National Frozen Foods Case Study

Leading global frozen food company uses Altova MapForce® to bring their EDI implementation in-house, reducing costs and turn-around time, while increasing overall efficiency of their external business transactions and communications.

Overview

National Frozen Foods Corporation is a family owned, private label company specializing in the production and packaging of frozen vegetables. As a small food manufacturer, the company contracts with a variety of other vendors to get their products to market, including a cold storage company that stores their vegetables before shipment. Communications with the warehouse are sent via the X12 dialect of Electronic Data Interchange (EDI), a standard data format for business transactions.

The company has a long history of using EDI for its business-to-business communications but had been relying on outside consultants for the conversion of documents from their internal formats to EDI for processing. Outsourcing proved to be expensive, often untimely, and introduced bugs, which would require additional development costs.

Recognizing a need to cut costs, save time, and create a more efficient overall documentation workflow, National Frozen Foods decided to bring their EDI conversion in-house. The project was assigned to an IT manager, who had some generalized coding and technical experience, but had never used code generation or data mapping tools before.

National Frozen Foods now uses Altova MapForce internally to convert their warehouse orders to EDI and to automatically generate code for order processing.

The Challenge

National Frozen Foods required a solution that they could use internally to convert the flat file format (EIF) produced by their back-end ERP system into the X12 dialect of EDI for processing by their e-commerce software, TrustedLink for Windows. It needed to be intuitive and simple enough for their IT team to master quickly, and robust enough to provide an advanced conversion utility as well as seamless code generation.

Adage ERP is an enterprise resource planning tool that allows manufacturers and distributors to monitor their orders, as well as other internal processes, from inception to product delivery, all the way through to payment. National Frozen Foods uses this application internally for enterprise-wide accounting and integration purposes. ERP systems, however, are not built for business to business communication and often have no option to output standards-compliant EDI data for these transactions.

When warehouse orders are created, they are created in Adage ERP and output as flat files. These files then need to be translated into the X12 Warehouse Shipping Order Transaction Set (940), a standard transmission file format for warehouse shipping orders in the context of an EDI environment. These, in turn, then need to be imported into the TrustedLink EDI processor which sends the communication to the cold storage facility via AS2, a widely adopted secure Web transmission standard.

The Solution

After carefully reviewing the data mapping products available on the market, the IT team at National Frozen Foods settled on Altova MapForce to handle the required document transformation and code generation. MapForce is a visual data mapping and code generation tool that supports any combination of XML, database, flat file, EDI and/or Web services data formats. MapForce was chosen because of its built-in EDI support, flat file conversion and code generation capabilities, and its ease-of-use.

In order to successfully bring their EDI implementation in-house, and to continue forward without the use of consultants, National Frozen Foods needed to parse their source data, create a data map to the X12 file format, and then generate code that could be used for future transformations. The requirements were as follows:

- **Parse the flat file source data**

  The source file contained both cryptic headings that did not provide a straightforward description of the associated data and some ancillary data that did not need to be exposed to the mapping engine. The FlexText utility in MapForce allows users to create custom references for nodes, as well as ignore data that is irrelevant to the mapping.

- **Set up the mapping environment**

  In addition to the flat file source data, some additional data needed to be pulled from the ERP database on-the-fly, including comments and a full name and unique ID of the warehouse initiating the order. This information and the parsed flat file would be mapped to MapForce's built-in template for the X12 940 transaction set.

- **Convert data to the X12 format**

  Certain computational functions needed to be applied to translate many different datatypes and formats from the source data to those required in the X12 vocabulary.

- **Generate control numbers**

  The ISA segment of the X12 format requires that unique control numbers be autogenerated and assigned to interchange and the sender of the message.

- **Export, compile, and execute the code**

  MapForce would be used to generate programmatic C# code for Microsoft® Visual Studio®, which would then be integrated into TrustedLink for Windows for ensuing EDI transactions.

ALTOVA®

The code generated by MapForce would allow National Frozen Foods to initiate an automated data transformation and mapping workflow for all future warehouse shipping orders. If changes were needed, they would be able to go back to their visual representation of the mapping to make edits, rather than being required to comb through the complex code.

