Etasoft
Building blocks of integration

Best practices for EDI setups

Main website http://www.etasoft.com
Extreme Product Line website http://www.xtranslator.com

Purpose

The purpose of this document is to explain basics of EDI setups. It also tries to provide some of the best practice tips on how to setup EDI systems. Document uses a lot of common EDI and computer terms. Some of the terms are also specific to Etasoft Inc. products.

EDI systems

EDI system is combination of computer software and hardware setup to process electronic documents in standard formats such as EDI X12, EDIFACT, HL7 and many others. Most complex and hardest to setup part of the system is usually software, so that is what we will focus on in this document.

Usually EDI systems contain these software packages:

1. EDI translator package (example: Etasoft Extreme Translator)

2. EDI validation package (example: Etasoft Extreme Validator)

3. EDI automation and communications (example: Etasoft Extreme Processing)

Setup depends on what functionality is expected from the system. If no validation is required EDI validation packages are omitted. Some small businesses and even mid size companies do not automate they EDI systems, so they do not use EDI automation tools. So we are left with EDI translator as core package that is present in almost all EDI systems.

Other requirements come into play when setting up EDI systems:

1. Performance – how many megabytes need to be processed per hour or per day?
   Most EDI transactions are small in size (about 60Kbt), however many trading partners batch process and combine them into bigger files of 1Mbt to 40Mbt. You may exchange data with trading partners that combine they quarterly files into one big file. Those files might get above 250Mbt in size. Big files present extra challenges.
   Many translators have no problem processing many small files, but big files take extra time and might need to be scheduled for overnight processing. Sometimes you might need to have extra processing step to break them down into smaller files with just one transaction or interchange in them.
2. Changes – do you expect many changes to translation process in the near future? If yes, then flexibility and extensibility of the tools you are going to use becomes very important.
3. Automation – do you want to have process automated or not? Automation takes time and effort to setup but usually pays back in the long term.
4. Validation – do you want incoming or outgoing transactions validated? Processing data without validation means putting your fate in the hands of your trading partners. You may get away without full featured validation product, but some basic sanity checks can go a long way.
5. Error reporting – do you want errors to be logger and reported?
6. Error recovery – do you want system to recover in the face of errors and what steps to take when errors happen?
7. Longevity - how long EDI system will be used in service? Some EDI systems are used for brief transition moments from one legacy application to another more modern application. But most EDI systems run for many years. Vendor support and chosen technologies become more important for long running systems.

Introduction to EDI Translators

EDI translation packages are the engines of EDI systems. Sometimes they even incorporate EDI validation features and provide full EDI validation as well as translation. Most translators can execute in automated fashion, semi-automated manual and test-debug modes.

1. Automated mode gives you ability to run translations in server environment where translations usually run at certain times (example: 2AM at night) or at time intervals such as every hour, or when some event occurs (example: file is placed in some directory that is being watched by the software). Automated mode provides best cost vs. investment ratio since it usually requires least amount of maintenance and does not require high skills to manage and execute.
   However error reporting and data validation is critical in these types of setups, since there is no human operator that checks translation results, every time translation runs errors have to be captured and forwarded, maybe send as email or SMS cell

phone messages. Without proper validation and reporting system might be running for days processing corrupted data, providing only false sense of security and comfort that everything is fine.

2. Semi-automated or manual mode is when human operator runs the translations and checks results. While these systems provide least amount of effort to setup, they usually have high running cost because operators have to be employed and possibility of the human missing some translation error is high. Operator might get sick and take days off halting system execution for weeks. Employees also get fired or leave companies and unsuspecting colleagues inherit systems they do not know about. Cost of retraining is usually high and errors tend to pop up during transition period.
   Usually only low volume or small businesses use this approach to EDI translation.

3. Test-debug mode is used to setup and test the translations in initial development phase of the EDI project. It is used by EDI specialist or software engineer. Once translations are developed and tested, this mode is not used anymore until changes need to be made to the running systems sometime later.

EDI translator can be divided into two main categories: stand-alone packages and build-in packages.

