

EXMARaLDA Add-In for MS Excel - Version 0.9.5.7

1 Summary

The EXMARaLDA Add-In for MS Excel is a freely available (GPL) plugin written using the Office VBA API which allows users of Excel to import data from the EXMARaLDA XML format into a spreadsheet (preserving cells and spans, as well as most types of metadata – see below for details) and back again from a spreadsheet to XML.

This add-in is compatible with Office XP, 2003 and 2007 under Windows XP, Vista or Windows 7 and comes with absolutely no warranty. The latest public version of the add-in can be found on <http://exmaralda.org>.

New in this release:

- Experimental exporter to PAULA XML (may be expanded into a separate add-in in future versions)
- Batch Importer/Exporter written by Marc Reznicek
- Function to add spaces in empty cells in the first column by Marc Reznicek
- Fixed number formatting (numbers are imported as text and not rounded etc. by Excel)
- More tolerance to order of attributes in events (bug fix)
- Correct handling of UTF-8 in metadata (bug fix)
- Correct import of consecutive files under each other (bug fix)

2 Installing

To install the add-in follow these steps:

1. Copy the file *exmaralda_io_0.9.5.7.xla* to the directory that holds your Excel add-ins. On an English language Windows XP installation this is usually:

```
C:\Documents and Settings\YOUR_USER_NAME\application
data\Microsoft\AddIns
```

Other languages may have slightly different paths (e.g. “Dokumente und Einstellungen” for “Documents and Settings” on German Windows).

2. Open Excel and choose *Tools -> Add-Ins...* (again, other languages may use different names, e.g. German *Extras* for *Tools* etc).
3. Click *Browse...* and navigate to the file *exmaralda_io_0.9.5.7.xla*. Once the file is chosen, the add-in *Exmaralda_Io_0.9.5.7* should appear on the list of add-ins.
4. Check the box next to *Exmaralda_Io_0.9.5.7* and click OK. A new menu called *Exmaralda* should now appear in the menu bar above.

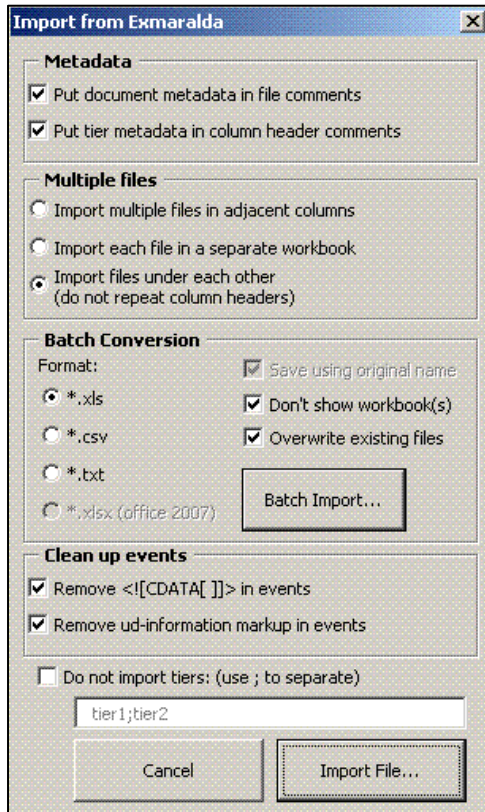
3 Uninstalling

To remove the add-in simply go to *Tools -> Add-Ins* and uncheck *Exmaralda_Io_0.9.5.7*. By pressing OK the Exmaralda menu will now be removed. You may then optionally delete the file *exmaralda_io_0.9.5.7.xla* if you wish.

4 Usage

The Exmaralda Add-In menu in Excel provides four basic functions as detailed below.

4.1 Import from Exmaralda



Choosing this function will open the Import Form.

Metadata

The form allows users to choose whether or not metadata applying to an entire EXMARaLDA document will be imported, and similarly for metadata applying to each tier within a document. Document metadata is stored in the Excel file's *Comments* property, which keeps the entire header of the original XML file. Tier metadata is stored in a comment to the header cell of each column. Note that at present **no metadata contained in <ud-tier-information> and <ud-information> tags within a <tier> element is imported** (support for this will probably be added in a future version. A further type of element which is ignored by the importer at present is the **<tierformat-table> of .exb files**, meaning that formatting information such as background colors for each tier etc. are not carried over.

