Müllerstrasse 64, 8004 Zürich, 043 311 59 15

Enter AG

MsgText

Manual

Version: 2.01

Longterm Preservation of E-Mail

A Program for converting *.msg*-files to *.txt*-files and extracting the attachments

Autor: Dr. sc. math. Hartwig Thomas, Enter AG

MsgText

Manual

TABLE OF CONTENTS

1 PURPOSE

When attempting to convert records of a government administration into a format suitable for long-term preservation, the problem of converting *.msg*-files into a non-proprietary format was encountered.

This led to the development of the program *MsgText*, which addresses this problem.

1.1 Why *.msg*-Files?

Many mail messages in large institutions today are generated and stored using Microsoft Outlook. This mail client uses various storage media for messages – e.g. local *.ps*-files, accounts on an Exchange Server, etc.

When storing mail externally in the Windows file system using Microsoft Outlook, the standard format offered is a *.msg*-file. This has the advantage, that the complete message (message header, message body, sender, recipients, HTML-body, attachments) ist stored in one file, whereas storing mail as a *.txt*-file only stores the message body as text.

It is fairly easy, to write programs, that will extract messages from Microsoft Outlook folders and store them externally as *.msg*-files in the file system. Such scripts need to address the various security restrictions that surround e-mail. One such program was developed by Enter AG and is called *MsgExport*.

In addition some records management systems – typically used by government administration – store mail in the form of *.msg*-files. When converting the content of such a records management system to formats suitable for long-term preservation, these *.msg*-files must be converted into a more reasonable format.

It is desirable, to separate the extraction of *.msg*-files from Microsoft Outlook storage from the conversion of *.msg*-files to formats more suitable for long-term preservation. The latter activity is not hampered by security problems. The program *MsgText* was created for this purpose. It can be executed on a Windows platform, even if no Outlook is installed.

1.2 Mail Formats Suitable For Long-Term Preservation

Microsoft's proprietary *.msg*-file format is not suitable for long-term preservation because – until recently – no open specification existed[1]. These files conform to the so-called OLE (Microsoft's Object Linking and Embedding) „docfile“ container format filled with MAPI (Microsoft's Messaging Application Programmer's Interface) content.

We have chosen the simple text format (as documented in in the open standards RFC 822, RFC 2822) which consists of a message header, an empty line and a message body, as a suitable text format for the long-term preservation of mail content.

Thus any *.msg*-file is converted by *MsgText* to a *.txt*-file which conforms to this standard.

Many mail users, however, send more structured mail, making use of an HTML (Hypertext Markup Language) mail format. Some mail users even send mail in an RTF (Rich Text Format) mail. The former is weakly standardized internationally, the latter is a proprietary but open Microsoft standard.

Where the mail content is available in a more structured form than just plain text, an HTML or RTF file is produced by *MsgText* in addition to the plain *.txt*-file. These structured versions of the mail content are stored alongside the output text file in an accompanying directory which is also used for storing attachments.

If attachments are part of the message, they are stored by *MsgText* in the attachment directory with slightly normalized file names in their original formats. If HTML content has links to attached files those are replaced by relative links, which make the HTML content appear complete, when opened in a browser.

Thus the *.msg*-file is converted

- into a *.txt*-file which is certainly suitable for long-term preservation,
- possibly into HTML- or RTF-files which may well be suitable for long-term preservation,
- into unchanged attachments most likely not suitable for long-term preservation.

The decision, how to treat attachments, (which may themselves be *.msg* files or *.zip* files containing *.msg* files, etc.) is left for further conversion steps. These must take into account, however, that (HTML-)links may be broken by further conversion of attached files.

1.3 User Interface

In the context of mass conversion of large amounts of records to be archived, it has become painfully clear, that only command-line programs without any user interaction whatsoever are useful. Attempts to make use of Microsoft Outlook itself to „print“ PDF-Versions of *.msg*-files, failed miserably in this respect. (In addition, they put the current user prominently into the title of the printed message, thus misleading the reader into believing, that the user who did the conversion was the sender or recipient of the message.)

MsgText is designed as a command-line program which will write its output to *stdout* and its errors to *stderr*. Also it returns an exit code, indicating whether the conversion was successful. Thus it is well suited to be called by batch scripts, which want to apply it to large numbers of *.msg*-files.

