In order to save on disk-space, it can be chosen to store the complete file as a GZIP compressed file. This is supported by the reference parser and automatically detected. In view of legacy applications the peakml file format stores ionization corrected masses (*i.e.* for positive ionization a proton is deducted).

Element <peakml>

This is the root element of the peakml file format, capturing extracted downstream mass spectrometry data.

Type:
-

Attributes:

Name	Type	Use	Definition
version	xs:string	required	The version of this PeakML format, which is setup as a major, minor and maintenance number. The maintenance denotes small internal changes, not breaking functionality. The minor denotes non-required additions, extending functionality. The major denotes major changes, breaking functionality.

Subelements:

Name	Min	Max	Definition
header	1	1	Container for meta information, describing the contents of the file.
peaks	1	1	Container for the data.

Element <header>

Container for meta information, describing the contents of the file. This information can be used by automatic processing information to interpret the contents of the file and take appropriate action. For example, the header provides a mechanism how multiple measurements (*e.g.* biological replicates) are set up in sets, allowing visualization software to color code mass traces according to the set they belong to.

Type:
-

Subelements:

Name	Min	Max	Definition
nrpeaks	1	1	Contains the number of peaks actually stored in the file.
date	1	1	Contains the creation date of the file.
owner	0	1	Optional owner of the file.
description	0	1	Optional description of the contents of the file.
annotations	0	1	Optional annotations as free text; optionally linked to a controlled vocabulary.
measurements	1	1	Required container for description(s) of the measurement(s) contained in the file.
applications	0	1	Optional container for description of the applications used to create the file.
samples	0	1	Optional container for description of the samples used for the measurements.
sets	0	1	Optional container for description of the manner in which the measurements are set up in sets (*e.g.* multiple biological replicates).

Element <nrpeaks>

Contains the number of peaks actually stored in the file. The data-element always contains a non-negative integer, which can be used by software to visualize the progress of loading the contents of the file.

Type:
xs:nonNegativeInteger

Element <date>

Contains the creation date of the file. This date is stored as YYYY-MM-DD.

Type:
xs:date

Element <owner>

Optional owner of the file.

Type:
xs:string

Element <description>

Optional description of the contents of the file.

Type:
xs:string

Element <annotations>

Optional annotations as free text; optionally linked to a controlled vocabulary.

Type:
-

Subelements:

Name	Min	Max	Definition
annotation	1	n	Container for a single annotation as a label-value pair; optionally linked to a controlled vocabulary.

Element <annotation>

Container for a single annotation as a label-value pair; optionally linked to a controlled vocabulary.

Type:
-

Attributes:

Name	Type	Use	Definition
ontologyref	xs:string	optional	A reference to the ontology this annotation is linked to. This reference should be the namespace of the ontology, like MS for the PSI MS ontology.

unit	xs:string	optional	An optional specifier for the unit of the value denoted in the annotation.

Subelements:

Name	Min	Max	Definition
label	1	1	The unique label of the annotation.
value	1	1	The value of the annotation.
valuetype	0	1	Optional value-type for non-controlled annotations, helping processing software to automatically interpret the value (*e.g.* for sorting).

Element <label>

The unique label of the annotation. Annotations are stored as a hashmap, allowing a single value per label.

Type:
xs:string

Element <value>

The value of the annotation.

Type:
xs:string

Element <valuetype>

Optional value-type for non-controlled annotations, helping processing software to automatically interpret the value (e.g. for sorting).

Type:
xs:string – STRING | INTEGER | DOUBLE

Element <measurements>

Required container for description(s) of the measurement(s) contained in the file. The purpose of the PeakML file format is to store grouped mass traces from multiple measurements. The meta information on these measurements supports automatic processing software to interpret the contents.

Type:
-

Subelements:

Name	Min	Max	Definition
measurement	1	n	Container for describing a single measurement.

Element <measurement>

Container for describing a single measurement. Usually a single raw-file contains all the data and meta information of a measurement, of which the meta data is reflected in this element.

Type:

-

Subelements:

Name	Min	Max	Definition
id	1	1	The unique id of the measurement, as an integer.
label	1	1	A label best describing the contents of the file.
sampleid	0	1	An optional sample-id linked to this measurement.
scans	0	1	Optional block describing all the scans of the measurement.
files	1	1	Block describing all the files, of which the first the original RAW-file should be.
annotations	0	1	Optional annotations as free text; optionally linked to a controlled vocabulary.

Element <id>

The unique id of the measurement, as an integer. The integer representation was chosen to reduce the storage space, as the peak elements mass chromatogram and background ion contain this id linking them to the measurement.

Type:
xs:nonNegativeInteger

Element <label>

A label best describing the contents of the file. This label is meant for visualization purposes only and is not required to be unique.

Type:
xs:string

Element <sampleid>

An optional sample-id linked to this measurement. This id can be used to look up the associated sample information in the sample block.

Type:
xs:string

Element <scans>

Optional block describing all the scans of the measurement.

Type:
-

Subelements:

Name	Min	Max	Definition
scan	1	n	Container for describing a single scan.

Element <scan>

Optional block describing a scan. It has been chosen to store this information in xml-blocks, as it is non-repeated information which can potentially be extended over time.

