

# The ten minute guide to mzTab

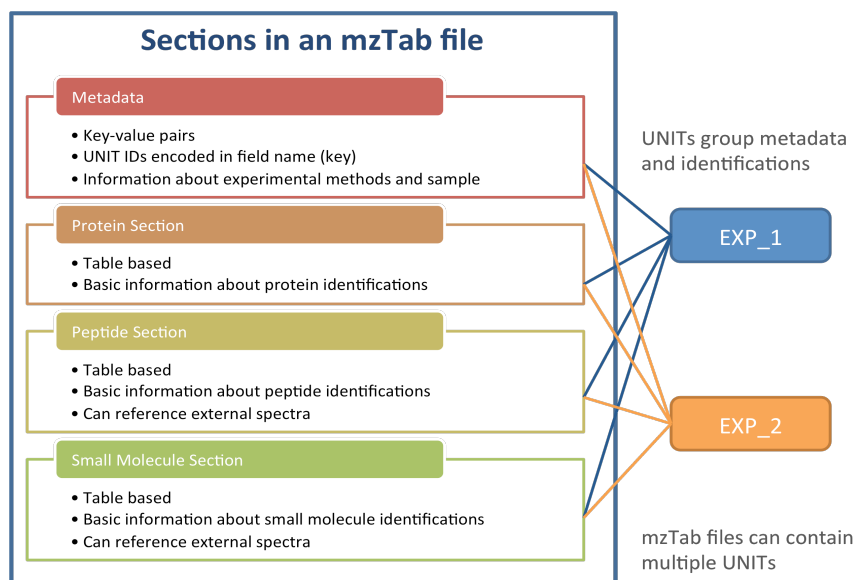
Johannes Griss & Juan Antonio Vizcaíno, EBI, [juan@ebi.ac.uk](mailto:juan@ebi.ac.uk), May 2012

## Introduction

The purpose of this guide is to give a quick introduction on how to use mzTab efficiently. It is targeted at both, developers and end-users alike. This guide is not intended to give a complete and detailed overview of mzTab but should only be a quick and easy to understand introduction. The complete format specification as well as example files can be found at <http://mztab.googlecode.com>.

## Basic structure

mzTab files can have four sections: The metadata section, the protein section, the peptide section, and the small molecule section (see Figure 1). All of these sections are optional and may not be present in every file.



**Figure 1:** Basic structure of an mzTab file.

All lines in an mzTab file start with a three letter code to identify the information held by the line:

<b>MTD</b>	for metadata
<b>PRH</b>	for the protein table header line (the column labels)
<b>PRT</b>	for rows of the protein table
<b>PEH</b>	for the peptide table header line (the column labels)
<b>PEP</b>	for rows of the peptide table
<b>SMH</b>	for small molecule table header line
<b>SML</b>	for rows of the small molecule table
<b>COM</b>	for comment lines

*For developers:*

mzTab is a tab separated file format. The three letter codes must be separated by a tab from the next field. Also, field names and values in the metadata section are separated by tabs as are the columns in the table based sections.

## Units in mzTab

Every identification and metadata field is assigned to a unit. A unit is only an identifier that groups these pieces of information together. The only limitation units have is that a protein must be unambiguously identified by its protein accession in the same unit. Units are not reported in a separate table or section but only exist through their Unit\_IDs, which are referenced by every metadata field and identification. One mzTab file can contain multiple units that can represent different concepts depending on the experimental design and the granularity needed by the data producer to communicate the results. Some examples:

- The results from one technical replicate (one LC-MS run) or the summary value after all replicates have been combined. Technical replicates can be explicitly reported in mzTab and grouped together by using special "Unit\_IDs" in case this level of detail is required. To report the results from an experimental setup (labelled as "EXP\_1") containing the data from two replicates as well as the final combined results the resulting mzTab file would contain three units: the replicates as "EXP\_1\_rep[1]" and "EXP\_1\_rep[2]", and the final results as "EXP\_1".

- The summary results of all experiments from one dataset submitted to a proteomics repository. For example, the example file "PXD000002 submission" represents the final results from one single dataset but it contains three units: "PRIDE\_22142", "PRIDE\_22143", and "PRIDE\_22144". Each of these represents the results of the corresponding PRIDE experiment (equivalent to one MS run). Thereby, a researcher can easily get an overview of the results of a whole submission by only looking at a single file. Another example of one unit in this context could be one complete PeptideAtlas build.

```
COM  Example showing the usage of UNIT IDs
MTD  EXP_1-title    The first experiment in the file
...
MTD  EXP_2-title    The second experiment in the file
...
PRH  accession      unit_id    description                taxid    species    ...
PRT  P02042         EXP_1      Hemoglobin subunit delta    9606     Homo sapie...
PRT  P02042         EXP_2      Hemoglobin subunit delta    9606     Homo sapie...

PEH  sequence       accession  unit_id    unique    ...
PEP  ABC            P02042    EXP_1      0         ...
PEP  CDE            P02042    EXP_1      1         ...
PEP  ABC            P02042    EXP_2      0         ...
PEP  FGH            P02042    EXP_2      0         ...
```

