameters are **REQUIRED** for different file types, as defined above and in Tables 2-6.

The fields in the metadata section should be reported in order of the various fields listed here. The field’s name and value **MUST** be separated by a tab character:

```
MTD    publication    [PRIDE, PRIDE:00000029, PubMed, 12345]
```

In the following list of fields any term encapsulated by {} is meant as a variable which **MUST** be replaced accordingly.

### 6.2.1 mzTab-version

<b>Description:</b>	The version of the mzTab file.		
<b>Type:</b>	String		
<b>Mandatory</b>		Summary	Complete
	Quantification	✓	✓
	Identification	✓	✓
<b>Example:</b>	MTD    mzTab-version    1.0.0		

### 6.2.2 mzTab-mode

<b>Description:</b>	The results included in an mzTab file can be reported in 2 ways: ‘Complete’ (when results for each assay/replicate are included) and ‘Summary’, when only the most representative results are reported.		
<b>Type:</b>	Enum		
<b>Mandatory</b>		Summary	Complete
	Quantification	✓	✓
	Identification	✓	✓
<b>Example:</b>	MTD    mzTab-mode    Complete MTD    mzTab-mode    Summary		

## 6.2.3 mzTab-type

<b>Description:</b>	The results included in an mzTab file MUST be flagged as 'Identification' or 'Quantification' - the latter encompassing approaches that are quantification only or quantification and identification.		
<b>Type:</b>	Enum		
<b>Mandatory</b>		Summary	Complete
	Quantification	✓	✓
	Identification	✓	✓
<b>Example:</b>	MTD    mzTab-type    Quantification MTD    mzTab-type    Identification		

## 6.2.4 mzTab-ID

<b>Description:</b>	The ID of the mzTab file.		
<b>Type:</b>	String		
<b>Mandatory</b>		Summary	Complete
	Quantification		
	Identification		
<b>Example:</b>	MTD    mzTab-ID    PRIDE_1234		

## 6.2.5 title

<b>Description:</b>	The file's human readable title.		
<b>Type:</b>	String		
<b>Mandatory</b>		Summary	Complete
	Quantification		
	Identification		
<b>Example:</b>	MTD    title    My first test experiment		

## 6.2.6 description

<b>Description:</b>	The file's human readable description.		
<b>Type:</b>	String		
<b>Mandatory</b>		Summary	Complete
	Quantification	✓	✓
	Identification	✓	✓
<b>Example:</b>	MTD    description    An experiment investigating the effects of Il-6.		

## 6.2.7 sample\_processing[1-n]

<b>Description:</b>	A list of parameters describing a sample processing step. The order of the data_processing items should reflect the order these processing steps were performed in. If multiple parameters are given for a step these MUST be separated by a " ".		
<b>Type:</b>	Parameter List		
<b>Mandatory</b>		Summary	Complete
	Quantification		
	Identification		
<b>Example:</b>	MTD    sample_processing[1]    [SEP, SEP:00173, SDS PAGE,] MTD    sample_processing[2]    [SEP, SEP:00142, enzyme digestion,]  [MS, ... <span style="float: right;">MS:1001251, Trypsin, ]</span>		

## 6.2.8 instrument[1-n]-name

<b>Description:</b>	The name of the instrument used in the experiment. Multiple instruments are numbered 1..n.		
<b>Type:</b>	Parameter		
<b>Mandatory</b>		Summary	Complete
	Quantification		

	Identification		
<b>Example:</b>	MTD instrument[1]-name [MS, MS:1000449, LTQ Orbitrap,]		
	...		
	MTD instrument[2]-name [MS, MS:1000031, Instrument model, name of the instrument not included in the CV]		

### 6.2.9 instrument[1-n]-source

<b>Description:</b>	The instrument's source used in the experiment. Multiple instruments are numbered 1..n.		
<b>Type:</b>	Parameter		
<b>Mandatory</b>		Summary	Complete
	Quantification		
	Identification		
<b>Example:</b>	MTD instrument[1]-source [MS, MS:1000073, ESI,]		
	...		
	MTD instrument[2]-source [MS, MS:1000598, ETD,]		

### 6.2.10 instrument[1-n]-analyzer[1-n]

<b>Description:</b>	The instrument's analyzer type used in the experiment. Multiple instruments are enumerated 1..n.		
<b>Type:</b>	Parameter		
<b>Mandatory</b>		Summary	Complete
	Quantification		
	Identification		
<b>Example:</b>	MTD instrument[1]-analyzer[1] [MS, MS:1000291, linear ion trap,]		
	...		
	MTD instrument[2]-analyzer[1] [MS, MS:1000484, orbitrap,]		

### 6.2.11 instrument[1-n]-detector

<b>Description:</b>	The instrument's detector type used in the experiment. Multiple instruments are numbered 1..n.		
<b>Type:</b>	Parameter		
<b>Mandatory</b>		Summary	Complete
	Quantification		
	Identification		
<b>Example:</b>	MTD instrument[1]-detector [MS, MS:1000253, electron multiplier,]		
	...		
	MTD instrument[2]-detector [MS, MS:1000348, focal plane collector,]		

### 6.2.12 software[1-n]

<b>Description:</b>	Software used to analyze the data and obtain the reported results. The parameter's value SHOULD contain the software's version. The order (numbering) should reflect the order in which the tools were used.		
<b>Type:</b>	Parameter		
<b>Mandatory</b>		Summary	Complete
	Quantification		✓
	Identification		✓
<b>Example:</b>	MTD software[1] [MS, MS:1001207, Mascot, 2.3]		
	MTD software[2] [MS, MS:1001561, Scaffold, 1.0]		

### 6.2.13 software[1-n]-setting[1-n]

<b>Description:</b>	A software setting used. This field MAY occur multiple times for a single software. The value of this field is deliberately set as a String, since there currently do not exist cvParams for every possible setting.		
<b>Type:</b>	String		
<b>Mandatory</b>		Summary	Complete
	Quantification		
	Identification		
<b>Example:</b>	MTD software[1]-setting Fragment tolerance = 0.1 Da		
	MTD software[2]-setting Parent tolerance = 0.5 Da		

#### 6.2.14 protein\_search\_engine\_score[1-n]

<b>Description:</b>	The type of protein search engine score MUST be reported as a CV parameter [1-n]. The order of the search engine scores SHOULD reflect their importance for the identification and be used to determine the identification's rank.		
<b>Type:</b>	Parameter		
<b>Mandatory</b>		Summary	Complete
	Quantification	✓	✓
	Identification	✓	✓
<b>Example:</b>	MTD protein_search_engine_score[1] [MS, MS:1001171, Mascot:score,]		

#### 6.2.15 peptide\_search\_engine\_score[1-n]

<b>Description:</b>	The type of peptide search engine score MUST be reported as a CV parameter [1-n]. The order of the search engine scores SHOULD reflect their importance for the identification and be used to determine the identification's rank.		
<b>Type:</b>	Parameter		
<b>Mandatory</b>		Summary	Complete
	Quantification	✓	✓
	Identification	✓	✓
<b>Example:</b>	MTD peptide_search_engine_score[1] [MS, MS:1001171, Mascot:score,]		

#### 6.2.16 psm\_search\_engine\_score[1-n]

<b>Description:</b>	The type of psm search engine score MUST be reported as a CV parameter [1-n]. The order of the search engine scores SHOULD reflect their importance for the identification and be used to determine the identification's rank.		
<b>Type:</b>	Parameter		
<b>Mandatory</b>		Summary	Complete
	Quantification	✓	✓
	Identification	✓	✓
<b>Example:</b>	MTD psm_search_engine_score[2] [MS, MS:1001330, X!Tandem:expect,]		

#### 6.2.17 smallmolecule\_search\_engine\_score[1-n]

<b>Description:</b>	The type of small molecule search engine score MUST be reported as a CV parameter [1-n]. The order of the search engine scores SHOULD reflect their importance for the identification and be used to determine the identification's rank.		
<b>Type:</b>	Parameter		
<b>Mandatory</b>		Summary	Complete
	Quantification	✓	✓
	Identification	✓	✓
<b>Example:</b>	MTD smallmolecule_search_engine_score[1] [, , LipidDataAnalyzer,]		

#### 6.2.18 false\_discovery\_rate

<b>Description:</b>	The file's false discovery rate(s) reported at the PSM, peptide, and/or protein level. False Localization Rate (FLD) for the reporting of modifications can also be reported here. Multiple parameters MUST be separated by " ".		
<b>Type:</b>	Parameter List		
<b>Mandatory</b>		Summary	Complete
	Quantification		
	Identification		
<b>Example:</b>	MTD false_discovery_rate [MS, MS:1001364, pep:global FDR, 0.01] ... [MS, MS:1001214, prot:global FDR, 0.08]		

**6.2.19 publication[1-n]**

<b>Description:</b>	A publication associated with this file. Several publications can be given by indicating the number in the square brackets after "publication". PubMed ids must be prefixed by "pubmed:", DOIs by "doi:". Multiple identifiers MUST be separated by ",".		
<b>Type:</b>	String		
<b>Mandatory</b>		Summary	Complete
	Quantification		
	Identification		
<b>Example:</b>	MTD publication[1] pubmed:21063943 doi:10.1007/978-1-60761-987-1_6 MTD publication[2] pubmed:20615486 doi:10.1016/j.jprot.2010.06.008		

**6.2.20 contact[1-n]-name**

<b>Description:</b>	The contact's name. Several contacts can be given by indicating the number in the square brackets after "contact". A contact has to be supplied in the format [first name] [initials] [last name] (see example).		
<b>Type:</b>	String		
<b>Mandatory</b>		Summary	Complete
	Quantification		
	Identification		
<b>Example:</b>	MTD contact[1]-name James D. Watson ... MTD contact[2]-name Francis Crick		

**6.2.21 contact[1-n]-affiliation**

<b>Description:</b>	The contact's affiliation.		
<b>Type:</b>	String		
<b>Mandatory</b>		Summary	Complete
	Quantification		
	Identification		
<b>Example:</b>	MTD contact[1]-affiliation Cambridge University, UK MTD contact[2]-affiliation Cambridge University, UK		

**6.2.22 contact[1-n]-email**

<b>Description:</b>	The contact's e-mail address.		
<b>Type:</b>	String		
<b>Mandatory</b>		Summary	Complete
	Quantification		
	Identification		
<b>Example:</b>	MTD contact[1]-email watson@cam.ac.uk ... MTD contact[2]-email crick@cam.ac.uk		

**6.2.23 uri[1-n]**

<b>Description:</b>	A URI pointing to the file's source data (e.g., a PRIDE experiment or a PeptideAtlas build).		
<b>Type:</b>	URI		
<b>Mandatory</b>		Summary	Complete
	Quantification		
	Identification		
<b>Example:</b>	MTD uri[1] http://www.ebi.ac.uk/pride/url/to/experiment MTD uri[2] http://proteomecentral.proteomexchange.org/cgi/GetDataset		

**6.2.24 fixed\_mod[1-n]**

<b>Description:</b>	A parameter describing a fixed modifications searched for. Multiple fixed modifications are numbered 1..n. If no fixed modifications are searched, include the CV param MS:1002453: No fixed modifications searched.		
---------------------	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--	--



<b>Type:</b>	Parameter		
<b>Mandatory</b>		Summary	Complete
	Quantification	✓	✓
	Identification	✓	✓
<b>Example:</b>	MTD fixed_mod[1] [UNIMOD, UNIMOD:4, Carbamidomethyl, ] MTD fixed_mod[2] [UNIMOD, UNIMOD:35, Oxidation, ] MTD fixed_mod[3] [CHEMMOD, CHEMMOD:-18.0913, , ]		

#### 6.2.25 fixed\_mod[1-n]-site

<b>Description:</b>	A string describing a fixed modifications site. Following the unimod convention, modification site is a residue (e.g. "M"), terminus ("N-term" or "C-term") or both (e.g. "N-term Q" or "C-term K").		
<b>Type:</b>	String		
<b>Mandatory</b>		Summary	Complete
	Quantification		
	Identification		
<b>Example:</b>	MTD fixed_mod[1] [UNIMOD, UNIMOD:35, Oxidation, ] MTD fixed_mod[1]-site M ... MTD fixed_mod[2] [UNIMOD, UNIMOD:1, Acetyl, ] MTD fixed_mod[2]-site N-term ... MTD fixed_mod[3] [UNIMOD, UNIMOD:2, Amidated, ] MTD fixed_mod[3]-site C-term		