1. Stand-alone translators are fully functional self contained translators. There are many EDI translator vendors. Each package has its own strengths and weaknesses (see next chapter on EDI Translator Strengths and Weaknesses).

2. Built-in EDI translators come already as part of other specific software package (example: HealthCare suite, Invoicing software, etc.). Those translators allow rather little customization and produce or consume only specific subset of EDI messages. However they are very easy to setup and in many cases do not need a lot of maintenance. Updates to translator come as part of main product updates. If you are developing software that needs translator capabilities Etasoft provides number of data translation products you can integrate into your applications using Developer SDK.

EDI Translator Strengths and Weaknesses

Here we cover stand-alone translators and they types.
Most translators fall into two basic categories: do-it-yourself translators and black-box translators with mapping tools that require little programming skills.

1. Do-it-yourself translators are designed for programmers and software engineers a like to help easily create they own full featured translators. Some of them generate code for you and let you fill in the blanks. Some simply provide interface in a form of low level developer API to create or parse EDI messages.
   The obvious benefit of this approach is that software engineer has full control and can tweak translations anyway he/she wants. Negative sides are lots of hours of repetitive coding, resulting “spaghetti” source code that is hard to maintain or change. Any future changes to the translation in most cases require full rebuild, retesting and redeployment of the software. Software engineer will be employed full or part time as long as you have EDI translator developed just for your business.
   Those types of EDI translators are essentially “lets reinvent the wheel” software packages designed strictly to your requirements. Many EDI systems run for ten or twenty years. Chances are that original developers will not be around for that long and new software engineers may not have skills to run old inflexible systems based on code written 20 years ago. New programmers will not know the programming languages and libraries that have been used to build EDI translator in-house.
   Positive side is that some of the translations and validations simply can not be developed by fully functional mapping tools. In those cases do-it-yourself translators are the only choice. Translations and validations are limitless since almost any data transformation is possible. In most cases do-it-yourself translators do not use translation maps, and if mapping tools are used they essentially are “spaghetti” source code generators used to generate initial skeleton for the translation. Once programmer modifies generated code, there is usually no way to use generator anymore.

2. Second category is translators with mapping tools that do not generate code and provide translation execution via pre-built translator engine. This is almost complete opposite for do-it-yourself translators. In this case translator is treated as “black box”. It takes inputs and produces outputs. Most translators in this category have ability to insert formatting or validation functions and scripts into the translation maps, allowing required customizations for your specific needs. They also come with pre-built templates for one or both sides of the translation mapping from input to output. Some also provide developer SDKs allowing your software engineers to plug-in they own functions into running translations or execute translations from your programs.
   Before deciding to purchase full featured translator look for these main features:
   a) Extensibility – ability to write scripts and/or functions and plug them into the translator “black-box”.

b) Flexibility – is it possible to create many various types of translations? The more - the better. You might find translator initially used only by one department in the organization, two years down the road being used by two or three to do other types of translations.
c) Templates – does translator come with pre-built templates?
d) Developer SDK – ability to call translator from your programs and integrate with your in-house systems.
e) Execution modes - ability to run in various execution modes (automated, manual or via developer SDK).
f) Bug fixes and updates – does translator vendor provide regular updates and bug fixes.
g) Technologies – was translator developed with new technologies and programming languages. Old translators will eventually drop off the picture and you will no longer have support for them. This is true even with established translators, since new versions may not port old mappings and you have no other way to upgrade to new version but to redesign translation maps using the new version.

Negative side is that “black-box” translators can not be as flexible as do-it-yourself translators. Some most complex translations are just way too difficult or nearly impossible to create by just mapping tools.
Positive side is that many translations can be created and modified by easy to learn mapping tools. They take less time to setup and maintain. In most cases your business does not need to employ software engineers forever. You may receive support, bug fixes and updates to the translator as long as original software vendor provides support. If translator comes with developer SDK your software engineers may integrate translator into other systems in your enterprise. Many EDI translators come as part of bigger group of related software packages including validation, automation and communication packages. Those combined can cost you thousands of programming hours to create if you decide to develop them in-house.