Parsing the flat file source data

National Frozen Foods used the MapForce FlexText utility to convert their flat file source data into a parsable structure for transformation and to isolate the relevant data that would be used in the mapping. The intuitive graphical user interface allowed the IT manager to get started right away preparing the EIF file for transformation into the 940 format. An example of the data that might be contained in a typical source file is below.

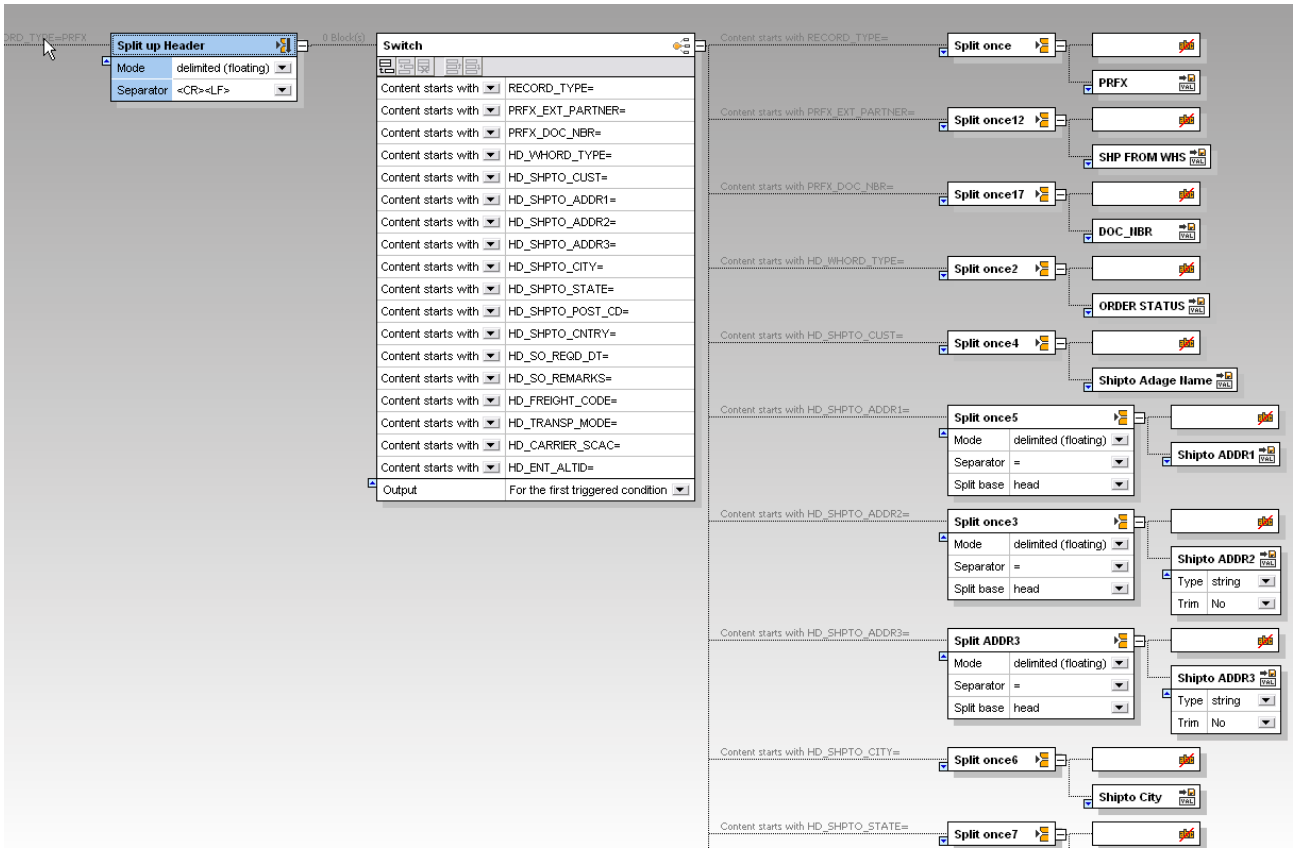
```
RECORD_TYPE=PRFX
PRFX_CO_ID=NA
PRFX_EXT_PARTNER=AMERICOLD
PRFX_DOC_TYPE=Whs Shipping Order
PRFX_DOC_NBR=1
PRFX_TRANS_DIR=O
PRFX_DATE=2007-11-09
PRFX_TIME=03:29:15
PRFX_CONTROL_NBR=20
RECORD_TYPE=SO_ID
HD_ENT_ALTID=007
HD_ENT_ALTIDTYP=DUNS
HD_EXT_XREF=
HD_WHORD_BRNCH_ID=NATL
HD_WHORD_TYPE=CHG
HD_SHPTO_CUST=A
HD_SHPTO_ADDR1=addr1
HD_SHPTO_ADDR2=addr2
HD_SHPTO_ADDR3=
HD_SHPTO_CITY=LAS VEGAS
HD_SHPTO_STATE=NV
HD_SHPTO_POST_CD=89115
HD_SHPTO_CNTRY=
HD_SHPTO_FRGT_ZNE=Northeast
HD_SHPTO_GEO_CODE=LasVega,NV
HD_CUST_ID=A
HD_CUST_DUNS=
HD_BLLTO_CUST_OUT=A
HD_BLLTO_DUNS=
HD_BLLTO_ADDR1=billaddr1
HD_BLLTO_ADDR2=billaddr2
HD_BLLTO_ADDR3=
HD_BLLTO_CITY=HOUSTON
```

