Etasoft – eTranslator 2.x

EDI eTranslator is designed for automated and manual translation of EDI X12 files into XML and back. It supports all EDI X12 40xx, 50xx, 60xx, 70xx message types.

Setup

For Windows install the software using provided setup file e_setup.exe.
For Linux install the software unzipping the file etranlinux.zip.

Windows installation contains both GUI based and command line based translation tools. Linux package only contains command line translation tool. If you want to try Linux package then copy and use license from Windows GUI License screen.

Scaffolding

First translation has to be **EDI to XML translation**. Idea behind this tool is simple: once you translate your typical EDI file into XML, resulting file will contain layout and tags (scaffolding) to reproduce EDI file.
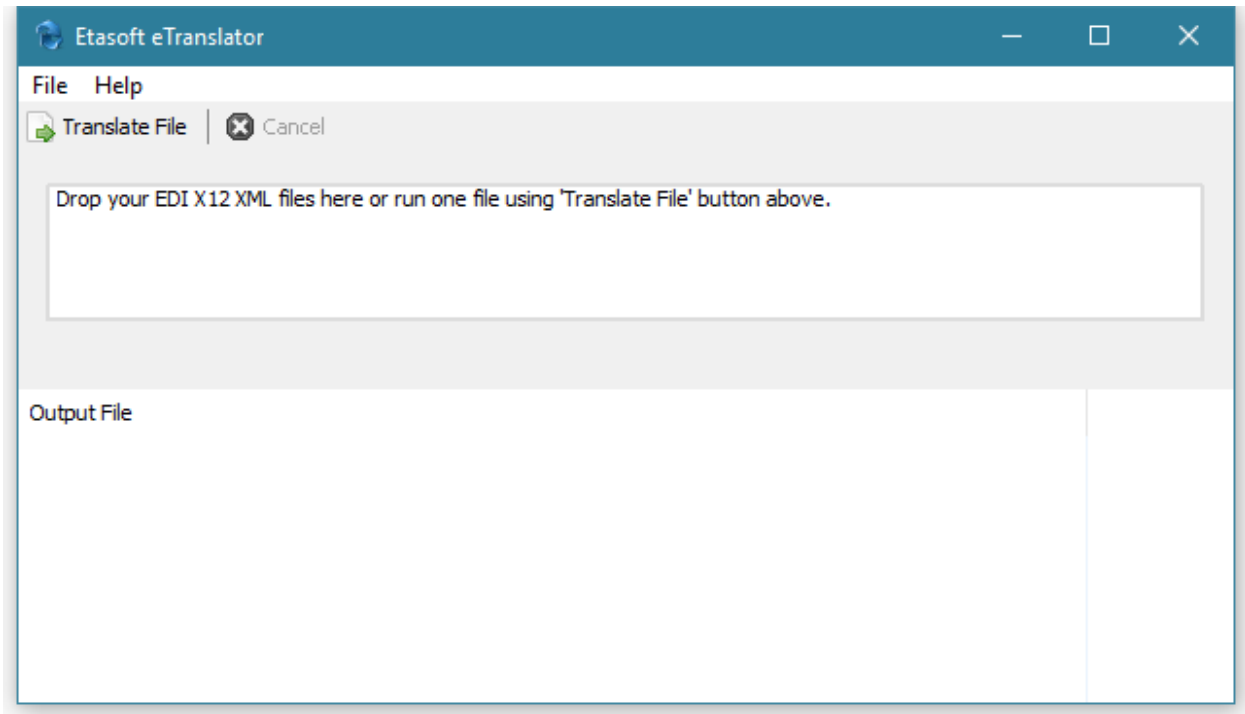
It is XML scaffolding you can reuse and built on. You can load it into databases or modify data and then translate it back into EDI files. XML scaffolding stays the same. You will simply reuse it.

This tool will not work if you must have specific XML tags nested in different structure and loops. eTranslator output is based on standard EDI layout. Resulting XML matches EDI loops exactly that is why it can be translated into EDI.

If you need to control XML tag names and structure precisely then our XTranslator product might be better choice.

Processing

GUI based translation tool is designed for manual processing. It can process single file or multiple files.

Translator produces XML files ready for XML parsers. You can also make changes to XML file and use it to translate back into EDI.

```xml
<?xml version="1.0" encoding="UTF-8"?>
<xml_edi>
    <metadata>
        <release>5010</release>
        <tsnum>837</tsnum>
        <segmentsep>~</segmentsep>
        <elementsep>*</elementsep>
        <subelemsep>:</subelemsep>
        <crlf></crlf>
        <style>LoopWithTags</style>
    </metadata>
    <ISA_InterchangeControlHeader>
        <eI01_AuthorizationInformationQualifier>00</eI01_AuthorizationInformationQualifier>
        <eI02_AuthorizationInformation>          </eI02_AuthorizationInformation>
        <eI03_SecurityInformationQualifier>00</eI03_SecurityInformationQualifier>
        <eI04_SecurityInformation>          </eI04_SecurityInformation>
        <eI05_InterchangeIDQualifier>ZZ</eI05_InterchangeIDQualifier>
        <eI06_InterchangeSenderID>870426777       </eI06_InterchangeSenderID>
        <eI05_InterchangeIDQualifier>ZZ</eI05_InterchangeIDQualifier>
        <eI07_InterchangeReceiverID>141822880      </eI07_InterchangeReceiverID>
        <eI08_InterchangeDate>160201</eI08_InterchangeDate>
        <eI09_InterchangeTime>2047</eI09_InterchangeTime>
        <eI65_RepetitionSeparator>^</eI65_RepetitionSeparator>
        <eI11_InterchangeControlVersionNumber>00501</eI11_InterchangeControlVersionNumber>
        <eI12_InterchangeControlNumber>000189323</eI12_InterchangeControlNumber>
        <eI13_AcknowledgmentRequested>1</eI13_AcknowledgmentRequested>
        <eI14_InterchangeUsageIndicator>P</eI14_InterchangeUsageIndicator>
        <eI15_ComponentElementSeparator>:</eI15_ComponentElementSeparator>
        <GS_FunctionalGroupHeader>
            <e479_FunctionalIdentifierCode>HC</e479_FunctionalIdentifierCode>
            <e142_ApplicationSendersCode>870426777</e142_ApplicationSendersCode>
```