Type:
-

Subelements:

Name	Min	Max	Definition
polarity	1	1	The polarity used for the scan.
retentiontime	1	1	The time in seconds when the scan was made.
annotations	0	1	Optional annotations as free text; optionally linked to a controlled vocabulary.

Element <polarity>

The polarity used for the scan. This value can either be: POSITIVE, NEGATIVE or NEUTRAL.

Type:
xs:string

Element <retentiontime>

The time in seconds when the scan was made.

Type:
xs:string

Element <files>

Block describing all the files, of which the first the original RAW-file should be.

Type:
-

Subelements:

Name	Min	Max	Definition
file	1	n	Container for describing a file.

Element <file>

Container for describing a file.

Type:
-

Subelements:

Name	Min	Max	Definition
label	1	1	A friendly label for the file.
name	1	1	The name of the file.
location	1	1	The location of the file (*i.e.* the directory it is located).

Element <label>

A friendly label for the file.

Type:

xs:string

Element <name>

The name of the file.

Type:
xs:string

Element <location>

The location of the file (i.e. the directory it is located).

Type:
xs:string

Element <applications>

Optional container for description of the applications used to create the file.

Type:
-

Subelements:

Name	Min	Max	Definition
application	1	n	

Element <samples>

Optional container for description of the samples used for the measurements.

Type:
-

Subelements:

Name	Min	Max	Definition
sample	1	n	Container for the sample description.

Element <sample>

Container for the sample description.

Type:
-

Subelements:

Name	Min	Max	Definition
id	1	1	Unique id of the sample
annotations	0	1	Optional annotations as free text; optionally linked to a controlled vocabulary.

Element <id>

Unique id of the sample.

Type:
xs:string

Element <sets>

Optional container for description of the manner in which the measurements are set up in sets (e.g. multiple biological replicates).

Type:
-

Subelements:

Name	Min	Max	Definition
set	1	n	Container for a set description.

Element <set>

Container for a set description

Type:
-

Subelements:

Name	Min	Max	Definition
id	1	1	
type	1	1	
measurementids	1	1	

Element <id>

Unique id of the set.

Type:
xs:string

Element <type>

The type of the set.

Type:
xs:string – TECHNICAL | BIOLOGICAL | SET

Element <measurementid>

Base64 encoded array 32 bits in little endian format, containing the measurement-id's linked to the set.

Type:
xs:string

Element <peaks>

Container for the peak entries, containing the mass trace information. This element will contain the bulk of the data.

Type:
-

Subelements:

Name	Min	Max	Definition
peak	1	n	Container for the data of a single peak entry.

Element <peak>

Container for the data of a single peak entry. The peak entry contains all the data associated to the peak, which can be of 3 different types. The *backgroundion* and *masschromatogram* types use the peakdata structure to store the mass trace data, while the *peakset* type uses a recursive peak entry to store other peaks. This means that we have a fully extendable setup.

Type:
-

Attributes:

Name	Type	Use	Definition
type	xs:string	required	Indication of the type of the peak. Three values are supported at this point: masschromatogram, backgroundion and peakset. The first subelements collect some compound information about the peak, which can be used to load only a flat peak table (considerably shortening loading times).

Subelements:

Name	Min	Max	Definition
scanid	1	1	The id of the scan where this peak was most intense.
retentiontime	1	1	The retention time where this peak was most intense.
mass	1	1	The mass of this peak.
intensity	1	1	The intensity of this peak.
measurementid	1	1	The measurementid this peak has been linked to. In the case of type=*peakset* this can be set to -1.
annotations	0	1	Optional annotations as free text; optionally linked to a controlled vocabulary.
peak	0	1	Recursive link to more peak entries in the case of type *peakset*.
peakdata	0	1	Block containing the trace and meta data in the case of type

Element <scanid>

The id of the scan where the trace is most intense. In the case of type *peakset* this is compounded from all the mass traces contained in the set.

Type:
xs:integer

Element <retentiontime>

The retentiontime where the trace is most intense. In the case of type *peakset* this is compounded from all the mass traces contained in the set.

Type:
xs:double

Element <mass>

The mass of the entry. In the case of type *peakset* this is compounded from all the mass traces contained in the set.

Type:
xs:double

Element <intensity>

The intensity of the entry. In the case of type *peakset* this is compounded from all the mass traces contained in the set.

Type:
xs:double

Element <measurementid>

The id of the measurement this peak entry is linked to. If the type of the peak is *peakset* this should be set to -1.

Type:
xs:integer

Element <peakdata>

Container for the mass trace data of either a *backgroundion* or a *masschromatogram*. All this data is stored as base64 encoded arrays in order to reduce the dataload, which are always stored in 32 bits precision in little endian ordering.

Type:
-

Attributes:

Name	Type	Use	Definition
type	xs:string	required	The type of the data stored in this element. For now only *centroid* is supported, but this provides extensibility to *profile* data.
size	xs:integer	required	The size of the arrays. This size is required to hold for all the arrays stored in this element.

Subelements:

Name	Min	Max	Definition
scanids	1	1	The array containing all the scanid's of the trace.
retentiontimes	1	1	The array containing all the retentiontimes of the trace.
masses	1	1	The array containing all the masses of the trace.
intensities	1	1	The array containing all the intensities of the trace.
measurementids	1	1	The array containing all the measurementids of the trace.