Using the FlexText utility, the relevant data was culled for inclusion in the data transformation. A simple split operation separated the header and detail sections, while a switch was invoked to select and ignore any root descriptors in the header, using the = sign as the delimiter.

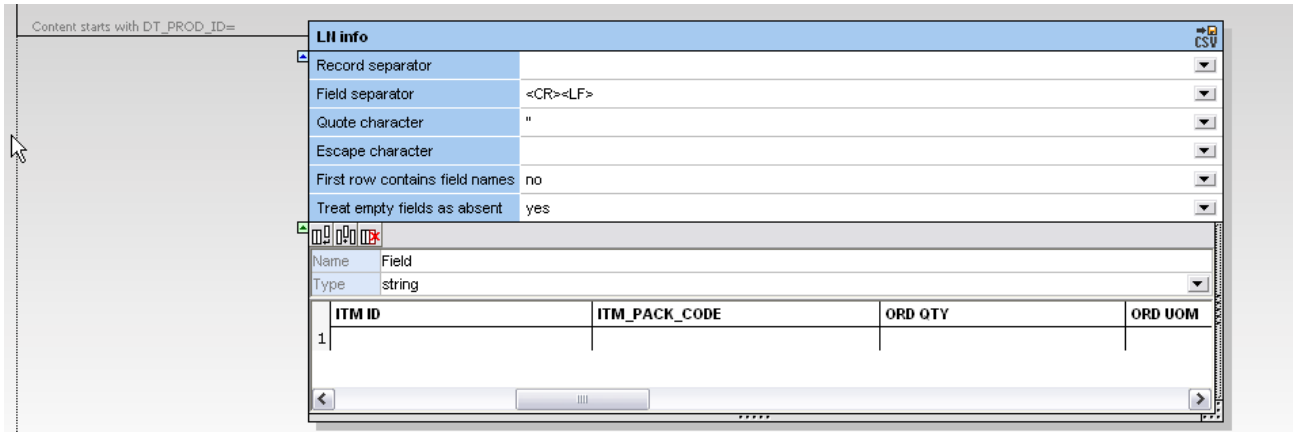
NATIONAL FROZEN FOODS CORPORATION

ALTOVA®

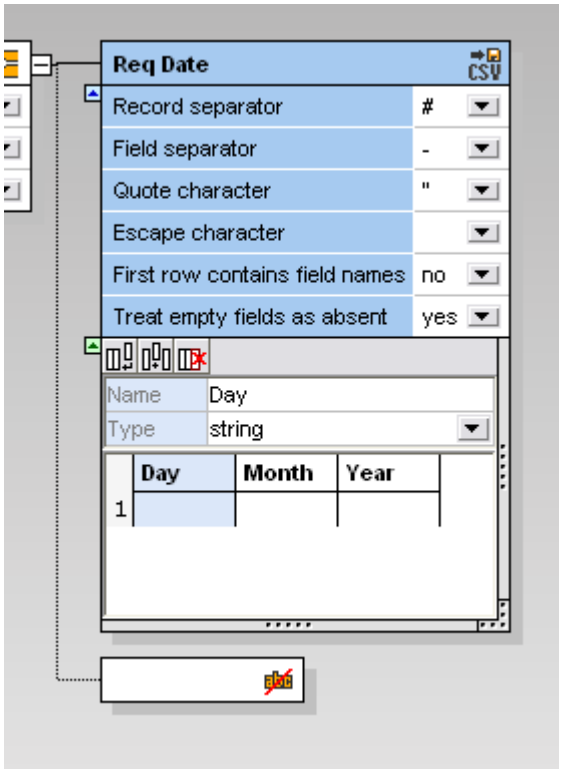
A detail of this portion of the FlexText configuration is shown below:



In order to correctly populate the EDI segments, which would rely heavily on looping operations, the detail data was saved as CSV:



Finally, the FlexText utility was used to extract the data from a date/time record so that the mapping component would be able to consider it as three separate entities. The IT manager used the split operator to exclude the time data and then again to sort the date and remove the hyphens from the original format, saving this data again in a CSV format that would later be referenced by the visual mapping component of MapForce.



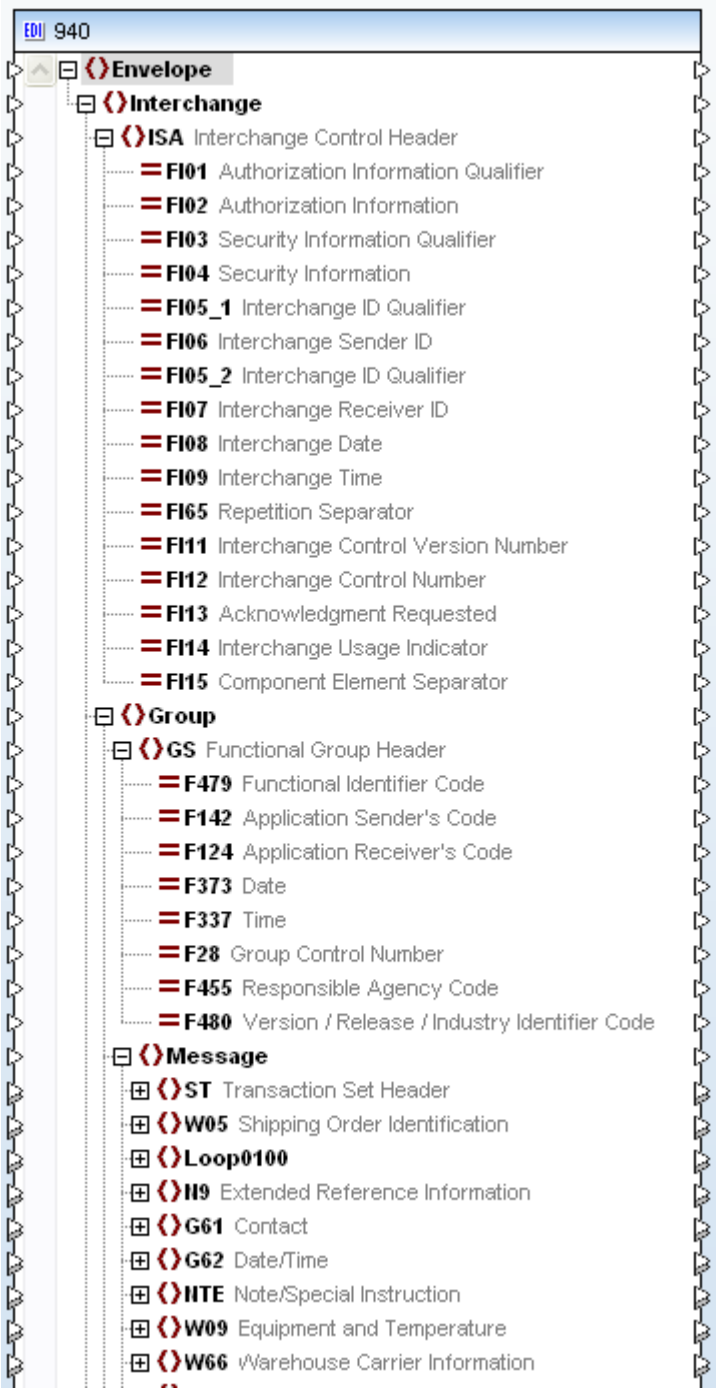
With the FlexText utility, the IT manager was able to create a template from the source data, transforming the flat file into a graphical representation. The ERP flat file data was now ready to be transformed into an EDI structure using the visual data transformation component of Altova MapForce.