#### 6.2.26 fixed\_mod[1-n]-position

<b>Description:</b>	A string describing the term specificity of a fixed modification. Following the unimod convention, term specificity is denoted by the strings "Anywhere", "Any N-term", "Any C-term", "Protein N-term", "Protein C-term".		
<b>Type:</b>	String		
<b>Mandatory</b>		Summary	Complete
	Quantification		
	Identification		
<b>Example:</b>	MTD fixed_mod[1] [UNIMOD, UNIMOD:35, Oxidation, ] MTD fixed_mod[1]-site M ... MTD fixed_mod[2] [UNIMOD, UNIMOD:1, Acetyl, ] MTD fixed_mod[2]-site N-term MTD fixed_mod[2]-position Protein N-term ... MTD fixed_mod[3] [UNIMOD, UNIMOD:2, Amidated, ] MTD fixed_mod[3]-site C-term MTD fixed_mod[3]-position Protein C-term		

#### 6.2.27 variable\_mod[1-n]

<b>Description:</b>	A parameter describing a variable modification searched for. Multiple variable modifications are numbered 1.. n. If no variable modifications were searched, include the CV param MS:1002454: No variable modifications searched.		
<b>Type:</b>	Parameter		
<b>Mandatory</b>		Summary	Complete
	Quantification	✓	✓
	Identification	✓	✓
<b>Example:</b>	MTD variable_mod[1] [UNIMOD, UNIMOD:21, Phospho, ] MTD variable_mod[2] [UNIMOD, UNIMOD:35, Oxidation, ] MTD variable_mod[3] [CHEMMOD, CHEMMOD:-18.0913, , ]		

#### 6.2.28 variable\_mod[1-n]-site

<b>Description:</b>	A string describing a variable modifications site. Following the unimod convention, modification site is a residue (e.g. "M"), terminus ("N-term" or "C-term") or both (e.g. "N-term Q" or "C-term K").		
---------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--	--

<b>Type:</b>	String		
<b>Mandatory</b>		Summary	Complete
	Quantification		
<b>Example:</b>	Identification		
	MTD variable_mod[1] [UNIMOD, UNIMOD:35, Oxidation, ] MTD variable_mod[1]-site M ... MTD variable_mod[2] [UNIMOD, UNIMOD:1, Acetyl, ] MTD variable_mod[2]-site N-term ... MTD variable_mod[3] [UNIMOD, UNIMOD:2, Amidated, ] MTD variable_mod[3]-site C-term		

#### 6.2.29 variable\_mod[1-n]-position

<b>Description:</b>	A string describing the term specificity of a variable modification. Following the unimod convention, term specificity is denoted by the strings “Anywhere”, “Any N-term”, “Any C-term”, “Protein N-term”, “Protein C-term”.		
<b>Type:</b>	String		
<b>Mandatory</b>		Summary	Complete
	Quantification		
<b>Example:</b>	Identification		
	MTD variable_mod[1] [UNIMOD, UNIMOD:35, Oxidation, ] MTD variable_mod[1]-site M ... MTD variable_mod[2] [UNIMOD, UNIMOD:1, Acetyl, ] MTD variable_mod[2]-site N-term MTD variable_mod[2]-position Protein N-term ... MTD variable_mod[3] [UNIMOD, UNIMOD:2, Amidated, ] MTD variable_mod[3]-site C-term MTD variable_mod[3]-position Protein C-term		

#### 6.2.30 quantification\_method

<b>Description:</b>	The quantification method used in the experiment reported in the file.		
<b>Type:</b>	Parameter		
<b>Mandatory</b>		Summary	Complete
	Quantification		✓
<b>Example:</b>	Identification		
	MTD quantification_method [MS, MS:1001837, iTRAQ quantitation analysis, ]		

#### 6.2.31 protein-quantification\_unit

<b>Description:</b>	Defines what type of units is reported in the protein quantification fields.		
<b>Type:</b>	Parameter		
<b>Mandatory</b>		Summary	Complete
	Quantification	(✓) <sup>1</sup>	(✓) <sup>1</sup>
<b>Example:</b>	Identification		
	MTD protein-quantification_unit [PRIDE, PRIDE:0000395, Ratio, ] <small><sup>1</sup>mandatory if protein section is present</small>		

#### 6.2.32 peptide-quantification\_unit

<b>Description:</b>	Defines what type of units is reported in the peptide quantification fields.		
<b>Type:</b>	Parameter		
<b>Mandatory</b>		Summary	Complete
	Quantification	(✓) <sup>1</sup>	(✓) <sup>1</sup>
<b>Example:</b>	Identification		
	MTD peptide-quantification_unit [PRIDE, PRIDE:0000395, Ratio, ] <small><sup>1</sup>mandatory if peptide section is present</small>		

#### 6.2.33 small\_molecule-quantification\_unit

<b>Description:</b>	Defines what type of units is reported in the small molecule quantification		
---------------------	-----------------------------------------------------------------------------	--	--

	fields.		
Type:	Parameter		
Mandatory		Summary	Complete
	Quantification	(✓) <sup>1</sup>	(✓) <sup>1</sup>
	Identification		
	<sup>1</sup> mandatory if small molecule section is present		
Example:	MTD small_molecule-quantification_unit [PRIDE, PRIDE:0000395, Ratio, ]		

## 6.2.34 ms\_run[1-n]-format

Description:	A parameter specifying the data format of the external MS data file. If ms_run[1-n]-format is present, ms_run[1-n]-id_format SHOULD also be present.		
Type:	Parameter		
Mandatory		Summary	Complete
	Quantification		
	Identification		
Example:	MTD ms_run[1]-format [MS, MS:1000584, mzML file, ] MTD ms_run[1]-id_format [MS, MS:1000530, mzML unique identifier, ] ... MTD ms_run[2]-format [MS, MS:1001062, Mascot MGF file, ] MTD ms_run[2]-id_format [MS, MS:1000774, multiple peak list nativeID format, ]		

## 6.2.35 ms\_run[1-n]-location

<b>Description:</b>	Location of the external data file. If the actual location of the MS run is unknown, a “null” MUST be used as a place holder value.		
<b>Type:</b>	URL		
<b>Mandatory</b>		Summary	Complete
	Quantification	✓	✓
	Identification	✓	✓
<b>Example:</b>	MTD ms_run_location[1] file://C:\path\to\my\file ... MTD ms_run_location[2] <a href="ftp://ftp.ebi.ac.uk/path/to/file">ftp://ftp.ebi.ac.uk/path/to/file</a>		

## 6.2.36 ms\_run[1-n]-id\_format

Description:	Parameter specifying the id format used in the external data file. If ms_run[1-n]-id_format is present, ms_run[1-n]-format SHOULD also be present.		
Type:	Parameter		
Mandatory		Summary	Complete
	Quantification		
	Identification		
Example:	MTD ms_run[1]-format [MS, MS:1000584, mzML file, ] MTD ms_run[1]-id_format [MS, MS:1000530, mzML unique identifier, ] ... MTD ms_run[2]-format [MS, MS:1001062, Mascot MGF file, ] MTD ms_run[2]-id_format [MS, MS:1000774, multiple peak list nativeID format, ]		

## 6.2.37 ms\_run[1-n]-fragmentation\_method

<b>Description:</b>	A list of “ ” separated parameters describing all the types of fragmentation used in a given ms run.		
<b>Type:</b>	Parameter List		
<b>Mandatory</b>		Summary	Complete
	Quantification		
	Identification		
<b>Example:</b>	MTD ms_run[1]-fragmentation_method [MS, MS:1000133, CID, ] ... MTD ms_run[2]-fragmentation_method [MS, MS:1000422, HCD ..., ]		

## 6.2.38 ms\_run[1-n]-hash

<b>Description:</b>	Hash value of the corresponding external MS data file defined in ms_run[1-n]-
---------------------	-------------------------------------------------------------------------------

	location. If ms_run[1-n]-hash is present, ms_run[1-n]-hash_method SHOULD also be present.		
<b>Type:</b>	String		
<b>Mandatory</b>		Summary	Complete
	Quantification Identification		
<b>Example:</b>	<pre>MTD ms_run[1]-hash_method [MS, MS: MS:1000569, SHA-1, ] MTD ms_run[1]-hash de9f2c7fd25e1b3afad3e85a0bd17d9b100db4b3</pre>		

#### 6.2.39 ms\_run[1-n]-hash\_method

<b>Description:</b>	A parameter specifying the hash methods used to generate the String in ms_run[1-n]-hash. Specifics of the hash method used MAY follow the definitions of the mzML format. If ms_run[1-n]-hash is present, ms_run[1-n]-hash_method SHOULD also be present.		
<b>Type:</b>	Parameter		
<b>Mandatory</b>		Summary	Complete
	Quantification Identification		
<b>Example:</b>	<pre>MTD ms_run[1]-hash_method [MS, MS: MS:1000569, SHA-1, ] MTD ms_run[1]-hash de9f2c7fd25e1b3afad3e85a0bd17d9b100db4b3</pre>		

#### 6.2.40 custom[1-n]

<b>Description:</b>	Any additional parameters describing the analysis reported.		
<b>Type:</b>	Parameter		
<b>Mandatory</b>		Summary	Complete
	Quantification Identification		
<b>Example:</b>	<pre>MTD custom[1] [,MS operator, Florian]</pre>		

#### 6.2.41 sample[1-n]-species[1-n]

<b>Description:</b>	The respective species of the samples analysed.		
<b>Type:</b>	Parameter		
<b>Mandatory</b>		Summary	Complete
	Quantification Identification		
<b>Example:</b>	<pre>COM Experiment where all samples consisted of the same two species MTD sample[1]-species[1] [NEWT, 9606, Homo sapiens (Human), ] MTD sample[2]-species[1] [NEWT, 12059, Rhinovirus, ]</pre>		
	<pre>COM Experiment where different two samples from different species (combinations) COM were analysed as biological replicates.  MTD sample[1]-species[1] [NEWT, 9606, Homo sapiens (Human), ] MTD sample[1]-species[2] [NEWT, 573824, Human rhinovirus 1, ] MTD sample[2]-species[1] [NEWT, 9606, Homo sapiens (Human), ] MTD sample[2]-species[2] [NEWT, 12130, Human rhinovirus 2, ]</pre>		

#### 6.2.42 sample[1-n]-tissue[1-n]

<b>Description:</b>	The respective tissue(s) of the sample.		
<b>Type:</b>	Parameter		
<b>Mandatory</b>		Summary	Complete
	Quantification Identification		
<b>Example:</b>	<pre>MTD sample[1]-tissue[1] [BTO, BTO:0000759, liver, ]</pre>		

#### 6.2.43 sample[1-n]-cell\_type[1-n]

<b>Description:</b>	The respective cell type(s) of the sample.		
---------------------	--------------------------------------------	--	--

<b>Type:</b>	Parameter
<b>Mandatory</b>	Summary
	Complete
	Quantification
<b>Example:</b>	Identification
	MTD sample[1]-cell_type[1] [CL, CL:0000182, hepatocyte, ]

**6.2.44 sample[1-n]-disease[1-n]**

<b>Description:</b>	The respective disease(s) of the sample.
<b>Type:</b>	Parameter
<b>Mandatory</b>	Summary
	Complete
	Quantification
<b>Example:</b>	Identification
	MTD sample[1]-disease[1] [DOID, DOID:684, hepatocellular carcinoma, ] MTD sample[1]-disease[2] [DOID, DOID:9451, alcoholic fatty liver, ]

**6.2.45 sample[1-n]-description**

<b>Description:</b>	A human readable description of the sample.
<b>Type:</b>	String
<b>Mandatory</b>	Summary
	Complete
	Quantification
<b>Example:</b>	Identification
	MTD sample[1]-description Hepatocellular carcinoma samples. MTD sample[2]-description Healthy control samples.