EDI Translator Ownership vs. EDI Clearinghouses

EDI Clearinghouses are companies that setup EDI for you and process your transactions based on per use fees. Most common fee model is pay-per-transaction or pay-per-amount of data model. If you have really low volume of data this model is attractive. However keep in mind that EDI Clearinghouses also charge setup fees since they create translation mappings for you. Also any future modifications to the translations have mapping costs as well.

When purchasing or paying for EDI Translator look at the total cost of ownership. You may find a translator package for 4,000 USD to be expensive and decide to go with EDI Clearinghouse but two years down the road your per-use fees might be way too high if you have grow in number of transactions or number of your trading partners increases. EDI Clearinghouse owns your translation mappings and most of the time you can not transfer ownership or move to other EDI Clearinghouse.

If EDI processing is mission critical for your business that is another important factor to own translation package. You may trust somebody to handle your data needs if that data does not carry sensitive business information, but in most cases EDI data is full of business sensitive information. Once you hand over your data to third party for processing it is out of your and your trading partner's control. With today's off-shoring trends your data might be traveling half way around the world and somebody outside of prosecution, and laws of your country might be simply using your data for they own benefit without any worries to be caught.

While we advocate ownership of the translator package, there are situations when EDI Clearinghouse is better option: you have low translation volume and do not expect your translations to grow, you do not have any skilled employees to setup EDI translation package and you have no infrastructure to run translations in-house.

EDI Validation packages

There are costs and dangers associated with not using EDI validation packages. If you opt-out to use only EDI translator since you trust your trading partner, you might end up processing corrupted or simply junk data without even knowing it. In best case scenario data might need to be resent and reprocessed. In worst case scenario there might be some financial losses involved.

This is especially true if you have more than one trading partner. The variation of errors and problems grows exponentially with the growth of variations of messages you might receive. In other words, more trading partners with more messages types – greater chance for the errors to occur. For those setups best practice is to treat any incoming EDI transactions as being full of errors. No trust policy.

EDI Automation and Communication

Many additional challenges rise in automated systems. Automated systems run in unattended mode, sometimes during night hours, and usually process much greater volumes of data than non-automated systems. They have much greater requirements in terms of performance, error reporting and reliability.

There is great variety of automation tools and methods. Some basic automation would be by using simple schedule tool that kicks off EDI translation at certain specific times during the day. It does not require great financial investment. If EDI translator can run as command line batch job, it can be automated using Scheduled Tasks or any other scheduler. Some translators able to scan directories for file names matching certain pattern, and ones that match get processed (this feature is supported by Extreme Translator).

Once translation is done, you might want the job to move incoming file into an archive folder or device. This will ensure your systems ability to reprocess data. Let's say that translator feeds some other system called Loader. Loader performs some checks and loads data into company's database. If Loader has a bug and fails, it has no records processed for a few days. Incoming data has to be reprocessed in order for the Loader to work thru the backlog.

EDI systems are being used for ten or twenty years, extra redundancy and ability to recover from errors goes a long way. Think of the system as a castle. If castle is under the siege and walls are breached, you want to have ability to retreat into the keep instead of falling. Have a backup copy of the incoming data so it could be reprocessed and have a backup plan if translation fails. When translation fails system should react based on the type of the error. If it is internal error (example: computer hard disk is full) you want to rerun the translation, and if it is external error (example: trading partner is sending invalid data) you want to contact trading partner.

Instead of using home grown basic automation tools it is better to use integrated products and systems. Integrated automation tools will run not just translation and validation but also execute other related communication and data transfer activities. Some automation tools also have ability to react to certain file and directory events, example: once file is copied to the directory, execution would start. They also capture errors and can send error notifications via email. Reporting is important for fully automated systems. No need to keep on running processing if your trading partner is sending you some junk or invalid data.

While there are many communication protocols at the time of this writing popular ones are AS2, SFTP and FTPS. AS2 was developed for EDI communications however it is much more expensive to acquire. Also SFTP and FTPS are much more wide spread since they are common protocols for all kinds of file transfers not just EDI.