### *For developers:*

Unit\_IDs are freely generated by the software that generated the mzTab file and should somehow sensibly identify the, for example, experiment. If mzTab files are generated by local software tools, these UNIT\_IDs can be any sensible identifier for the, for example, experiment. Unit\_IDs must only contain the following characters: 'A'-'Z', 'a'-'z', '0'-'9', and '\_'.

## Metadata in mzTab

The metadata section in mzTab files contains information about the units and consists of key - value pairs separated by a tab. The name of every field contains the Unit\_ID. Thereby, every field can be attributed to a unit. A complete list of available fields can be found in the specification document.

```
COM Example showing the use of the metadata section
MTD EXP_1-title The unit's / experiment's title
MTD EXP_1-description This is just an example. No description needed...
MTD EXP_1-instrument[1]-source [MS, MS:1000073, ESI,]
MTD EXP_1-instrument[1]-analyzer [MS, MS:1000291, linear ion trap,]
MTD EXP_1-instrument[1]-detector [MS, MS:1000253, electron multiplier,]
MTD EXP_1-software[1] [MS, MS:1001207, Mascot, 2.3]
MTD EXP_1-software[2] [MS, MS:1001561, Scaffold, 1.0]
MTD EXP_1-false-discovery-rate [MS, MS:1001364, pep:global FDR, 0.01]
MTD EXP_1-contact[1]-name James D. Watson
MTD EXP_1-contact[1]-affiliation Cambridge University, UK
MTD EXP_1-contact[1]-email watson@cam.ac.uk
```

## Proteins in mzTab

Protein identifications are reported in the protein section. The protein section is table based. The table header is identified by the prefix "PRH", entries in the protein table are identified through "PRT". The protein section must only be present once but can contain identifications from multiple units. Columns are separated by a tab.

```
COM Example of the protein section. Other sections are omitted
PRH accession unit_id description taxid species database ...
PRT P12345 EXP_1 mAspAT 9986 Rabbit UniProtKB/SwissProt ...
PRT P02042 EXP_2 Hemoglobin 9606 Human UniProtKB/SwissProt ...
```

All columns in the protein section are mandatory, except the quantitative and optional columns. The full list of columns can be found in the specification document.

## Peptides in mzTab

The peptide section is similar to the protein section. It is table based, columns are separated by a tab and all columns are mandatory apart from quantitative and optional columns. The header of the peptide table is indicated by "PEH", and entries in the table by "PEP". The peptide section must also be present only once but may contain identifications from multiple units. The full list of columns can be found in the specification document.

```
COM Example of the peptide section. Other sections are omitted.
PEH sequence accession unit_id unique ... search_engine ...
PEP ABC P12345 EXP_1 0 ... [MS,MS:1001207,Mascot,] ...
PEP ABC P12345 EXP_2 0 ... [MS,MS:1001208,Sequest,] ...
```

## Small Molecules in mzTab

The small molecule section is also a table based section (same rules apply). Small molecules are identified through an "identifier" in mzTab. This identifier can be any text that sensibly identifies the given small molecule in the given field of research. These identifiers should generally be entries in compound databases used in the respective field (for example, Human Metabolome Database entries, ChEBI identifiers, PubChem IDs or LIPID MAPS IDs). Apart from this identifier, small molecules can be assigned a chemical formula, SMILES and/or InChi identifier, a human readable description, a precursor *m/z* value, a charge state, retention time(s), a species, source database and search engine including score. We are aware, that these fields are not applicable to all

fields of metabolomics, but we believe that they represent a sensible selection. A more detailed list of possible columns can be found in the specification document.

```
COM Example of the small molecule section. Other sections are omitted.
SMH identifier unit_id chemical_formula description mass_to_charge charge ...
SML TG54:0 EXP_1 H110O6C57 - 949.892 1
```

## Missing values

The table-based sections (protein, peptide, and small molecule) must not contain empty cells. In case a given property is not available for an entry “NA” should be reported.

Any calculation that results in “not a number” (NaN type) must also be reported using “NA”. If ratios are included and the denominator is zero, the “INF” value must be given. 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 (“NA”).