**6.2.46 sample[1-n]-custom[1-n]**

<b>Description:</b>	Parameters describing the sample's additional properties.
<b>Type:</b>	Parameter
<b>Mandatory</b>	Summary
	Complete
	Quantification
<b>Example:</b>	Identification
	MTD sample[1]-custom[1] [,,Extraction date, 2011-12-21] MTD sample[1]-custom[2] [,,Extraction reason, liver biopsy]

**6.2.47 assay[1-n]-quantification\_reagent**

<b>Description:</b>	The reagent used to label the sample in the assay. For label-free analyses the "unlabeled sample" CV term SHOULD be used. For the "light" channel in label-based experiments the appropriate CV term specifying the labelling channel should be used.
<b>Type:</b>	Parameter
<b>Mandatory</b>	Summary
	Complete
	Quantification
<b>Example:</b>	Identification
	MTD assay[1]-quantification_reagent [PRIDE,PRIDE:0000114,iTRAQ reagent,114] MTD assay[2]-quantification_reagent [PRIDE,PRIDE:0000115,iTRAQ reagent,115]  OR MTD assay[1]-quantification_reagent [MS,MS:1002038,unlabeled sample,]  OR MTD assay[1]-quantification_reagent [PRIDE, PRIDE:0000326, SILAC light] MTD assay[2]-quantification_reagent [PRIDE, PRIDE:0000325, SILAC heavy]

**6.2.48 assay[1-n]-quantification\_mod[1-n]**

<b>Description:</b>	A parameter describing a modification associated with a quantification_reagent. Multiple modifications are numbered 1..n.
---------------------	---------------------------------------------------------------------------------------------------------------------------

Type:	Parameter		
Mandatory		Summary	Complete
	Quantification		
	Identification	1	1
	not recommended for identification only files		
Example:	MTD assay[2]-quantification_mod[1] [UNIMOD, UNIMOD:188, Label:13C(6), ]		

## 6.2.49 assay[1-n]-quantification\_mod[1-n]-site

<b>Description:</b>	A string describing the modifications site. Following the unimod convention, modification site is a residue (e.g. “M”), terminus (“N-term” or “C-term”) or both (e.g. “N-term Q” or “C-term K”).		
<b>Type:</b>	String		
<b>Mandatory</b>		Summary	Complete
	Quantification		
	Identification	1	1
	not recommended for identification only files		
<b>Example:</b>	MTD assay[2]-quantification_mod[1] [UNIMOD, UNIMOD:188, Label:13C(6), ] MTD assay[2]-quantification_mod[2] [UNIMOD, UNIMOD:188, Label:13C(6), ] MTD assay[2]-quantification_mod[1]-site R MTD assay[2]-quantification_mod[2]-site K		

## 6.2.50 assay[1-n]-quantification\_mod[1-n]-position

Description:	A string describing the term specificity of the modification. Following the unimod convention, term specificity is denoted by the strings “Anywhere”, “Any N-term”, “Any C-term”, “Protein N-term”, “Protein C-term”.		
Type:	String		
Mandatory		Summary	Complete
	Quantification		
	Identification	1	1
	not recommended for identification only files		
Example:	MTD	assay[2]-quantification_mod[1]	[UNIMOD, UNIMOD:188, Label:13C(6), ]
	MTD	assay[2]-quantification_mod[2]	[UNIMOD, UNIMOD:188, Label:13C(6), ]
	MTD	assay[2]-quantification_mod[1]-site	R
	MTD	assay[2]-quantification_mod[2]-site	K
	MTD	assay[2]-quantification_mod[1]-position	Anywhere
	MTD	assay[2]-quantification_mod[2]-position	Anywhere

## 6.2.51 assay[1-n]-sample\_ref

Description:	An association from a given assay to the sample analysed.		
Type:	{SAMPLE_ID}		
Mandatory		Summary	Complete
	Quantification		
	Identification		
Example:	MTD assay[1]-sample_ref	sample[1]	
	MTD assav[2]-sample_ref	sample[2]	

## 6.2.52 assay[1-n]-ms\_run\_ref

Description:	An association from a given assay to the source MS run.		
Type:	{MS_RUN_ID}		
Mandatory		Summary	Complete
	Quantification	(✓) <sup>1</sup>	✓
	Identification	(✓) <sup>1</sup>	(✓) <sup>1</sup>
	1 mandatory if assays are reported		
Example:	MTD      assay[1]-ms_run_ref      ms_run[1]		

## 6.2.53 study\_variable[1-n]-assay\_refs

<b>Description:</b>	Comma-separated references to the IDs of assays grouped in the study variable.
---------------------	--------------------------------------------------------------------------------



Type:	{ASSAY_ID}, ...		
Mandatory		Summary	Complete
	Quantification	(✓) <sup>1</sup>	✓
	Identification		
	1 mandatory if both assays and study variables are reported		
Example:	MTD      study_variable[1]-assay_refs    assay[1], assay[2], assay[3]		

## 6.2.54 study\_variable[1-n]-sample\_refs

Description:	Comma-separated references to the samples that were analysed in the study variable.		
Type:	{SAMPLE_ID}, ... {SAMPLE_ID}		
Mandatory		Summary	Complete
	Quantification		
	Identification		
Example:	MTD        study_variable[1]-sample_refs    sample[1]		

## 6.2.55 study\_variable[1-n]-description

Description:	A textual description of the study variable.		
Type:	String		
Mandatory		Summary	Complete
	Quantification	(✓) <sup>1</sup>	✓
	Identification	(✓) <sup>1</sup>	(✓) <sup>1</sup>
	<sup>1</sup> mandatory of study variables reported		
Example:	MTD      study_variable[1]-description Group B (spike-in 0.74 fmol/uL)		

## 6.2.56 cv[1-n]-label

<b>Description:</b>	A string describing the labels of the controlled vocabularies/ontologies used in the mzTab file											
<b>Type:</b>	String											
<b>Mandatory</b>	<table><tr><td></td><td>Summary</td><td>Complete</td></tr><tr><td>Quantification</td><td></td><td></td></tr><tr><td>Identification</td><td></td><td></td></tr></table>		Summary	Complete	Quantification			Identification				
	Summary	Complete										
Quantification												
Identification												
<b>Example:</b>	MTD    cv[1]-label    MS ...											

## 6.2.57 cv[1-n]-full\_name

<b>Description:</b>	A string describing the full names of the controlled vocabularies/ontologies used in the mzTab file											
<b>Type:</b>	String											
<b>Mandatory</b>	<table><tr><td></td><td>Summary</td><td>Complete</td></tr><tr><td>Quantification</td><td></td><td></td></tr><tr><td>Identification</td><td></td><td></td></tr></table>		Summary	Complete	Quantification			Identification				
	Summary	Complete										
Quantification												
Identification												
<b>Example:</b>	MTD    cv[1]-full_name    MS ...											

## 6.2.58 cv[1-n]-version

<b>Description:</b>	A string describing the version of the controlled vocabularies/ontologies used in the mzTab file											
<b>Type:</b>	String											
<b>Mandatory</b>	<table><tr><td></td><td>Summary</td><td>Complete</td></tr><tr><td>Quantification</td><td></td><td></td></tr><tr><td>Identification</td><td></td><td></td></tr></table>		Summary	Complete	Quantification			Identification				
	Summary	Complete										
Quantification												
Identification												
<b>Example:</b>	MTD    cv[1]-version    3.54.0 ...											

## 6.2.59 cv[1-n]-url

Description:	A string containing the URLs of the controlled vocabularies/ontologies used in the mzTab file		
Type:	String		
Mandatory		Summary	Complete
	Quantification		
	Identification		
Example:	MTD cv[1]-url <a href="http://psidev.cvs.sourceforge.net/viewvc/psidev/psi/psi-ms/mzML/controlledVocabulary/psi-ms.obo">http://psidev.cvs.sourceforge.net/viewvc/psidev/psi/psi-ms/mzML/controlledVocabulary/psi-ms.obo</a> ...		

## 6.2.60 colunit-protein

<b>Description:</b>	Defines the unit for the data reported in a column of the protein section. The format of the value has to be {column name}={Parameter defining the unit} This field MUST NOT be used to define a unit for quantification columns. The unit used for protein quantification values MUST be set in <i>protein-quantification_unit</i> .											
<b>Type:</b>	String											
<b>Mandatory</b>	<table><tr><td></td><td>Summary</td><td>Complete</td></tr><tr><td>Quantification</td><td></td><td></td></tr><tr><td>Identification</td><td></td><td></td></tr></table>		Summary	Complete	Quantification			Identification				
	Summary	Complete										
Quantification												
Identification												
<b>Example:</b>	MTD    colunit-protein molecular_mass=[UO, UO:0000222, kilodalton,]											

## 6.2.61 colunit-peptide

<b>Description:</b>	Defines the used unit for a column in the peptide section. The format of the value has to be {column name}={Parameter defining the unit} This field MUST NOT be used to define a unit for quantification columns. The unit used for peptide quantification values MUST be set in peptide-quantification_unit.											
<b>Type:</b>	String											
<b>Mandatory</b>	<table><tr><td></td><td>Summary</td><td>Complete</td></tr><tr><td>Quantification</td><td></td><td></td></tr><tr><td>Identification</td><td></td><td></td></tr></table>		Summary	Complete	Quantification			Identification				
	Summary	Complete										
Quantification												
Identification												
<b>Example:</b>	MTD      colunit-peptide retention_time=[UO,UO:0000031, minute,]											

## 6.2.62 colunit-psm

<b>Description:</b>	Defines the used unit for a column in the PSM section. The format of the value has to be {column name}={Parameter defining the unit} This field MUST NOT be used to define a unit for quantification columns. The unit used for peptide quantification values MUST be set in peptide-quantification_unit.											
<b>Type:</b>	String											
<b>Mandatory</b>	<table><tr><td></td><td>Summary</td><td>Complete</td></tr><tr><td>Quantification</td><td></td><td></td></tr><tr><td>Identification</td><td></td><td></td></tr></table>		Summary	Complete	Quantification			Identification				
	Summary	Complete										
Quantification												
Identification												
<b>Example:</b>	MTD      colunit-psm retention_time=[UO,UO:0000031, minute,]											

## 6.2.63 colunit-small\_molecule

<b>Description:</b>	Defines the used unit for a column in the small molecule section. The format of the value has to be {column name}={Parameter defining the unit}		
---------------------	-------------------------------------------------------------------------------------------------------------------------------------------------	--	--

	This field <b>MUST NOT</b> be used to define a unit for quantification columns. The unit used for small molecule quantification values <b>MUST</b> be set in <code>small_molecule-quantification_unit</code> .		
<b>Type:</b>	String		
<b>Mandatory</b>		Summary	Complete
	Quantification		
	Identification		
<b>Example:</b>	MTD colunit-small_molecule retention_time=[UO,UO:0000031, minute,]		

### 6.3 Protein Section

The protein section is table-based. The protein section **MUST** always come after the metadata section. All table columns **MUST** be tab-separated. There **MUST NOT** be any empty cells. Missing values **MUST** be reported using “null”. Most columns are mandatory. The order of columns is not specified although for ease of human interpretation, it is **RECOMMENDED** to follow the order specified below.

#### 6.3.1 accession

<b>Description:</b>	The accession of the protein in the source database. A protein accession <b>MUST</b> be unique within one mzTab file. If different quantification values are required for the same underlying accession, for example if differentially modified forms of a protein have been quantified, a the suffix [1-n] <b>SHOULD</b> be appended to the accession e.g. P12345[1], P12345[2].		
<b>Type:</b>	String		
<b>Mandatory</b>		Summary	Complete
	Quantification	✓	✓
	Identification	✓	✓
<b>Example:</b>	PRH accession ... PRT P12345 ... PRT P12346 ...		

#### 6.3.2 description

<b>Description:</b>	The protein's name and or description line.		
<b>Type:</b>	String		
<b>Mandatory</b>		Summary	Complete
	Quantification	✓	✓
	Identification	✓	✓
<b>Example:</b>	PRH accession description ... PRT P12345 Aspartate aminotransferase, mitochondrial ... PRT P12346 Serotransferrin ...		

#### 6.3.3 taxid

<b>Description:</b>	The NCBI/NEWT taxonomy id for the species the protein was identified in.		
<b>Type:</b>	Integer		
<b>Mandatory</b>		Summary	Complete
	Quantification	✓	✓
	Identification	✓	✓
<b>Example:</b>	PRH accession ... taxid ... PRT P12345 ... 10116 ... PRT P12346 ... 10116 ...		

#### 6.3.4 species

<b>Description:</b>	The human readable species the protein was identified in - this <b>SHOULD</b> be the NCBI entry's name.		
<b>Type:</b>	String		
<b>Mandatory</b>		Summary	Complete

<b>Example:</b>	Quantification				✓	✓
	Identification				✓	✓
	PRH	accession	...	taxid	species	...
	PRT	P12345	...	10116	Rattus norvegicus (Rat)	...
	PRT	P12346	...	10116	Rattus norvegicus (Rat)	...