EDI setups usually are based on “Mailbox” concept. Mailbox is nothing more than a directory on local drive, network server or FTP folder somewhere on the Internet. Mailbox is dedicated to handle one type of message. Example: mailbox for EDI X12 810 Invoices. Mailbox only handles incoming or outgoing files. So in our case let’s say it is “mailbox for incoming EDI X12 810 Invoices on local drive”.

Since most EDI translators work on files matching certain pattern, once you run translation, it will know what translation map to use against this specific directory (“mailbox”) and what pattern to look for. Every message type gets its own mailbox. So if you need to receive EDI X12 810 Invoices and send out EDI X12 997 Acknowledgements you would have one mailbox for 810s and one for 997s.

Now imagine you have all your files in one directory and you have no idea about this “mailbox” concept. Every time you look at the directory contents you see long list of files and you have no way of knowing where you have 810s and where are 997s. Only way to distinguish them would be to have specific file naming convention or manually open each and every one of those files using text editor. That’s not so bad right? But imagine that you have many different trading partners sending you many message types. You can not force everyone to adopt your file naming convention. Next thing you know your processing directory is clogged with all kinds of files. Now real nightmare begins – once some translations or validations fail you need to find what went wrong by going thru thousands of files in one big giant directory. Even if you have perfect error reporting facilities, things just tend to become hard to track if all files are in one big mesh.

While not that common some trading partners may send you files with the same names. Chances are slim but you would not want one trading partner overwriting files from another trading partner. Since all your files come into one directory it is possible.

If you setup mailboxes translator does not need to waist time going thru all the files, read they ISA/IEA envelopes one by one trying to find out what message type is in the file, who is sender and receiver and what translation map should handle it.

There are many other reasons why separating files by message type and direction (incoming vs. outgoing) is a good idea. Mailbox concept simply makes EDI setup much more cleaner. It is easy to setup and does not require any special training or software to manage.

Performance

Unless you are going to be processing just few small files a day system performance will be a factor. It is one of the most important factors for EDI systems. When designing and developing the system consider those factors to improve performance:

1. How complex the mapping is? More loops and translation functions – slower it will run.
2. How many other processes are running on the same machine at that same time?
3. Are there any other processes accessing same databases or disks at that time?
4. How much data is being translated or sent/received? Volume. More data - slower it gets.
5. If you are mapping to the database tables do you map to just one table or a lot of tables? More tables - slower it gets. Also keep in mind that some databases are faster than others. The way how you setup database tables is also important. Example: tables with many indexed fields will load with new records slower than ones that have no indices. You might want to disable or drop indices before the load and recreate them just after the load.
6. As a general rule – everything that needs to be accessed via network is slower than using local disk.
   If you translate data files that are on the network drive consider copying data to the local disk just before running the translation or validation. Chances are your network will fail more often than hard disk. So getting data from local disk improves performance and reliability of the system.
   Same applies to the SQL databases. If you have remote database, it will be slower to access than local one.

If you have to use network to access files and databases consider running most of EDI processing overnight. During the night hours load on networks in most organizations is lower than during the day.

7. Another general rule – everything that is in memory (RAM) is faster than local disk. Usually this is much harder to improve since most EDI tools do not allow you to change how they load data, use buffers or perform disk input/output functions.

Good Setup

There is an example of good setup when data is being received:

1. Use automation software to run the EDI system on dedicated server.
2. Receive data via secure communication.
3. Validate EDI format and structure first.
4. Once EDI validation passes, you will run data thru EDI translator.
5. Translation result has to be checked before it gets loaded into your backend systems or databases. Usually this step is done by in-house developed software that performs the load. Some basic sanity checks to make sure that incoming dates and numbers are valid will come before queries against existing data to make sure that duplicates are not being loaded. Loader may run other business rules to check incoming data.

When data is being sent EDI translator runs first, then EDI validation checks produced message and only then message goes out.
Real world setups usually do not have that many steps and sometimes are not automated in any way. Automation should be the key for organizations that receive number of messages per day since it brings obvious cost benefits in the long term.

Final Word

Every EDI system is different. Some have low requirements and run once a day or once a week. Some process tens of thousands transactions per day and run 24/7. Use common sense when making setup decisions or purchasing the software.

This document tried to address basic issues with EDI setups. If you have any questions please contact our technical support.