Translator adds header with <metadata> tags used for translation back into EDI. XML data is also escaped to produce valid XML. In the screenshot above ">" is replaced with ">" .

XML output looping matches EDI X12 loops.

Tags for EDI elements contain tag names prefixed with lowercase "e". Full tag name for element starts with "e", then element number from EDI Standards and full name of the element. Example: <e142_ApplicationSendersCode>.
Tags for EDI segments start with X12 segment name from Standards then full name of the segment. Example: <GS_FunctionalGroupHeader>.

Command Line Execution

While GUI interface is more convenient command line interface lets you automate translations via scheduling tools on Windows and Linux. You can also run command line translation as part of your program (sometimes this is called 'shell out' execution). eTranslator does not use any heavy runtimes such as JVM or .NET or any other runtimes that need JIT 'warm-up' for execution and might have slow

startup.

eTranslator 'shell out' is an option because load time for translator is small. It is designed to be fast.

Use utility called etrans to run translations via command line. Command line tool has more flexible options. Run command line tool with "—help" parameter to find all available options.

Usually companies run automated jobs via command line. That means using scheduling tools or services to run jobs. Use Scheduled Tasks on Windows and cron on Linux if you do not have any other tools to automate processing.
Advantage of this approach is that you do not need to start GUI to run translations. Scheduled Tasks or cron will continue running it in the background even if GUI is closed (or never opened).

```
C:\x\godev\src\edi\prod\etrans
λ etrans --help
Usage of etrans:
  -i string
        input file name or wildcard pattern
  -license string
        license
  -o string
        output directory

C:\x\godev\src\edi\prod\etrans
λ etrans -i=204.txt -license=EVGVtcG9yYXJ5fDIwMTgwODIzfDF8RVRSQU58MzkwODA0MDYyNQ==
Output file: "204-20180806_013025.xml"

C:\x\godev\src\edi\prod\etrans
λ |
```

Few important parameters:
Use "-i" parameter to specify input file. You can also use wildcards to process multiple files. For example:

etrans -i=C:\prod\837*.txt -license=EVGVtcG9yYXJ5fDIwMTYxMjA5fDF8RF8RURJVkk=

This will process all files with *.txt extension in directory C:\prod\837.

Use "-o" parameter to specify output directory. By default output file name is formed using input file name, date, time and specific extension (XML or EDI).
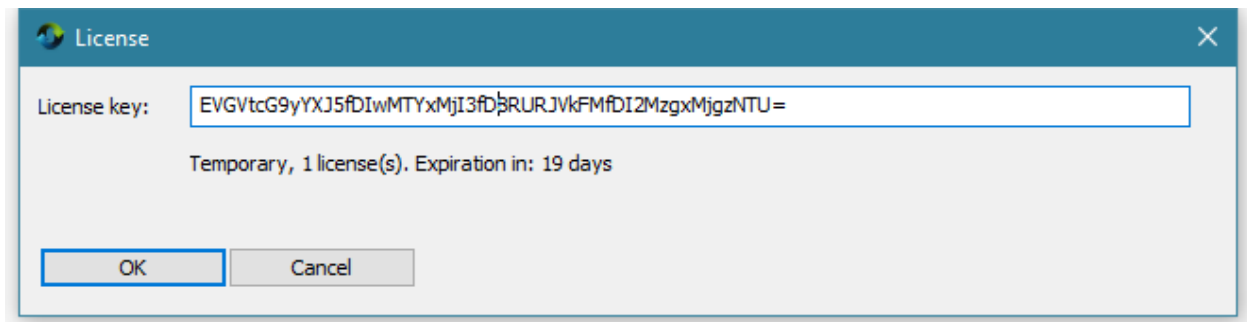
When processing files on Linux use "/" directory separator instead of "\".

If any of the files fail translation command line tool returns non-zero exit code.
Zero exit code means no-error.

License

eTranslator comes with trial and retail license. You can try and evaluate it free for 3 weeks. Once trail version expires please purchase license for continues use.

To start a trial version and receive license key for 3 weeks simply run GUI tool and open Help-License screen.

Use license key from GUI Help-License screen to run command line tool, and pass license into it via –license parameter.