### 6.3.5 database

<b>Description:</b>	The protein database used for the search (could theoretically come from a different species). Wherever possible the Miriam ( <a href="http://www.ebi.ac.uk/miriam">http://www.ebi.ac.uk/miriam</a> ) assigned name SHOULD be used.					
<b>Type:</b>	String					
<b>Mandatory</b>			Summary		Complete	
	Quantification		✓		✓	
	Identification		✓		✓	
<b>Example:</b>	PRH	accession	...	taxid	species	database
	PRT	P12345	...	10116	Rattus norvegicus (Rat)	UniProtKB
	PRT	P12346	...	10116	Rattus norvegicus (Rat)	UniProtKB

### 6.3.6 database\_version

<b>Description:</b>	The protein database's version – in case there is no version available (custom build) the creation / download (e.g., for NCBI nr) date SHOULD be given. Additionally, the number of entries in the database MAY be reported in round brackets after the version in the format: {version} ({#entries} entries), for example “2011-11 (1234 entries)”.					
<b>Type:</b>	String					
<b>Mandatory</b>			Summary		Complete	
	Quantification		✓		✓	
	Identification		✓		✓	
<b>Example:</b>	PRH	accession	...	taxid	species	database database_version
	PRT	P12345	...	10116	Rattus norvegicus (Rat)	UniProtKB 2011_11
	PRT	P12346	...	10116	Rattus norvegicus (Rat)	UniProtKB 2011_11

### 6.3.7 search\_engine

Description:	A “ ” delimited list of search engine(s) that identified this protein. Search engines MUST be supplied as parameters.					
Type:	Parameter List					
Mandatory		Summary	Complete			
	Quantification	✓	✓			
	Identification	✓	✓			
Example:	COM	In this example the first protein was identified by Mascot and X!Tandem while				
	COM	the second protein was only identified by Mascot.				
	PRH	accession	...	search_engine	...	...
	PRT	P12345	...	[MS,MS:1001207,Mascot,]   [MS,MS:1001476,X!Tandem,]	...	...
	PRT	P12346	...	[MS,MS:1001207,Mascot,]	...	...

### 6.3.8 best\_search\_engine\_score[1-n]

Description:	The best search engine score (for this type of score) for the given protein across all replicates reported. The type of score MUST be defined in the metadata section. If the protein was not identified by the specified search engine “null” must be reported.					
Type:	Double					
Mandatory			Summary	Complete		
	Quantification		✓	✓		
	Identification		✓	✓		
Example:	MTD	protein_search_engine_score[1]		[MS,MS:1001171,Mascot score,]		
	...					
	PRH	accession	...	best_search_engine_score[1]	...	
	...					
	PRT	P12345	...	50		...
	PRT	P12346	...	36...		

COM Protein P12346 was only identified using Mascot and not X!Tandem

### 6.3.9 search\_engine\_score[1-n]\_ms\_run[1-n]

<b>Description:</b>	The search engine score for the given protein in the defined ms run. The type of score MUST be defined in the metadata section. If the protein was not identified by the specified search engine "null" must be reported.		
<b>Type:</b>	Parameter List		
<b>Mandatory</b>		Summary	Complete
	Quantification		✓
	Identification		✓
<b>Example:</b>	<pre> MTD protein_search_engine_score[1] [MS,MS:1001171,Mascot score,] ... PRH accession ... search_engine_score[1]_ms_run[1] PRT P12345 ... 50 PRT P12346 ... 36 </pre>		

### 6.3.10 reliability

<b>Description:</b>	The reliability of the given protein identification. This must be supplied by the resource and has to be one of the following values: 1: high reliability 2: medium reliability 3: poor reliability Important: An identification's reliability is resource-dependent.		
<b>Type:</b>	Integer		
<b>Mandatory</b>		Summary	Complete
	Quantification		
	Identification		
<b>Example:</b>	<pre> PRH accession ... reliability ... PRT P12345 ... 2 ... PRT P12346 ... 1 ... </pre>		

### 6.3.11 num\_psms\_ms\_run[1-n]

<b>Description:</b>	The count of the total significant PSMs that can be mapped to the reported protein.		
<b>Type:</b>	Integer		
<b>Mandatory</b>		Summary	Complete
	Quantification		
	Identification		✓
<b>Example:</b>	<pre> COM P12345 is identified through ABCM, ABCM+Oxidation, CDE, CDE ... PRH accession ... num_psms_ms_run[1] ... PRT P12345 ... 4 ... </pre>		

### 6.3.12 num\_peptides\_distinct\_ms\_run[1-n]

<b>Description:</b>	The count of the number of different peptide sequences that have been identified above the significance threshold. Different modifications or charge states of the same peptide are not counted.		
<b>Type:</b>	Integer		
<b>Mandatory</b>		Summary	Complete
	Quantification		
	Identification		✓
<b>Example:</b>	<pre> COM P12345 is identified through ABCM, ABCM+Oxidation, CDE, CDE ... PRH accession ... num_peptides_distinct_ms_run[1] ... PRT P12345 ... 3 ... </pre>		

### 6.3.13 num\_peptides\_unique\_ms\_run[1-n]

<b>Description:</b>	The number of peptides that can be mapped uniquely to the protein reported. If ambiguity members have been reported, the count MUST be derived from		
---------------------	-----------------------------------------------------------------------------------------------------------------------------------------------------	--	--

	the number of peptides that can be uniquely mapped to the group of accessions, since the assumption is that these accessions are supported by the same evidence.		
<b>Type:</b>	Integer		
<b>Mandatory</b>		Summary	Complete
	Quantification		
	Identification		✓
<b>Example:</b>	<p>COM P12345 is identified through ABCM, ABCM+Oxidation, CDE, CDE</p> <p>COM ABCM is only from P12345, CDE from P12345 and P12346</p> <p>...</p> <p>PRH accession ... num_peptides_unique_ms_run[1] ...</p> <p>PRT P12345 ... 2 ...</p>		

#### 6.3.14 ambiguity\_members

<b>Description:</b>	<p>A comma-delimited list of protein accessions. This field should be set in the representative protein of the ambiguity group (the protein identified through the accession in the first column). The accessions listed in this field should identify proteins that could also be identified through these peptides (e.g. “same-set proteins”) but were not chosen by the researcher or resource, often for arbitrary reasons. It is NOT RECOMMENDED to report subset proteins as ambiguity_members, since the proteins reported here, together with the representative protein are taken to be a group that cannot be separated based on the peptide evidence.</p>		
<b>Type:</b>	String List		
<b>Mandatory</b>		Summary	Complete
	Quantification	✓	✓
	Identification	✓	✓
<b>Example:</b>	<p>COM P12345, P12347, and P12348 can all be identified through the same peptides</p> <p>...</p> <p>PRH accession ... ambiguity_members ...</p> <p>PRT P12345 ... P12347,P12348 ...</p>		

#### 6.3.15 modifications

<b>Description:</b>	<p>In contrast to the PSM section, fixed modifications or modifications caused by the quantification reagent (i.e. the SILAC/iTRAQ label) SHOULD NOT be reported in this column.</p> <p>Column entries are a comma delimited list of modifications found in the given protein. Modifications have to be reported in the following format: {position in protein}{parameter}-{modification or substitution identifier}{neutral loss}]</p> <p>Modification location scores cannot be supplied at the Protein level. Furthermore, in case a position is unknown no position information MAY be supplied.</p> <p>Terminal modifications MUST be reported at position 0 or protein size + 1 respectively.</p> <p>Valid modification identifiers are either PSI-MOD or UNIMOD accession (including the “MOD:” / “UNIMOD:” prefix) or CHEMMODS. CHEMMODS have the format CHEMMOD:+/-{chemical formula or <i>m/z</i> delta}. Valid CHEMMODS are for example “CHEMMOD:+NH4” or “CHEMMOD:-10.1098”. CHEMMODS MUST NOT be used if the modification can be reported using a PSI-MOD or UNIMOD accession. Mass deltas MUST NOT be used for CHEMMODS if the delta can be expressed through a known chemical formula.</p> <p>Neutral losses MAY be reported as cvParams. If a neutral loss is not associated with an existing modification it is reported as separated comma-</p>		
---------------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--	--



	<p>separated entry.</p> <p>Additionally, it is possible to report substitutions of amino acids using SUBST:{amino acid}.</p> <p>If different modifications are identified from different ms_runs, a superset of the identified modifications SHOULD be reported here. Detailed modification mapping to individual ms_runs is provided through the PSM table.</p> <p>If protein level modifications are not reported, a “null” MUST be used. If protein level modifications are reported but not present on a given protein, a “0” MUST be reported.</p>											
Type:	String											
Mandatory	<table><tr><td></td><td>Summary</td><td>Complete</td></tr><tr><td>Quantification</td><td>✓</td><td>✓</td></tr><tr><td>Identification</td><td>✓</td><td>✓</td></tr></table>				Summary	Complete	Quantification	✓	✓	Identification	✓	✓
	Summary	Complete										
Quantification	✓	✓										
Identification	✓	✓										
Example:	<p>COM Protein P12345 TESTPEPTIDES with 2 phosphorylation sites: TEpSTPEpPTIDES</p> <p>COM Common use cases without score:</p> <p>COM Example 1: Both locations have been determined</p> <p>PRH accession ... modifications ...</p> <p>PRT P12345 ... 3-MOD:00412,8-MOD:00412 ...</p> <p>COM Example 2: Like Ex. 1, but first site localization is ambiguous (S or T)</p> <p>PRH accession ... modifications ...</p> <p>PRT P12345 ... 3 4-MOD:00412,8-MOD:00412 ...</p> <p>COM Example 3: Protein only known to contain two phosphor sites in the range 3 to 8</p> <p>PRH accession ... modifications ...</p> <p>PRT P12345 ... 3 4 8-MOD:00412, 3 4 8-MOD:00412 ...</p> <p>COM Example 4: No position information or only accurate mass available</p> <p>PRH accession ... modifications ...</p> <p>PRT P12345 ... CHEMMOD:+159.93 ...</p> <p>COM Common use cases with probability scores:</p> <p>COM Example 5: MOD:00412 with associated probabilities at position 3 and 4 and a probability of 0.3 at position 8</p> <p>PRH accession ... modifications ...</p> <p>PRT P12345 ... 3[MS,MS:1001876, modification probability, 0.8] 4[MS,MS:1001876, modification probability, 0.2]-MOD:00412,8[MS,MS:1001876, modification probability, 0.3]-MOD:00412 ...</p> <p>COM Reporting substitutions</p> <p>COM Example 6: Substitution of amino acid at position 3 with R (Original sequence is reported in sequence column)</p> <p>PRH accession ... modifications ...</p> <p>PRT P12345 ... 3-SUBST:R</p> <p>COM Example 7: Modification with an associated neutral loss</p> <p>PRH accession ... modifications ...</p> <p>PRT P12345 ... 3-UNIMOD:21,3-[MS, MS:1001524, fragment neutral loss, 63.998285]</p>											

### 6.3.16 uri

<b>Description:</b>	A URI pointing to the protein's source entry in the unit it was identified in (e.g., the PRIDE database or a local database / file identifier).				
<b>Type:</b>	URI				
<b>Mandatory</b>		Summary	Complete		
	Quantification				
	Identification				
<b>Example:</b>	PRT	accession	...	uri	...
	PRH	P12345	...	<a href="http://www.ebi.ac.uk/pride/url/to/P12345">http://www.ebi.ac.uk/pride/url/to/P12345</a>	...

## 6.3.17 go\_terms

<b>Description:</b>	A ' ' delimited list of GO accessions for this protein.		
<b>Type:</b>	String List		
<b>Mandatory</b>		Summary	Complete
	Quantification		
	Identification		
<b>Example:</b>	PRT accession ... go_terms PRH P12345 ... GO:0006457 GO:0005759 GO:0005886 GO:0004069 ...		

## 6.3.18 protein\_coverage

<b>Description:</b>	A value between 0 and 1 defining the protein coverage.		
<b>Type:</b>	Double		
<b>Mandatory</b>		Summary	Complete
	Quantification		✓
	Identification		✓
<b>Example:</b>	PRT accession ... protein_coverage ... PRH P12345 ... 0.4 ...		