Import Behavior and Multiple Files

Each tier is imported in a separate column in Excel, so that horizontal layers from EXMARaLDA are represented vertically as in the image below (this is because Excel can only accommodate 256 columns; see "Transposing" below for an EXMARaLDA style horizontal view). The user may determine where imported data will appear by selecting a cell or cells – data will then be imported starting from the top-leftmost selected cell. Each tier's *category* attribute is used as a column header in bold.

	0	1	2	3	4	5	6
[word]	Dieser	Text	kommt	aus	dem	Buch	"
[lemma]	dies	Text	kommen	aus	d	Buch	"
[pos]	PDAT	NN	VVFIN	APPR	ART	NN	\$(
[matrix-satz_felder]	VF_MS	LSK_MS_1	MF_MS_1				

	A	B	C	D
1	word	lemma	pos	matrix-satz_felder
2	Dieser	dies	PDAT	
3	Text	Text	NN	VF_MS
4	kommt	kommen	VVFIN	LSK_MS_1
5	aus	aus	APPR	
6	dem	d	ART	
7	Buch	Buch	NN	

When importing multiple files, users can select whether the columns representing the annotation levels of each document should be imported side by side, each document in a separate Excel workbook, or continuously underneath each other (usually only suitable if all documents have precisely the same layers in the same order). When documents are imported underneath each other, a header row is only generated for the first document, and the other documents are assumed to have exactly the same columns. If multiple files are imported into the same workbook and importing of document metadata is still selected, only the first document's header will be inserted into the *Comments* property.

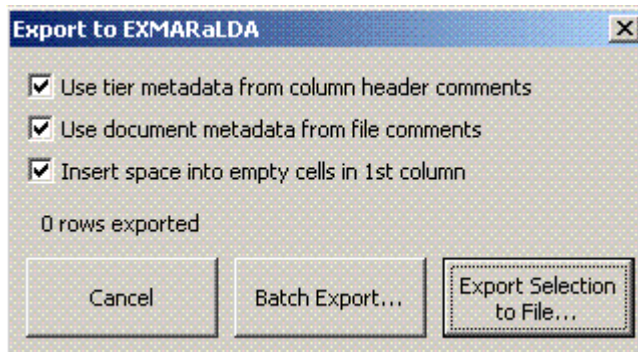
Batch Conversion

To convert multiple files to Excel use the Batch Import option. Files can be saved as plain text (tab delimited), comma separated values (csv), Excel format (.xls), and in Excel 2007, also in the new .xlsx format. Currently the file name is simply retained and the new extension is substituted for the original one. You can also choose not to show the result in Excel ("don't show workbooks") and whether or not existing files should be overwritten.

Clean Up

Users can also select to clean up the input XML by removing `<![CDATA[]>` tags and ud-information markup within events (formatting within individual EXMARaLDA cells). Finally, it is also possible to omit certain layers that should not be imported. To do so, select *do not import tiers* and input the names of the tier categories that should be ignored, separated by a semicolon, e.g. *lemma;pos*

4.2 Export to Exmaralda



A selected range of cells may be exported to EXMARaLDA XML using the *Export to EXMARaLDA* menu. The first row of the exported data is assumed to contain the category names for each annotation layers, and all other rows are treated as data proper. If the *Comments* field of the Excel workbook contains an EXMARaLDA header (usually generated by the importer), this may be used for the exported file, otherwise a generic header is generated automatically. Similarly, tier attributes stored by the importer in comments to the header cell of each layer may be exported, with the exception of user defined tier metadata in **<ud-tier-information>** and **<ud-information>** tags within tiers. If no tier attributes are found, default values are inserted, which assume that the first column is a transcription layer (type="t") and all other layers are annotation layers (type="a"), and that the display name of the layer is the same as the category name.

You can also choose to insert a space in to empty cells in the first column, which can be useful if you're using that column for tokens and exporting the data to a program that does not tolerate empty tokens. Finally, the Batch Export works similarly to the Batch Import – you can select multiple files and export them all using the above settings.

