
PRIDE Converter 2 Command-Line User Guide

Table of Contents

1. Introduction	2
2. PRIDE Converter 2 Requirements	2
2.1. System Requirements	2
2.2. Additional Requirements	2
3. <i>PRIDE Converter 2</i> CLI User Guide	3

1. Introduction

The PRIDE Converter 2 tool suite is composed of 4 independent applications:

- The **PRIDE Converter 2** application will convert MS search result files containing identification and spectra into PRIDE XML.
- The **PRIDE mzTab Generator** will produce skeleton mzTab files from MS search results files. These skeleton files require either manual or scripted editing to add quantitation and/or gel information.
- The **PRIDE XML Filter** will remove identifications or spectra from PRIDE XML files based on a series of configurable filters.
- The **PRIDE XML Merger** will combine several PRIDE XML files into a single one.

All tools have both a Graphical User Interface (GUI) and a Command-Line Interface (CLI). The GUIs have been designed to provide a rich, user-friendly interface while the CLIs have been developed mainly for tool and pipeline developers to be able to integrate the PRIDE Converter 2 tools in their own software to provide an efficient way to generate PRIDE XML from their own resources.

2. PRIDE Converter 2 Requirements

2.1. System Requirements

- Java: JRE 1.5 +
- CPU: 1 gigahertz (GHz) or faster 32-bit or 64-bit processor.
- Memory: 1 gigabyte (GB) RAM.
- Hard Disk: 55 MB available for installation, more if doing file conversions.
- Platform: Tested in Mac OS X, Linux, and Windows (XP, Vista, 7).

2.2. Additional Requirements

PRIDE Converter 2 requires Internet access for connection to the Ontology Lookup Service (OLS) web service and automatic access to PubMed records already published.

3. PRIDE Converter 2 CLI User Guide

Using the PRIDE Converter CLI is a multi-step process. Given the enormous complexity and heterogeneity of the data that *PRIDE Converter 2* is trying to capture, it was basically impossible to design a command-line structure that would be suitable to task. Therefore, *PRIDE Converter 2* is designed to work in two modes, *prescan* and *conversion*.

By default, if the converter is launched from a command-line prompt without arguments, the GUI will start. In order to start the CLI, arguments must be provided and, if unsure of what arguments to use, users can always use ‘-help’ to obtain on-screen assistance:

```
C:\pride-converter>java -jar pride-converter-2.0-SNAPSHOT.jar -help
Usage: java -jar pride_converter.jar [mode]

PRIDE Converter 2
PRIDE Converter can operate in several different modes.
To launch the graphical mode simply specify no parameters.

The following modes are supported by PRIDE Converter:
  -converter    Launches the PRIDE Converter in convert mode.
                 This mode allows one to convert several search
                 engine result files into the PRIDE XML format.
  -filter       Launches the PRIDE Converter in filter mode.
                 This mode allows one to remove f.e. unidentified
                 spectra or a given set of proteins from an existing
                 PRIDE XML file.
  -merger       Launches PRIDE Converter in merger mode.
                 This mode allows one to merge several PRIDE XML files
                 into a single file.

For mode specific help please use java -jar pride_converter.jar [mode] -help
```

Note that there is no mzTab generator tool. To generate mzTab files, use the converter tool and use ‘-mode mzTab’. To obtain more information for a specific tool, simply follow the instructions. For example, the converter tool:

```
C:\pride-converter>java -jar pride-converter-2.0-SNAPSHOT.jar -converter -help
usage: PrideConverter -converter [-compress] [-D <property=value>]
  [-debug] [-engine <engine>] [-fastafile <file>] [-fastaformat
  <format>] [-gel_identifier <gel identifier>] [-gel_spot_identifier
  <spot identifier>] [-gel_spot_regex <regular expression>]
  [-generate_quant_fields <nr. of reagents>] [-help] [-mode <mode>]
  [-mztabfile <file>] [-outputfile <file>] [-reportfile <file>]
  [-reportOnlyIdentifiedSpectra] [-sourcefile <file>] [-spectrafile
  <file>] [-submit_to_intact] [-useHybridSearchDatabase
  <useHybridSearchDatabase>] [-version]

Note that -mode, -engine and -sourcefile are required parameters for conversion.

  -compress                turn on gzip
                           compression for
                           output file

  -D <property=value>      use value for given
                           property. If passing
                           engine-specific
                           options, this should
                           only be used with
                           -mode=PRESCAN. In
                           mode=SCAN,
                           engine-specific
                           configuration
                           options are parsed
                           from the report
                           file.

  -debug                   print debugging
                           information
```