## 6.3.19 protein\_abundance\_assay[1-n]

<b>Description:</b>	The protein's abundance as measured in the given assay through whatever technique was employed.		
<b>Type:</b>	Double		
<b>Mandatory</b>		Summary	Complete
	Quantification	(✓) <sup>1</sup>	✓
	Identification		
<sup>1</sup> mandatory if quantification data is provided for assays			
<b>Example:</b>	PRT accession ... protein_abundance_assay[1] ... protein_abundance_assay[2] ... PRH P12345 ... 0.4 ... 0.2 ...		

## 6.3.20 protein\_abundance\_study\_variable[1-n]

<b>Description:</b>	The protein's abundance as measured in the given Study Variable, for example mean or median of quantitative values reported in Assays.		
<b>Type:</b>	Double		
<b>Mandatory</b>		Summary	Complete
	Quantification	✓	✓
	Identification		
<b>Example:</b>	PRT accession ... protein_abundance_study_variable[1] ... protein_abundance_study_variable[2] ... PRH P12345 ... 0.4 ... 0.2 ...		

## 6.3.21 protein\_abundance\_stdev\_study\_variable[1-n]

<b>Description:</b>	The standard deviation of the protein's abundance. If a protein's abundance is given for a certain study variable, the corresponding standard deviation column MUST also be present (in case the value is not available "null" should be used).		
<b>Type:</b>	Double		
<b>Mandatory</b>		Summary	Complete
	Quantification	(✓) <sup>1</sup>	(✓) <sup>1</sup>
	Identification		
<sup>1</sup> mandatory if protein abundance study variable reported			
<b>Example:</b>	PRT accession ... protein_abundance_stdev_study_variable[1] ... PRH P12345 ... 0.4 ...		

## 6.3.22 protein\_abundance\_std\_error\_study\_variable [1-n]

<b>Description:</b>	The standard error of the protein's abundance. If a protein's abundance is given for a certain study variable, the corresponding standard error column MUST also be present (in case the value is not available "null" should be		
---------------------	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--	--

	used).		
Type:	Double		
Mandatory		Summary	Complete
	Quantification	(✓) <sup>1</sup>	(✓) <sup>1</sup>
	Identification		
	<sup>1</sup> mandatory if protein abundance study variable reported		
Example:	PRT accession ... protein_abundance_study_variable[1] ...		
	protein_abundance_std_error_study_variable[1] ...		
	PRH P12345	... 0.4	... 0.03
			...

### 6.3.23 opt\_{identifier}\_\*

Description:	Additional columns can be added to the end of the protein table. These column headers MUST start with the prefix “opt_” followed by the {identifier} of the object they reference: assay, study variable, MS run or “global” (if the value relates to all replicates). Column names MUST only contain the following characters: ‘A’-‘Z’, ‘a’-‘z’, ‘0’-‘9’, ‘_’, ‘-’, ‘[’, ‘]’, and ‘.’. CV parameter accessions MAY be used for optional columns following the format: opt_{identifier}_cv_{accession}_{parameter name}. Spaces within the parameter’s name MUST be replaced by ‘_’.		
Type:	Column		
Mandatory		Summary	Complete
	Quantification		
	Identification		
Example:	PRT    accession        ...    opt_assay[1]_my_value    opt_global_another_value PRH    P12345            ...    My value about assay[1]        some other value that is across reps		

## 6.4 Peptide Section

The peptide section is table based. The peptide section must always come after the metadata section and or protein section if these are present in the file. All table columns MUST be tab separated. There MUST NOT be any empty cells. Missing values MUST be reported using “null”. Most columns are mandatory. The order of columns is not specified although for ease of human interpretation, it is RECOMMENDED to follow the order specified below.

### 6.4.1 sequence

<b>Description:</b>	The peptide's sequence		
<b>Type:</b>	String		
<b>Mandatory</b>		Summary	Complete
	Quantification	✓	✓
	Identification	1	1
	¹Not recommended in identification only files		
<b>Example:</b>	PEH    sequence		

### 6.4.2 accession

<b>Description:</b>	The protein's accession the peptide is associated with. In case no protein section is present in the file or the peptide was not assigned to a protein the field should be filled with “null”. If the peptide can be assigned to more than one protein, multiple rows SHOULD be provided for each peptide to protein mapping.		
<b>Type:</b>	String		
<b>Mandatory</b>		Summary	Complete
	Quantification	✓	✓
	Identification	1	1

<sup>1</sup> Not recommended in identification only files

<b>Example:</b>	PEH sequence accession ... PEP KVPQVSTPTLVEVSR P02768 ...
-----------------	--------------------------------------------------------------

#### 6.4.3 unique

<b>Description:</b>	Indicates whether the peptide is unique for this protein in respect to the searched database.		
<b>Type:</b>	Boolean (0/1)		
<b>Mandatory</b>		Summary	Complete
	Quantification	✓	✓
	Identification	↑	↑
	↑ Not recommended in identification only files		
<b>Example:</b>	PEH sequence accession unique ... PEP KVPQVSTPTLVEVSR P02768 0 ... PEP VFDEFKPLVEEPQNLIK P02768 1 ...		

#### 6.4.4 database

<b>Description:</b>	The protein database used for the search (could theoretically come from a different species) and the peptide sequence comes from.		
<b>Type:</b>	String		
<b>Mandatory</b>		Summary	Complete
	Quantification	✓	✓
	Identification	↑	↑
	↑ Not recommended in identification only files		
<b>Example:</b>	PEH sequence accession unique database ... PEP KVPQVSTPTLVEVSR P02768 0 UniProtKB ... PEP VFDEFKPLVEEPQNLIK P02768 1 UniProtKB ...		

#### 6.4.5 database\_version

<b>Description:</b>	The protein database's version – in case there is no version available (custom build) the creation / download (e.g., for NCBI nr) date should be given. Additionally, the number of entries in the database MAY be reported in round brackets after the version in the format: {version} ({#entries} entries), for example “2011-11 (1234 entries)”.		
<b>Type:</b>	String		
<b>Mandatory</b>		Summary	Complete
	Quantification	✓	✓
	Identification	↑	↑
	↑ Not recommended in identification only files		
<b>Example:</b>	PEH sequence accession unique database database_version ... PEP KVPQVSTPTLVEVSR P02768 0 UniProtKB 2011_11 ... PEP VFDEFKPLVEEPQNLIK P02768 1 UniProtKB 2011_11 ...		

#### 6.4.6 search\_engine

<b>Description:</b>	A “ ” delimited list of search engine(s) that identified this peptide. Search engines must be supplied as parameters.		
<b>Type:</b>	Parameter List		
<b>Mandatory</b>		Summary	Complete
	Quantification	✓	✓
	Identification	↑	↑
	↑ Not recommended in identification only files		
<b>Example:</b>	COM In this example the first protein was identified by Mascot and X!Tandem while COM the second protein was only identified by Mascot. PEH sequence ... search_engine ... PEP KVPQVSTPTLVEVSR ... [MS,MS:1001207,Mascot,]   [MS,MS:1001476,X!Tandem,] ... PEP VFDEFKPLVEEPQNLIK ... [MS,MS:1001207,Mascot,] ...		

#### 6.4.7 best\_search\_engine\_score[1-n]

<b>Description:</b>	The best search engine score (for this type of score) for the given peptide
---------------------	-----------------------------------------------------------------------------

	across all replicates reported. The type of score MUST be defined in the metadata section. If the peptide was not identified by the specified search engine, “null” MUST be reported.		
<b>Type:</b>	Double		
<b>Mandatory</b>		Summary	Complete
	Quantification	✓	✓
	Identification	1	1
	*Not recommended in identification only files		
<b>Example:</b>	MTD peptide_search_engine_score[1] [MS,MS:1001171,Mascot score,] ... PEH sequence ... best_search_engine_score[1] PEP KVPQVSTPTLVEVSR ... 47 PEP VFDEFKPLVEEPQNLIK ... 29		

#### 6.4.8 search\_engine\_score[1-n]\_ms\_run[1-n]

<b>Description:</b>	The search engine score for the given peptide in the defined ms run. The type of score MUST be defined in the metadata section. If the peptide was not identified by the specified search engine “null” must be reported.		
<b>Type:</b>	Double		
<b>Mandatory</b>		Summary	Complete
	Quantification		✓
	Identification	1	1
	*Not recommended in identification only files		
<b>Example:</b>	MTD peptide_search_engine_score[1] [MS,MS:1001171,Mascot score,] ... PEH sequence ...search_engine_score[1]_ms_run[1] PEP KVPQVSTPTLVEVSR ... 47 PEP VFDEFKPLVEEPQNLIK ... 29		

#### 6.4.9 reliability

<b>Description:</b>	The reliability of the given peptide identification. This must be supplied by the resource and has to be one of the following values: 1: high reliability 2: medium reliability 3: poor reliability  Important: An identification's reliability is resource dependent.		
<b>Type:</b>	Integer		
<b>Mandatory</b>		Summary	Complete
	Quantification		
	Identification	1	1
	*Not recommended in identification only files		
<b>Example:</b>	PEH sequence ... reliability ... PEP KVPQVSTPTLVEVSR ... 3 PEP VFDEFKPLVEEPQNLIK ... 1		

#### 6.4.10 modifications

<b>Description:</b>	The peptide's modifications or substitutions. To further distinguish peptide terminal modifications, these SHOULD be reported at position 0 or <i>peptide size</i> + 1 respectively. For detailed information see the modifications section in the protein table. If substitutions are reported, the “sequence” column MUST contain the original, unaltered sequence. Note that in contrast to the PSM section, fixed modifications or modifications caused by the quantification reagent i.e. the SILAC labels/tags SHOULD NOT be reported. It is thus also expected that modification reliability scores will typically be reported at the		
---------------------	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--	--

	PSM-level only.		
<b>Type:</b>	String		
<b>Mandatory</b>		Summary	Complete
	Quantification	✓	✓
	Identification	1	1
	Not recommended in identification only files		
<b>Example:</b>	PEH sequence ... modifications ...		
	PEP KVPQVSTPTLVEVSR ... 10-MOD:00412 ...		
	PEP VFDEFKPLVEEPQNLIK ... NULL ...		

#### 6.4.11 retention\_time

<b>Description:</b>	A '['-separated list of time points. Semantics may vary on how retention times are reported. For quantification approaches, different exporters MAY wish to export the retention times of all spectra used for quantification (e.g. in MS <sup>2</sup> approaches) or the centre point of the feature quantified for MS <sup>1</sup> approaches. It is assumed that the reported value(s) are for a given “master” peptide from one assay only (and the unlabeled peptide in label-based approaches). If the exporter wishes to export values for all assays, this can be done using optional columns. Retention time MUST be reported in seconds. Otherwise, units MUST be reported in the Metadata Section (“colunit-peptide”).		
<b>Type:</b>	Double List		
<b>Mandatory</b>		Summary	Complete
	Quantification	✓	✓
	Identification	1	1
	Not recommended in identification only files		
<b>Example:</b>	PEH sequence ... retention_time ...		
	PEP KVPQVSTPTLVEVSR ... 10.2 ...		
	PEP VFDEFKPLVEEPQNLIK ... 15.8 ...		

#### 6.4.12 retention\_time\_window

<b>Description:</b>	Start and end of the retention time window separated by a single '['. Semantics may vary but its primary intention is to report feature boundaries of eluting peptides (along with feature centroids in the retention_time column). It is assumed that the reported interval is for a given “master” peptide from one assay only (and the unlabeled peptide in label-based approaches). If the exporter wishes to export values for all assays, this can be done using optional columns. Retention time windows MUST be reported in seconds. Otherwise, units MUST be reported in the Metadata Section (“colunit-peptide”).		
<b>Type:</b>	Double List		
<b>Mandatory</b>		Summary	Complete
	Quantification	✓	✓
	Identification	1	1
	Not recommended in identification only files		
<b>Example:</b>	PEH sequence ... retention_time_window ...		
	PEP KVPQVSTPTLVEVSR ... 1123.2 1145.3 ...		