4.3 Transpose Selection in a New Sheet

Since an Excel 2003 spreadsheet can only contain a maximum of 256 columns but as many as 65536 rows, imported data is presented vertically by default. It is however possible to transpose a selection into a horizontal format similar to the EXMARaLDA editor using this command. The first row is assumed to contain layer headers and is rendered in bold face. If the selected data contains more than 256 rows, the rows are broken up into groups of up to 256 rows each and the headers are repeated for each block to make navigation easier. Note that some spans may cross the border of a block of 256 lines. In such cases the add-in will attempt to find a cutoff point starting at a 200 row block. Data that cannot be split up in this way (or in any way, if there is a span of more than 256 rows), will not be able to be transposed.

4.4 Export to PAULA

Export to PAULA works similarly to EXMARaLDA export, but with several additional options. It is assumed that the first selected column represents the tokens of the document and the first selected row contains annotation level names, which should be valid XML attribute names. It is possible to select automatic correction of annotation level names, which at the moment replaces spaces with underscores and umlauts and β ligatures with plain vowel followed by *e* and *ss* respectively. The token column is also used to generate the base text of the document, which is generated with spaces to separate the contents in each row. This column may not contain spans. Alternatively, it is possible to specify an already existing token file in the text field at the bottom of the form. If this is done, all references to a token file are replaced with this string, and no raw text and token files are created.

Export to PAULA

Use one Seg file for all span columns

Interpret ns:header as namespace

Generate annoset files

Fix invalid column header names

Export these columns as token annotations:

pos;lemma

Generate DTDs

None

All PAULA DTDs

Only general and markable DTDs

Output Files

Corpus prefix:

exmaralda.mycorpus

Merge to tok file:

merged.mycorpus.tok.xml

0 rows exported

Cancel Export

Users can choose whether or not an annoset will be generated for the data (a PAULA XML manifest of the annotation levels present in the document), what DTDs should be generated (all PAULA DTD's, only those relevant to span annotations, or none) and whether or not all span annotations are equally granular, in which case only one markable (Seg) file can be generated to define all spans, with each annotation level creating only a single feature file referring to the joint Seg file. This last option relies on the identical granularity of the annotation levels and will fail if this assumption is violated.

A further text field lets users choose annotation names (separated by semi-colon) which will be imported as feature attributes of the tokens, without a markable Seg file. To use this option, the corresponding level should contain no spans, or else unexpected results may follow.

Finally, users may specify a corpus prefix, which will be used at the beginning of all standoff file names in the PAULA document. It is also possible to use namespaces in column headers with an intervening colon (namespace:annoname), in which case the namespace is prefixed to the respective file names with a period as a separator, to form a PAULA namespace.

5 Limitations and Assumptions about the Data

5.1 EXMARaLDA XML

For the sake of simplicity and speed in handling large files, the add-in does not actually parse EXMARaLDA XML on import, but rather reads each file line by line, making certain assumptions about the order of elements and attributes. The add-in supports the following variations:

- Time line events' start and end attributes may stand in any order
- Time line elements may have labels and may be numbered arbitrarily
- For each tier, a category attribute is expected, which may appear anywhere

In all other cases it should be assumed that the importer expects data that looks like the data produced by the exporter. That said, the importer is somewhat robust in that it ignores anything it does not understand: this means it will happily import invalid XML if it also contains the elements it expects. If you run into problems caused by hidden assumptions made by the importer, please file a bug report, ideally with a sample of the data which caused the problem.

5.2 PAULA XML

As noted above, the following assumptions are made about the Excel data to be exported:

- The tokens are in the leftmost selected column, which contains no spans.
- Column headers contain only alphanumeric ascii strings beginning with a letter and without spaces or special characters (Umlaut, accents etc.; but see automatic header correction in 4.4 above)
- When merging to one markable (Seg) file, conflicting spans will lead to errors (the exporter does not validate for such errors at the moment).

For more information on PAULA XML see:

<http://www.sfb632.uni-potsdam.de/~d1/paula/doc/>

6 Contact

For questions, bug reports or feature requests regarding the add-in please contact Amir Zeldes (amir {dot} zeldes {at} rz {dot} hu-berlin {dot} de).