<code>-engine <engine></code>	search engine. Must be one of the following values: [MASCOT, MGF, DTA, PKL, MS2, mzML, XTandem, mzIdentML, mzXML, mzData, MSGF, crux_txt, SpectraST, OMSSA]
<code>-fastafile <file></code>	full path and filename of FASTA file used as a search database
<code>-fastaformat <format></code>	The format of the FASTA id line. OPTIONAL. Must be one of [FULL, UNIPROT_MATCH_ID, UNIPROT_MATCH_AC, FIRST_WORD]. Defaults to FULL
<code>-gel_identifier <gel identifier></code>	sets the gel identifier to be used for identifications in the generated mzTab file. This option only takes effect when generating mzTab files.
<code>-gel_spot_identifier <spot identifier></code>	sets the gel spot identifier to be used for identifications in the generated mzTab file. This option only takes effect when generating mzTab files. This option is ignored if gel_spot_regex is set.
<code>-gel_spot_regex <regular expression></code>	used to extract the gel spot identifier based on the sourcefile's name. The first matching group in the pattern is used as a spot identifier.
<code>-generate_quant_fields <nr. of reagents></code>	adds (empty) quantitative fields to the generated mzTab file for the number of specified reagents.
<code>-help</code>	print this message. If combined with -engine, will also output engine-specific options
<code>-mode <mode></code>	The mode in which to run PrideConverter. Must be one of the following values: [PRESCAN, CONVERT, MZTAB]
<code>-mztabfile <file></code>	full path and filename of mzTab file
<code>-outputfile <file></code>	full path and filename of PRIDE XML output file. OPTIONAL. Will default to <sourcefile>.xml.gz

<code>-reportfile <file></code>	full path and filename of report file. OPTIONAL. Will default to <code><sourcefile>-report.xml</code>
<code>-reportOnlyIdentifiedSpectra</code>	Indicates that only identified spectra should be reported in the generated PRIDE XML file.
<code>-sourcefile <file></code>	full path and filename of source file.
<code>-spectrafile <file></code>	overwrites the path to the spectrum file(s) with the set value. This can either specify a directory containing multiple MS data files referenced in the search result file or one MS data file directly depending on the file format.
<code>-submit_to_intact</code>	Indicates that the generated XML file contains interaction data that should be submitted to IntAct
<code>-useHybridSearchDatabase <useHybridSearchDatabase></code>	Indicates if the search database contains a combination of valid and decoy protein sequences. Must be [TRUE FALSE]. Defaults to TRUE.
<code>-version</code>	print the version information and exit

PRIDE Converter Tool suite 2.0-SNAPSHOT-20120621-1200

To obtain DAO-specific help, when and if DAO-specific options are available, add `-engine [ENGINE_NAME]` to the command-line. For example, for the Mascot DAO:

```
C:\pride-converter>java -jar pride-converter-2.0-SNAPSHOT.jar -converter -engine mascot -help
```

This command would display the help information as shown above, plus the following information:

usage: Mascot engine options

use the `-Dproperty=value` syntax to use these options

<code>-compatibility_mode</code>	If set to true (default) the precursor charge will also be reported at the spectrum level using the best ranked peptide's charge state. This might lead to wrong precursor charges being reported. The correct charge state is always additionally reported at the peptide level.
<code>-decoy_accession_prefix</code>	An accession prefix that identifies decoy hits. Every protein with an accession starting with this precursor will be flagged as decoy hit. Furthermore, any decoy hit who's accession does not start with this prefix will be altered accordingly.

<code>-enable_protein_grouping</code>	Indicates whether the grouping mode (Occam's Razor, see Mascot documentation) should be enabled. This is the default behaviour for Mascot. This mode is not equivalent to the protein clustering introduced in Mascot 2.3.
<code>-homology_threshold</code>	If set to true (default is "false" the homology instead of the identity threshold will be used to identify significant identifications.
<code>-ignore_below_ions_score</code>	Peptides with a lower expect ratio (of being false positives) will be ignored completely. Set to 1 to deactivate. Default value is 0.0
<code>-include_error_tolerant</code>	Indicates whether integrated error tolerant search results should be included in the PRIDE XML support. These results are not included in the protein scores by Mascot.
<code>-min_probability</code>	Specifies a cut-off point for protein scores, a cut-off for an Integrated error tolerant search and a threshold for calculating MudPIT scores. This value represents a probability threshold.
<code>-only_significant</code>	Indicates whether only significant peptides / (in PMF searches) proteins should be included in the generated PRIDE file.
<code>-remove_duplicates_different_query</code>	Indicates whether duplicate peptides having the same sequence (but maybe different modifications) coming from different queries (= spectra) should be removed.
<code>-remove_duplicates_same_query</code>	Indicates whether duplicate peptides having the same sequence and coming from the same query (= spectrum) should be removed. These peptides may have different modifications reported.
<code>-remove_empty_spectra</code>	If set to true (default) spectra without any peaks are ignored and not reported in the PRIDE XML file.
<code>-use_mudpit_scoring</code>	Indicates whether MudPIT or normal scoring should be used.

The *prescan* will generate a report file that contains placeholders for all of the data that requires annotation intervention (software, sample, protocol, instrumentation, PTMs, etc). It is expected that pipeline maintainers will develop their own code to update the report files with their own metadata and then run PRIDE Converter in *conversion* mode, to generate fully-annotated PRIDE XML files.

For most users, the *PRIDE Converter 2* GUI will handle all of the report annotation, but it is essentially working in the same fashion, while doing most of the file I/O in the background. It generates a report file in the background, and then presents the users with a form-based wizard to capture the metadata. It then updates the report file and runs in conversion mode to generate PRIDE XML.

Users who wish to integrate the PRIDE Converter CLI into their own applications are encouraged to read the Developer Guide section entitled "Report File Manual Annotation Guidelines".