#### 6.4.13 charge

<b>Description:</b>	The charge assigned by the search engine/software. In case multiple charge states for the same peptide are observed these should be reported as distinct entries in the peptide table. In case the charge is unknown “null” MUST be used.		
<b>Type:</b>	Integer		
<b>Mandatory</b>		Summary	Complete
	Quantification	✓	✓
	Identification	1	1



	Not recommended in identification only files			
<b>Example:</b>	PEH	sequence	...	charge
	PEP	KVPQVSTPTLVEVSR	...	2
	PEP	VFDEFKPLVEEPQNLIK	...	3

## 6.4.14 mass\_to\_charge

Description:	The precursor's experimental mass to charge ( $m/z$ ). It is assumed that the reported value is for a given "master" peptide from one assay only (and the unlabeled peptide in label-based approaches). If the exporter wishes to export values for all assays, this can be done using optional columns.				
Type:	Double				
Mandatory		Summary	Complete		
	Quantification	✓	✓		
	Identification	1	1		
	¹Not recommended in identification only files				
Example:	PEH	sequence	...	mass_to_charge	...
	PEP	KVPQVSTPTLVEVSR	...	1234.4	...
	PEP	VFDEFKPLVEEPQNLIK	...	123.4	...

## 6.4.15 uri

<b>Description:</b>	A URI pointing to the peptide's entry in the experiment it was identified in (e.g., the peptide's PRIDE entry).			
<b>Type:</b>	URI			
<b>Mandatory</b>		Summary	Complete	
	Quantification			
	Identification	1	1	
	¹Not recommended in identification only files			
<b>Example:</b>	PEH	sequence	...	uri
	PEP	KVPQVSTPTLVEVSR	...	http://www.ebi.ac.uk/pride/link/to/peptide
	PEP	VFDEFKPLVEEPQNLIK	...	http://www.ebi.ac.uk/pride/link/to/peptide

## 6.4.16 spectra\_ref

Description:	Reference to spectra in a spectrum file. It is expected that spectra_ref SHOULD only be used for MS <sup>2</sup> -based quantification approaches, in which retention time values cannot identify the spectra used for quantitation. The reference must be in the format ms_run[1-n]:{SPECTRA_REF} where SPECTRA_REF MUST follow the format defined in 5.2. Multiple spectra MUST be referenced using a “]” delimited list.			
Type:	String			
Mandatory		Summary	Complete	
	Quantification		(✓) <sup>2</sup>	
	Identification	1	1	
	<sup>1</sup> Not recommended in identification only files			
	<sup>2</sup> Mandatory only if MS2 based quantification is used			
Example:	PEH	sequence	...	spectra_ref
	PEP	KVPQVSTPTLVEVSR	...	ms_run[1]:index=5
	PEP	VFDEFKPLVEEPQNLIK	...	ms_run[2]:index=7 ms_run[2]:index=9

## 6.4.17 peptide\_abundance\_assay[1-n]

Description:	The peptide's abundance in the given assay.			
Type:	Double			
Mandatory		Summary	Complete	
	Quantification		✓	
	Identification	1	1	
	1Not recommended in identification only files			
	2If quantification data is reported on assays level			
Example:	PEH	sequence	...	peptide_abundance_assay[1] peptide_abundance_assay[2]...
	PEP	KVPOVSTPTLVEVSR	...	0.4 0.5

**6.4.18 peptide\_abundance\_study\_variable[1-n]**

<b>Description:</b>	The peptide's abundance in the given study variable, for example calculated as an average of assay values.		
<b>Type:</b>	Double		
<b>Mandatory</b>		Summary	Complete
	Quantification	✓	✓
	Identification	1	1
	<sup>1</sup> Not recommended in identification only files <sup>2</sup> mandatory if study variables are reported		
<b>Example:</b>	PEH sequence ... peptide_abundance_study_variable[1] ... PEP KVPQVSTPTLVEVSR ... 0.4 ...		

**6.4.19 peptide\_abundance\_stdev\_study\_variable[1-n]**

<b>Description:</b>	The standard deviation of the peptide's abundance for a given study variable.		
<b>Type:</b>	Double		
<b>Mandatory</b>		Summary	Complete
	Quantification	(✓) <sup>2</sup>	(✓) <sup>2</sup>
	Identification	1	1
	<sup>1</sup> Not recommended in identification only files <sup>2</sup> mandatory if peptide_abundance_study_variable reported		
<b>Example:</b>	PEH sequence ... peptide_abundance_study_variable [1] peptide_abundance_stdev_study_variable[1] ... PEP KVPQVSTPTLVEVSR ... 0.4 ... 0.2 ...		

**6.4.20 peptide\_abundance\_std\_error\_study\_variable[1-n]**

<b>Description:</b>	The standard error of the peptide's abundance for a given study variable.		
<b>Type:</b>	Double		
<b>Mandatory</b>		Summary	Complete
	Quantification	(✓) <sup>2</sup>	(✓) <sup>2</sup>
	Identification	1	1
	<sup>1</sup> Not recommended in identification only files <sup>2</sup> mandatory if peptide_abundance_study_variable reported		
<b>Example:</b>	PEH sequence ... peptide_abundance_study_variable[1] ... peptide_abundance_std_error_study_variable[1] ... PEP KVPQVSTPTLVEVSR ... 0.4 ... 0.2 ...		

**6.4.21 opt\_{identifier}\_\***

<b>Description:</b>	Additional columns can be added to the end of the peptide table. These column headers MUST start with the prefix "opt_" followed by the {identifier} of the object they reference: assay, study variable, MS run or "global" (if the value relates to all replicates). Column names MUST only contain the following characters: 'A'-'Z', 'a'-'z', '0'-'9', '_', '-', '[', ']', and ':'. CV parameter accessions MAY be used for optional columns following the format: opt_{identifier}_cv_{accession}_{parameter name}. Spaces within the parameter's name MUST be replaced by '_'.		
<b>Type:</b>	Column		
<b>Mandatory</b>		Summary	Complete
	Quantification	1	1
	Identification	1	1
	<sup>1</sup> Not recommended in identification only files		
<b>Example:</b>	PRT accession ... opt_assay[1]_my_value opt_global_another value PRH P12345 ... My value about assay[1] some other value that is across reps		

**6.5 PSM Section**

The PSM section is table-based. The PSM section MUST always come after the metadata section, peptide section and or protein section if they are present in the file. All table columns

MUST be tab separated. Missing values MUST be reported using “null”. Most columns are mandatory. The order of columns is not specified although for ease of human interpretation, it is RECOMMENDED to follow the order specified below.

#### 6.5.1 sequence

<b>Description:</b>	The peptide's sequence corresponding to the PSM		
<b>Type:</b>	String		
<b>Mandatory</b>		Summary	Complete
	Quantification	✓	✓
	Identification	✓	✓
<b>Example:</b>	PSH sequence ... PSM KVPQVSTPTLVEVSR ... PSM EIEILACEIR ...		

#### 6.5.2 PSM\_ID

<b>Description:</b>	A unique identifier for a PSM within the file. If a PSM can be matched to multiple proteins, the same PSM should be represented on multiple rows with different accessions and the same PSM_ID.		
<b>Type:</b>	Integer		
<b>Mandatory</b>		Summary	Complete
	Quantification	✓	✓
	Identification	✓	✓
<b>Example:</b>	PSH sequence PSM_ID accession... PSM KVPQVSTPTLVEVSR 1 P02768 ... PSM PEPTIDR 2 P04267 ... PSM PEPTIDR 2 P04268 ...		

#### 6.5.3 accession

<b>Description:</b>	The protein's accession the corresponding peptide sequence (coming from the PSM) is associated with. In case no protein section is present in the file or the peptide was not assigned to a protein the field should be filled with “null”. If the PSM can be assigned to more than one protein, the same PSM should be represented on multiple rows with the same unique identifier.		
<b>Type:</b>	String		
<b>Mandatory</b>		Summary	Complete
	Quantification	✓	✓
	Identification	✓	✓
<b>Example:</b>	PSH sequence accession ... PSM KVPQVSTPTLVEVSR P02768 ...		

#### 6.5.4 unique

<b>Description:</b>	Indicates whether the peptide sequence (coming from the PSM) is unique for this protein in respect to the searched database.		
<b>Type:</b>	Boolean (0/1)		
<b>Mandatory</b>		Summary	Complete
	Quantification	✓	✓
	Identification	✓	✓
<b>Example:</b>	PSH sequence accession unique ... PSM KVPQVSTPTLVEVSR P02768 0 ... PSM VFDEFKPLVEEPQNLIK P02768 1 ...		

#### 6.5.5 database

<b>Description:</b>	The protein database used for the search (could theoretically come from a different species) and the peptide sequence comes from.		
<b>Type:</b>	String		
<b>Mandatory</b>		Summary	Complete
	Quantification	✓	✓
	Identification	✓	✓

<b>Example:</b>	PSH sequence accession unique database ... PSM KVPQVSTPTLVEVSR P02768 0 UniProtKB ... PSM VFDEFKPLVEEPQNLIK P02768 1 UniProtKB ...
-----------------	------------------------------------------------------------------------------------------------------------------------------------------

### 6.5.6 database\_version

<b>Description:</b>	The protein database's version – in case there is no version available (custom build) the creation / download (e.g., for NCBI nr) date should be given. Additionally, the number of entries in the database MAY be reported in round brackets after the version in the format: {version} ({#entries} entries), for example “2011-11 (1234 entries)”.						
<b>Type:</b>	String						
<b>Mandatory</b>		Summary	Complete				
	Quantification	✓	✓				
	Identification	✓	✓				
<b>Example:</b>	PSH	sequence	accession	unique	database	database_version	...
	PSM	KVPQVSTPTLVEVSR	P02768	0	UniProtKB	2011_11	...
	PSM	VFDEFKPLVEEPONLIK	P02768	1	UniProtKB	2011_11	...

### 6.5.7 search\_engine

Description:	A “ ” delimited list of search engine(s) that identified the PSM. Search engines must be supplied as parameters.				
Type:	Parameter List				
Mandatory		Summary	Complete		
	Quantification	✓	✓		
	Identification	✓	✓		
Example:	COM In this example the first protein was identified by Mascot and X!Tandem while				
	COM the second protein was only identified by Mascot.				
	PSH	sequence	...	search_engine	...
	PSM	KVPQVSTPTLVEVSR	...	[MS,MS:1001207,Mascot,]  [MS,MS:1001476,X!Tandem,]	...
	PSM	VFDEFKPLVEEPONLIK	...	[MS,MS:1001207,Mascot,]	...

### 6.5.8 search\_engine\_score[1-n]

Description:	The search engine score for the given PSM. The type of score MUST be defined in the metadata section. If the peptide was not identified by the specified search engine “null” must be reported.					
Type:	Double					
Mandatory		Summary	Complete			
	Quantification	✓	✓			
	Identification	✓	✓			
Example:	MTD	psm_search_engine_score[1]	[MS,MS:1001171,Mascot score,]			
	MTD	psm_search_engine_score[2]	[MS,MS:1001330,X!Tandem:expect,]			
	...					
	PSH	sequence	...	search_engine_score[1]	search_engine_score[2]	...
	PSM	KVPQVSTPTLVEVSR	...	47	0.001	...
	PSM	VFDEFKPLVEEPONLIK	...	29	null	...

### 6.5.9 reliability

<b>Description:</b>	The reliability of the given PSM. This must be supplied by the resource and has to be one of the following values: 1: high reliability 2: medium reliability 3: poor reliability  Important: An identification's reliability is resource dependent.					
<b>Type:</b>	Integer					
<b>Mandatory</b>		Summary	Complete			
	Quantification					
	Identification					

<b>Example:</b>	PSH sequence ... reliability ... PSM KVPQVSTPTLVEVSR ... 3 ... PSM VFDEFKPLVEEPQNLIK ... 1 ...
-----------------	------------------------------------------------------------------------------------------------------

#### 6.5.10 modifications

<b>Description:</b>	The peptide's (coming from the PSM) modifications or substitutions. To further distinguish peptide terminal modifications, these SHOULD be reported at position 0 or <i>peptide size</i> + 1 respectively. For detailed information see the modifications section in the protein table. If substitutions are reported, the "sequence" column MUST contain the original, unaltered sequence. Note that in contrast to the PRT and PEP section all modifications (variable and fixed modifications, including those induced by quantification reagents) MUST BE reported in the PSM section.		
<b>Type:</b>	String		
<b>Mandatory</b>		Summary	Complete
	Quantification	✓	✓
	Identification	✓	✓
<b>Example:</b>	PSH sequence ... modifications ... PSM KVPQVSTPTLVEVSR ... 10[MS,MS:1001876, modification probability, 0.8]-MOD:00412 ... PSM VFDEFKPLVEEPQNLIK ... NULL ...		