---

[1]The sepcifications that were made available by Microsoft are included in the developer‘s package of MsgText. The reader can judge by himself, that they leave a lot to be desired.

For users who only want to convert a single .msg file a trivial graphical user interface is available as *mt.js*. Double clicking it will display a dialog for parameter entry and then execute MsgText.exe.

2 PREREQUISITES

As *.msg*-files are in a Microsoft proprietary format, they are always created on the Windows platform. This proprietary format has been published by Microsoft after the first version of *MsgText* was published. The current second version does not require an installed version of *MS Outlook* any more for its operation. Nevertheless it makes use of the published interfaces of parts of the Windows platform (*ole32.dll, mapi32.dll*). Consequently *MsgText* can only be run under Windows (Win32).

Whereas the first version of *MsgText* was implemented in the programming language C#, for the second version C++ was chosen, because the „.NET-wrappers“ for the important OLE interfaces turned out to be insufficient. For „old“ C++ the Platform SDK (Software Development Kit) of Microsoft contains all the relevant interface information.

If a future version of *MsgText* is to run completely independent of the Windows operating system, it will probably have to be implemented in JAVA. The navigation in the „docfile“ format will then have to be implemented then without dependence of the Windows system DLLs (dynamic link libraries) *ole32.dll* and *mapi32.dll*.

For editing and compiling the sources of *MsgText* it is recommended, but not required, that Microsoft's Visual Studio be used.

3 INSTALLATION

MsgText is „installed“, by copying the distribution files to a suitable location (e.g. „*C:\Programs\Enter AG\MsgText*“ or „*My Documents\Enter AG\MsgText*“, if the *Programs* folder is not accessible to the user). No further „installation“ is necessary. The user installing *MsgText* does not need any special access rights.

4 USAGE

In a command window enter on the command line

```
MsgText /h | [/d] [/l] <msgfile> [<textfile> [<attachmentdir>]]
```

where

/h	displays usage information
/d	overwrite <textfile> and delete previous contents of <attachmentdir>
/l	list files that would be created (.msg file will not be converted)
<msgfile>	file name of input .msg file
<textfile>	file name of output (default: <msgfile> with '.msg' replaced by '.txt')
<attachmentdir>	attachment directory will be created if needed (default: <msgfile> without extension)

If the executable *MsgText.exe* is not in the Windows system directory, in the current directory or in a directory in the %PATH% environment variable, its absolute file name must be entered instead.

The file names of the attachments are used. For embedded messages the Subject line is used as a file name but invalid characters like : '*', '?', '\', '/', '"', '<', '>', and '|' are removed.

5 FUNCTION

If the message file *example.msg* contains HTML mail and two attachments *example.att* and *prüfung.att*, then running

```
MsgText /d examples.msg
```

will produce the following directories and files in the directory where *example.msg* is located:

- *example.txt*
- *example\example.html*
- *example\example.att*
- *example\prüfung.att*

If *example.html* contained a link to *prüfung.att*, it will be replaced by a local link to *prüfung.att*. Thus opening *example.html* with a browser will display the complete HTML message with its linked components (e.g. images).

If *example.txt* existed before, it is overwritten without warning. If the /d switch were not present, the conversion would be aborted in such a case. If a subdirectory *example* existed before, it is deleted with all its contents without warning. The subdirectory *example* is then created and filled with the HTML message and the attachments.

The original message file *example.msg* will remain in its place unchanged. It has only been opened for reading.

If *example.msg* represented a received message, then the message header in *example.txt* is an exact copy of the Internet header of the actual transmission. If it represented a sent or unsent message, then the message header may only contain partial information. In particular the *Date:* header is missing in messages that were not sent.

6 DISTRIBUTION

MsgText being an open source program is distributed with the sources available in the version control of SourceForge[1]. For users who just want to use *MsgText* a ZIP file *mtbin_<v>_<rr>.zip* just containing the executables is available.

7 DEVELOPMENT

For developers we have packaged the relevant documentation from Microsoft as well as the useful *DFVIEW.EXE* from Microsoft in the *documents* folder.

Zurich, October 8, 2009 Dr. sc. math. Hartwig Thomas

[1] https://sourceforge.net/projects/msgtext/develop