## Reliability score

All protein, peptide and small molecule identifications reported in an mzTab file should be assigned a reliability score (column “reliability” in all tables). The idea is to provide a way for researcher and/or MS proteomics or metabolomics repositories or data producers to score the reported identifications based on their own criteria. This score is completely resource-dependent and must not be seen 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, “NA” must be provided as the value for each of the protein, peptide or small molecule identifications.

The reliability must be an integer between 1-3 and should be interpreted as follows:

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

The idea behind this score is 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. 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, for example 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.

## Quantitative Data

There are multiple quantification techniques available for MS-based experiments that often result in slightly different types of data. mzTab was not designed to capture any of these specific differences. The goal for mzTab was to provide a generic view on quantitative MS-based identification data that is applicable to as many different quantitation methods as possible.

Quantitative technologies generally result in some kind of abundance measurement of the identified analyte. Several of the available techniques furthermore allow/require multiple similar samples to be multiplexed and analyzed in a single MS run. When several biological samples are multiplexed these samples are referred to as “subsamples” in mzTab. Subsamples must furthermore be linked to the used labels in the metadata section of the mzTab file (see example below). In case a quantification method is used that does not lead to multiplexed biological samples, the generated quantification values are reported as subsample 1. Detailed information about how to report subsamples in the metadata section can be found in the specification document.

```
COM The following example shows how two different quantitative experiments
COM can be reported in one mzTab file. Not all labels are shown
...
MTD EXP_1-quantification_method [MS,MS:1001837,iTraq,]
MTD EXP_1-sub[1]-description Healthy human liver tissue
MTD EXP_1-sub[1]-quantification_reagent [PRIDE,PRIDE:0000114,iTRAQ reagent 114,]
MTD EXP_1-sub[2]-description Human hepatocellular carcinoma sample.
MTD EXP_1-sub[2]-quantification_reagent [PRIDE,PRIDE:0000115,iTRAQ reagent 115,]
...
MTD EXP_2-quantification_method [MS,MS:100999,SILAC,]
MTD EXP_2-sub[1]-description Healthy rat liver tissue
MTD EXP_2-sub[1]-quantification_reagent [PRIDE,PRIDE:0000325,SILAC heavy,]
MTD EXP_2-sub[2]-description Intoxicated rat liver.
MTD EXP_2-sub[2]-quantification_reagent [PRIDE,PRIDE:0000326,SILAC light,]
...
PRH accession unit_id ... protein_abundance_sub[1] ... protein_abundance_sub[2] ...
PRT P12345 EXP_1 ... 1 ... 0.82749
PRT P15151 EXP_2 ... 2.42114 ... 1
...
```

MS<sup>2</sup> spectral counting-based approaches can be reported using optional columns in the peptide table as well as the protein table as they only result in one single value per analyte. In case the approach used also generates standard deviation and standard errors the quantification results may also be reported using the subsample 1 columns. MS label-free quantification techniques do not require any additional support in mzTab as they simply need to report abundance values per sample in a straight-forward manner. CV parameter accessions may be used as optional column names following the following format: opt\_cv\_{accession}.

```
COM Example showing how emPAI values are reported in an additional column using
COM MS CV parameter "emPAI value" (MS:1001905)
...
PRH accession ... opt_cv_MS:1001905
PRT P12345 ... 0.658
```

## Protein Inference

There are multiple approaches to report protein inference. mzTab is designed to only hold experimental results which in proteomics experiments can be very complex. At the same time, for down-stream 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 “mention” the protein inference problem in mzTab files but not provide detailed information on how it was resolved. Protein entries in mzTab files contain the field “ambiguity\_members”. The protein accessions listed in this field should identify proteins that could also be identified through the same (sub-)set of peptides but were not chosen as the primary identification. The members of the ambiguity group are not reported in the peptide table for the respective unit.

```

COM  In the following example only one peptide was identified that can be attributed to
COM  multiple proteins. The choice which one to pick as primary accession depends on
COM  the resource generating the mzTab file.
...
PRH  accession  unit_id  ...  ambiguity_members
PRT  P19012     EXP_1    ...  P13646, P08779, P02533, Q7Z3Z0, Q7Z3Y9, Q7Z3Y8 ...
...
PEH  sequence   accession  unit_id ...
PEP  ALEENADLEVK  P19012    EXP_1    ...

```

## Advanced topics

There are several other features in mzTab that could not be introduced here. Detailed information about these features can be found in the specification document such as:

- Reporting post-translational modifications including modification position ambiguity.
- Reporting results from multiple search engines.
- Reporting replicates in mzTab.
- Merging mzTab files.
- Referencing external spectra.
- Referencing external resources such as mzIdentML or mzQuantML files.
- Adding optional columns.

An up-to-date list of example files can be found at <http://code.google.com/p/mztab/wiki/ExampleFiles>. The specification document can be found at <http://code.google.com/p/mztab/>.