#### 6.5.11 retention\_time

<b>Description:</b>	The retention time of the spectrum. A '['-separated list of multiple time points is allowed in case multiple spectra were combined by the search engine to make the PSM. It MUST be reported in seconds. Otherwise, the units MUST be reported in the Metadata Section ('columnit_psm').		
<b>Type:</b>	Double List		
<b>Mandatory</b>		Summary	Complete
	Quantification	✓	✓
	Identification	✓	✓
<b>Example:</b>	PSH sequence ... retention_time ... PSM KVPQVSTPTLVEVSR ... 10.2 ... PSM VFDEFKPLVEEPQNLIK ... 15.8 ...		

#### 6.5.12 charge

<b>Description:</b>	The charge assigned by the search engine/software.		
<b>Type:</b>	Integer		
<b>Mandatory</b>		Summary	Complete
	Quantification	✓	✓
	Identification	✓	✓
<b>Example:</b>	PSH sequence ... charge ... PSM KVPQVSTPTLVEVSR ... 2 ... PSM VFDEFKPLVEEPQNLIK ... 3 ...		

#### 6.5.13 exp\_mass\_to\_charge

<b>Description:</b>	The PSM's experimental mass to charge ( $m/z$ ).		
<b>Type:</b>	Double		
<b>Mandatory</b>		Summary	Complete
	Quantification	✓	✓
	Identification	✓	✓
<b>Example:</b>	PSH sequence ... mass_to_charge ... PSM KVPQVSTPTLVEVSR ... 1234.4 ... PSM VFDEFKPLVEEPQNLIK ... 123.4 ...		

#### 6.5.14 calc\_mass\_to\_charge

<b>Description:</b>	The PSM's calculated (theoretical) mass to charge ( $m/z$ ).		
<b>Type:</b>	Double		

<b>Mandatory</b>		Summary	Complete
	Quantification	✓	✓
	Identification	✓	✓
<b>Example:</b>	PSH sequence	...	mass_to_charge ...
	PSM KVPQVSTPTLVEVSR	...	1234.4 ...
	PSM VFDEFKPLVEEPQNLIK	...	123.4 ...

## 6.5.15 uri

<b>Description:</b>	A URI pointing to the PSM's entry in the experiment it was identified in (e.g., the peptide's PRIDE entry).		
<b>Type:</b>	URI		
<b>Mandatory</b>		Summary	Complete
	Quantification		
	Identification		
<b>Example:</b>	PSH sequence	...	uri ...
	PSM KVPQVSTPTLVEVSR	...	http://www.ebi.ac.uk/pride/link/to/peptide ...
	PSM VFDEFKPLVEEPQNLIK	...	http://www.ebi.ac.uk/pride/link/to/peptide ...

## 6.5.16 spectra\_ref

<b>Description:</b>	Reference to a spectrum in a spectrum file. The reference must be in the format <code>ms_run[1-n]:{SPECTRA_REF}</code> where SPECTRA_REF MUST follow the format defined in 5.2. Multiple spectra MUST be referenced using a "]" delimited list for the (rare) cases in which search engines have combined multiple spectra to make identifications.		
<b>Type:</b>	String		
<b>Mandatory</b>		Summary	Complete
	Quantification	✓	✓
	Identification	✓	✓
<b>Example:</b>	PSH sequence	...	spectra_ref ...
	PSM KVPQVSTPTLVEVSR	...	ms_run[1]:index=5 ...
	PSM VFDEFKPLVEEPQNLIK	...	ms_run[2]:index=7 ms_run[2]:index=9 ...

## 6.5.17 pre

<b>Description:</b>	Amino acid preceding the peptide (coming from the PSM) in the protein sequence. If unknown "null" MUST be used, if the peptide is N-terminal "-" MUST be used.		
<b>Type:</b>	String		
<b>Mandatory</b>		Summary	Complete
	Quantification	✓	✓
	Identification	✓	✓
<b>Example:</b>	PSH sequence	...	pre post ...
	PSM KVPQVSTPTLVEVSR	...	K D ...
	PSM VFDEFKPLVEEPQNLIK	...	R L ...

## 6.5.18 post

<b>Description:</b>	Amino acid following the peptide (coming from the PSM) in the protein sequence. If unknown "null" MUST be used, if the peptide is C-terminal "-" MUST be used.		
<b>Type:</b>	String		
<b>Mandatory</b>		Summary	Complete
	Quantification	✓	✓
	Identification	✓	✓
<b>Example:</b>	PSH sequence	...	pre post ...
	PSM KVPQVSTPTLVEVSR	...	K D ...
	PSM VFDEFKPLVEEPQNLIK	...	R L ...

## 6.5.19 start

<b>Description:</b>	The start position of the peptide (coming from the PSM) within the protein, counting 1 as the N-terminus of the protein.		
---------------------	--------------------------------------------------------------------------------------------------------------------------	--	--

<b>Type:</b>	Integer				
<b>Mandatory</b>		Summary	Complete		
	Quantification	✓	✓		
	Identification	✓	✓		
<b>Example:</b>	PSH sequence	...	start	end	...
	PSM KVPQVSTPTLVEVSR	...	45	57	...
	PSM VFDEFKPLVEEPQNLIK	...	34	46	...

## 6.5.20 end

<b>Description:</b>	The end position of the peptide (coming from the PSM) within the protein, counting 1 as the N-terminus of the protein.				
<b>Type:</b>	Integer				
<b>Mandatory</b>		Summary	Complete		
	Quantification	✓	✓		
	Identification	✓	✓		
<b>Example:</b>	PSH sequence	...	start	end	...
	PSM KVPQVSTPTLVEVSR	...	45	57	...
	PSM VFDEFKPLVEEPQNLIK	...	34	46	...

## 6.5.21 opt\_{identifier}\_\*

<b>Description:</b>	Additional columns can be added to the end of the PSM table. These column headers MUST start with the prefix “opt_” followed by the {identifier} of the object they reference: assay, study variable, MS run or “global” (if the value relates to all replicates). Column names MUST only contain the following characters: ‘A’-‘Z’, ‘a’-‘z’, ‘0’-‘9’, ‘_’, ‘-’, ‘[’, ‘]’, and ‘.’. CV parameter accessions MAY be used for optional columns following the format: opt_{identifier}_cv_{accession}_{parameter name}. Spaces within the parameter’s name MUST be replaced by ‘_’.				
<b>Type:</b>	Column				
<b>Mandatory</b>		Summary	Complete		
	Quantification				
	Identification				
<b>Example:</b>	PSH sequence	...	opt_assay[1]_my_value	opt_global_another_value	
	PSM PEPTIDER	...	My value about assay[1]	some other value that is across reps	

## 6.6 Small Molecule Section

The small molecule section is table-based. The small molecule section MUST always come after the metadata section, peptide section and or protein section if they are present in the file. All table columns MUST be Tab separated. There MUST NOT be any empty cells. Missing values MUST be reported using “null”. Most columns are mandatory. The order of columns is not specified although for ease of human interpretation, it is RECOMMENDED to follow the order specified below.

## 6.6.1 identifier

<b>Description:</b>	A list of “[” separated possible identifiers for these small molecules. The database identifier must be preceded by the resource description followed by a colon (in case this is not already part of the identifier format).				
<b>Type:</b>	String List				
<b>Mandatory</b>		Summary	Complete		
	Quantification	✓	✓		
	Identification	✓	✓		
<b>Example:</b>	SMH identifier	...			
	SML CID:00027395	...			
	SML HMDB:HMDB12345	...			



## 6.6.2 chemical\_formula

<b>Description:</b>	<p>The chemical formula of the identified compound.</p> <p>This should be specified in Hill notation (EA Hill 1900), i.e. elements in the order C, H and then alphabetically all other elements. Counts of one may be omitted. Elements should be capitalized properly to avoid confusion (e.g., “CO” vs. “Co”). The chemical formula reported should refer to the neutral form. Charge state is reported by the charge field. This permits the comparison of positive and negative mode results.</p> <p><b>Example:</b> N-acetylglucosamine would be encoded by the string “C8H15NO6”</p>		
<b>Type:</b>	String		
<b>Mandatory</b>		Summary	Complete
	Quantification	✓	✓
	Identification	✓	✓
<b>Example:</b>	SMH identifier ... chemical formula ...		
	SML CID:00027395 ... C17H20N4O2 ...		

## 6.6.3 smiles

<b>Description:</b>	<p>The molecules structure in the simplified molecular-input line-entry system (SMILES). If there are more than one SMILES for a given small molecule, use the “ ” separator.</p>		
<b>Type:</b>	String List		
<b>Mandatory</b>		Summary	Complete
	Quantification	✓	✓
	Identification	✓	✓
<b>Example:</b>	SMH identifier ... chemical formula smiles		
	SML CID:00027395 ... C17H20N4O2 C1=CC=C(C=C1)CCNC(=O)CCNCC(=O)C2=CC=NC=C2 ...		

## 6.6.4 inchi\_key

<b>Description:</b>	<p>The standard IUPAC International Chemical Identifier (InChI) Key of the given substance. If there are more than one InChI identifier for a given small molecule, use the “ ” separator.</p>		
<b>Type:</b>	String List		
<b>Mandatory</b>		Summary	Complete
	Quantification	✓	✓
	Identification	✓	✓
<b>Example:</b>	SMH identifier ... chemical formula ... inchi_key		
	SML CID:00027395 ... C17H20N4O2 ... QXBMEGUKVLFJAM-UHFFFAOYSA-N ...		

## 6.6.5 description

<b>Description:</b>	The small molecule’s description / name.		
<b>Type:</b>	String		
<b>Mandatory</b>		Summary	Complete
	Quantification	✓	✓
	Identification	✓	✓
<b>Example:</b>	SMH identifier ... description		
	SML CID:00027395 ... N-(2-phenylethyl)-3-[2-(pyridine-4-carbonyl)hydrazinyl]propanamide...		

## 6.6.6 exp\_mass\_to\_charge

<b>Description:</b>	The small molecule’s experimental mass to charge ( <i>m/z</i> ).		
<b>Type:</b>	Double		
<b>Mandatory</b>		Summary	Complete
	Quantification	✓	✓
	Identification	✓	✓
<b>Example:</b>	SMH sequence ... mass_to_charge ...		
	SMM CID:00027395 ... 1234.4 ...		

## 6.6.7 calc\_mass\_to\_charge

<b>Description:</b>	The small molecule's precursor's calculated (theoretical) mass to charge ratio.		
<b>Type:</b>	Double		
<b>Mandatory</b>		Summary	Complete
	Quantification	✓	✓
	Identification	✓	✓
<b>Example:</b>	SMH identifier ...	mass_to_charge ...	
	SML CID:00027395 ...	1234.5 ...	

## 6.6.8 charge

<b>Description:</b>	The charge assigned by the search engine/software.		
<b>Type:</b>	Integer		
<b>Mandatory</b>		Summary	Complete
	Quantification	✓	✓
	Identification	✓	✓
<b>Example:</b>	SMH identifier ...	charge ...	
	SML CID:00027395 ...	2 ...	

## 6.6.9 retention\_time

<b>Description:</b>	A ' '-separated list of time points. Semantics may vary. This time should refer to the small molecule's retention time if determined or the mid point between the first and last spectrum identifying the small molecule. It MUST be reported in seconds. Otherwise, the corresponding unit MUST be specified in the Metadata Section ('colunit-small_molecule').		
<b>Type:</b>	Double List		
<b>Mandatory</b>		Summary	Complete
	Quantification	✓	✓
	Identification	✓	✓
<b>Example:</b>	SMH identifier ...	retention_time ...	
	SML CID:00027395 ...	10.2 11.5 ...	

## 6.6.10 taxid

<b>Description:</b>	The taxonomy id coming from the NEWT taxonomy for the species (if applicable).		
<b>Type:</b>	Integer		
<b>Mandatory</b>		Summary	Complete
	Quantification	✓	✓
	Identification	✓	✓
<b>Example:</b>	SMH identifier ...	taxid ...	
	SML CID:00027395 ...	null ...	

## 6.6.11 species

<b>Description:</b>	The species as a human readable string (if applicable).		
<b>Type:</b>	String		
<b>Mandatory</b>		Summary	Complete
	Quantification	✓	✓
	Identification	✓	✓
<b>Example:</b>	SMH identifier ...	species ...	
	SML CID:00027395 ...	null ...	

## 6.6.12 database

<b>Description:</b>	Generally references the used spectral library (if applicable).		
<b>Type:</b>	String		
<b>Mandatory</b>		Summary	Complete
	Quantification	✓	✓
	Identification	✓	✓
<b>Example:</b>	SMH identifier ...	database ...	
	SML CID:00027395 ...	name of used database ...	

## 6.6.13 database\_version

<b>Description:</b>	Either the version of the used database if available or otherwise the date of creation. Additionally, the number of entries in the database MAY be reported in round brackets after the version in the format: {version} ({#entries} entries), for example “2011-11 (1234 entries)”.		
<b>Type:</b>	String		
<b>Mandatory</b>		Summary	Complete
	Quantification	✓	✓
	Identification	✓	✓
<b>Example:</b>	SMH identifier ... database_version ... SML CID:00027395 ... 2011-12-22 ...		

## 6.6.14 reliability

<b>Description:</b>	The reliability of the given small molecule identification. This must be supplied by the resource and MUST be reported as an integer between 1-4: 1: identified metabolites 2: putatively annotated compounds 3: putatively characterized compound classes 4: unknown compounds		
<b>Type:</b>	Integer		
<b>Mandatory</b>		Summary	Complete
	Quantification		
	Identification		
<b>Example:</b>	SMH identifier ... reliability ... SML CID:00027395 ... 3 ...		

## 6.6.15 uri

<b>Description:</b>	A URI pointing to the small molecule's entry in the experiment it was identified in (e.g., the small molecule's PRIDE entry).		
<b>Type:</b>	URI		
<b>Mandatory</b>		Summary	Complete
	Quantification		
	Identification		
<b>Example:</b>	SMH identifier ... uri ... SML CID:00027395 ... http://www.ebi.ac.uk/pride/link/to/identification ...		

## 6.6.16 spectra\_ref

<b>Description:</b>	Reference to a spectrum in a spectrum file. The reference must be in the format ms_run[1-n]:{SPECTRA_REF} where spectra_ref MUST follow the format defined in 5.2. Multiple spectra can be referenced using a “ ” delimited list.		
<b>Type:</b>	String		
<b>Mandatory</b>		Summary	Complete
	Quantification	✓	✓
	Identification	✓	✓
<b>Example:</b>	SMH identifier ... spectra_ref ... SML CID:00027395 ... ms_run[1]:index=1002 ...		

## 6.6.17 search\_engine

<b>Description:</b>	A “ ” delimited list of search engine(s) that identified this small molecule. Search engines must be supplied as parameters.		
<b>Type:</b>	Parameter List		
<b>Mandatory</b>		Summary	Complete

	Quantification	✓	✓
	Identification	✓	✓
<b>Example:</b>	SMH identifier ... search_engine ... SML CID:00027395 ... [MS, MS:1001477, SpectraST,] ...		

**6.6.18 best\_search\_engine\_score[1-n]**

<b>Description:</b>	The best search engine score (for this type of score) for the given small molecule across all replicates reported. The type of score MUST be defined in the metadata section. If the small molecule was not identified by the specified search engine, "null" MUST be reported.		
<b>Type:</b>	Parameter List		
<b>Mandatory</b>		Summary	Complete
	Quantification	✓	✓
	Identification	✓	✓
<b>Example:</b>	MTD smallmolecule_search_engine_score[1] [MS, MS:1001419, SpectraST:discriminant score F,] ... SMH identifier ... best_search_engine_score[1] ... SML CID:00027395 ... 0.7 ...		

**6.6.19 search\_engine\_score[1-n]\_ms\_run[1-n]**

<b>Description:</b>	The search engine score for the given small molecule in the defined ms run. The type of score MUST be defined in the metadata section. If the small molecule was not identified by the specified search engine "null" must be reported.		
<b>Type:</b>	Parameter List		
<b>Mandatory</b>		Summary	Complete
	Quantification		✓
	Identification	1	1
	1Not recommended in identification only files		
<b>Example:</b>	MTD smallmolecule_search_engine_score[1] [MS, MS:1001419, SpectraST:discriminant score F,] ... SMH identifier ... search_engine_score[1]_ms_run[1] search_engine_score[1]_ms_run[2] SML CID:00027395 ... 0.7 ... 0.74 ...		

**6.6.20 modifications**

<b>Description:</b>	The small molecule's modifications or adducts. The position of the modification must be given relative to the small molecule's beginning. The exact semantics of this position depends on the type of small molecule identified. In case the position information is unknown or not applicable it should not be supplied. For detailed information see protein table.		
<b>Type:</b>	String		
<b>Mandatory</b>		Summary	Complete
	Quantification	✓	✓
	Identification	✓	✓
<b>Example:</b>	COM example where an ammonium loss is found and the position is not applicable in the given small molecule ... SMH identifier ... modifications ... SML CID:00027395 ... CHEMMOD:-NH4 ...  COM reporting adducts: sodiated glycine SMH ... formula ... charge ... modifications SML ... C2H5NO2 ... 1 ... CHEMMOD:+Na-H		

**6.6.21 smallmolecule\_abundance\_assay[1-n]**

<b>Description:</b>	The small molecule's abundance in the given assays.
<b>Type:</b>	Double

<b>Mandatory</b>		Summary	Complete
	Quantification	(✓) <sup>1</sup>	(✓) <sup>1</sup>
	Identification		
mandatory if assays are reported			
<b>Example:</b>	SMH identifier	...	smallmolecule_abundance_assay[1] ...
	SML CID:00027395	...	0.3 ...

**6.6.22 smallmolecule\_abundance\_study\_variable[1-n]**

<b>Description:</b>	The small molecule's abundance in the given study variables.		
<b>Type:</b>	Double		
<b>Mandatory</b>		Summary	Complete
	Quantification	✓	✓
	Identification		
<b>Example:</b>	SMH identifier	...	smallmolecule_abundance_study_variable[1] ...
	SML CID:00027395	...	0.3 ...

**6.6.23 smallmolecule\_abundance\_stdev\_study\_variable [1-n]**

<b>Description:</b>	The standard deviation of the small molecule's abundance in the given study variable.		
<b>Type:</b>	Double		
<b>Mandatory</b>		Summary	Complete
	Quantification	✓	✓
	Identification		
<sup>1</sup> In case the abundance for a respective study variable is given the standard deviation column MUST also be present (in case the value is not available "null" MUST be used).			
<b>Example:</b>	SMH identifier	...	smallmolecule_abundance_study_variable[1]
	smallmolecule_abundance_stdev_study_variable[1]...		
	SML CID:00027395...	0.3	0.04 ...

**6.6.24 smallmolecule\_abundance\_std\_error\_study\_variable[1-n]**

<b>Description:</b>	The standard error of the small molecule's abundance in the given study variable.		
<b>Type:</b>	Double		
<b>Mandatory</b>		Summary	Complete
	Quantification	✓	✓
	Identification		
<sup>1</sup> In case the abundance for a respective study variable is given the standard error column MUST also be present (in case the value is not available "null" MUST be used).			
<b>Example:</b>	SMH identifier	...	smallmolecule_abundance_std_error_study_variable[1] ...
	SML CID:00027395	...	0.04 ...

**6.6.25 opt\_{identifier}\_\***

<b>Description:</b>	Additional columns can be added to the end of the small molecule table. These column headers MUST start with the prefix "opt_" followed by the {identifier} of the object they reference: assay, study variable, MS run or "global" (if the value relates to all replicates). Column names MUST only contain the following characters: 'A'-'Z', 'a'-'z', '0'-'9', '_', '-', '[', ']', and ':'. CV parameter accessions MAY be used for optional columns following the format: opt_{identifier}_cv_{accession}_{parameter name}. Spaces within the parameter's name MUST be replaced by '_'.  <b>Type:</b> Column		
<b>Mandatory</b>		Summary	Complete
	Quantification		
	Identification		
<b>Example:</b>	SMH identifier	...	opt_assay[1]_my_value opt_global_another_value
	SML CID:00027395	...	My value some other value

## 7. Non-supported use cases

There are a number of use cases that were discussed during the development process and it was decided that they are not explicitly supported in mzTab version 1.0. They may be implemented in future versions of the standard.

- Sequence Tag approaches.
- Grouped modification position scoring systems.

## 8. Conclusions

This document contains the specifications for using the mzTab format to represent results from peptide, small molecule and protein identification pipelines, in the context of a proteomics investigation. This specification constitutes a proposal for a standard from the Proteomics Standards Initiative. These artefacts are currently undergoing the PSI document process, which will result in a standard officially sanctioned by PSI.

## 9. Authors

Johannes Griss, European Bioinformatics Institute, United Kingdom  
Timo Sachsenberg, University of Tübingen, Germany  
Mathias Walzer, Center for Bioinformatics, University of Tübingen, Germany  
Oliver Kohlbacher, Center for Bioinformatics, University of Tübingen, Germany  
Andrew R. Jones, University of Liverpool, United Kingdom  
Henning Hermjakob, European Bioinformatics Institute, United Kingdom  
Juan Antonio Vizcaíno, European Bioinformatics Institute, United Kingdom

Correspondence – Henning Hermjakob ([hhe@ebi.ac.uk](mailto:hhe@ebi.ac.uk)), Juan Antonio Vizcaíno ([juan@ebi.ac.uk](mailto:juan@ebi.ac.uk))

## 10. Contributors

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

- Nuno Bandeira, Center for Computational Mass Spectrometry, University of California, San Diego, CA, USA.
- Robert J. Chalkley, Department of Pharmaceutical Chemistry, University of California San Francisco, CA, USA.
- Jürgen Cox, Max Planck Institute of Biochemistry, Martinsried, Germany.
- Martin Eisenacher, Medizinisches Proteom-Center, Ruhr-Universität Bochum, Germany.
- Jun Fan, Queen Mary University of London, United Kingdom.
- Fawaz Ghali, University of Liverpool, United Kingdom.

- Laurent Gatto, University of Cambridge, Cambridge, United Kingdom.
- Jürgen Hartler, Graz University of Technology, Graz, Austria.
- Gerhard Mayer, Medizinisches Proteom-Center, Ruhr-Universität Bochum, Germany.
- Nadin Neuhauser, Max Planck Institute of Biochemistry, Martinsried, Germany.
- Yasset Pérez-Riverol, European Bioinformatics Institute, Cambridge, United Kingdom.
- Steffen Neumann, Leibniz Institute of Plant Biochemistry, Halle, Germany.
- Florian Reisinger, European Bioinformatics Institute, Cambridge, United Kingdom.
- Reza M. Salek, European Bioinformatics Institute, Cambridge, United Kingdom.
- Christoph Steinbeck, European Bioinformatics Institute, Cambridge, United Kingdom.
- Noemi del Toro, European Bioinformatics Institute, Cambridge, United Kingdom.
- Gerhard Thallinger, Graz University of Technology, Graz, Austria.
- Ioannis Xenarios, Swiss Institute of Bioinformatics, Geneva, Switzerland.
- Qing-Wei Xu, European Bioinformatics Institute, United Kingdom.

## 11. References

- Bradner, S. (1997). Key words for use in RFCs to Indicate Requirement Levels, Internet Engineering Task Force. RFC 2119.
- Martens, L., et al. (2011). "mzML--a community standard for mass spectrometry data." Mol Cell Proteomics 10(1): R110 000133.
- Jones, A. R., et al. (2012). "The mzIdentML data standard for mass spectrometry-based proteomics results." Mol Cell Proteomics doi:10.1074/mcp.M111.014381
- EA Hill (1900). "ON A SYSTEM OF INDEXING CHEMICAL LITERATURE; ADOPTED BY THE CLASSIFICATION DIVISION OF THE U. S. PATENT OFFICE." J. Am. Chem. Soc. 22 (8): 478–494. doi:10.1021/ja02046a005
- Walzer et al. The mzQuantML data standard for mass spectrometry-based quantitative studies in proteomics (2013) Mol Cell Proteomics doi: 10.1074/mcp.O113.028506.

## 12. 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).



## TradeMark Section

Microsoft Excel<sup>®</sup>

## Copyright Notice

Copyright (C) Proteomics Standards Initiative (2014). 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."