Setting up the mapping environment

MapForce offers extensive support for EDI and provides templates for transaction sets in both the UN/EDIFACT and ANSI X12 dialects. This allowed the IT manager to simply select the X12 940 format from a list to incorporate it into the mapping project as the target data structure.

NATIONAL
FROZEN FOODS CORPORATION

ALTOVA®

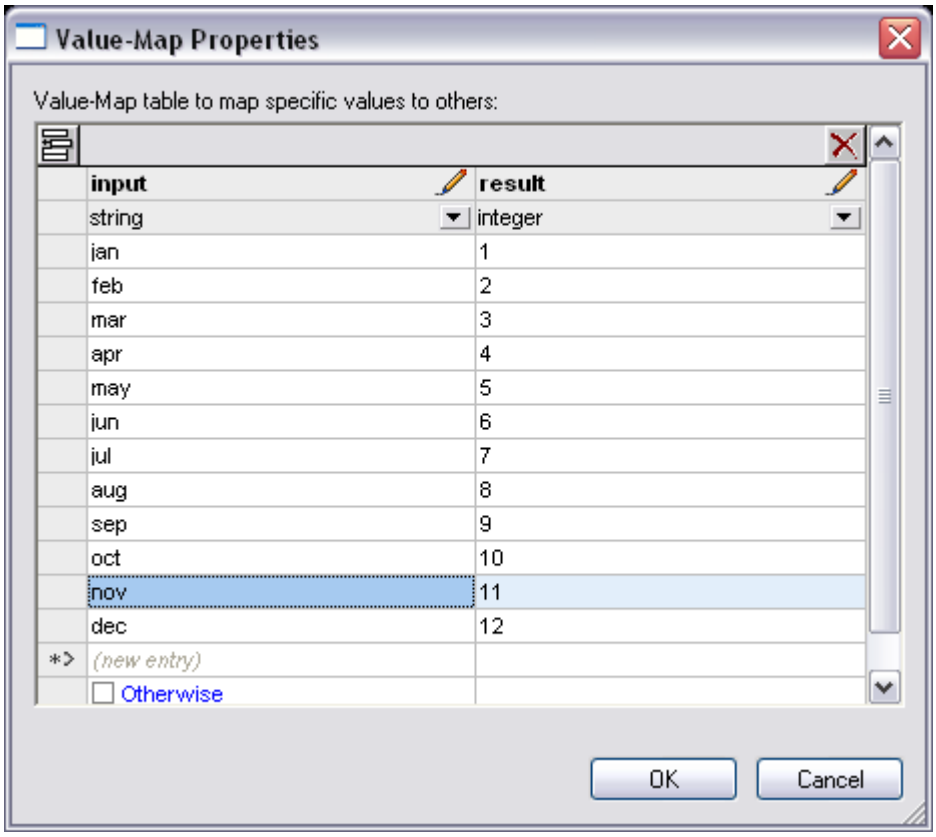


Once the source and target data structures were set up in the MapForce interface, the majority of the mapping operations were very straightforward, executed by connecting the lines between associated element nodes. However, there were several instances when data needed to be manipulated to achieve the proper EDI output.

Converting data to the X12 format

Because the source file and the X12 template vary widely in both structure and data representation, the IT manager needed to apply additional processing to the data that was being mapped. Using the built-in MapForce function library, he associated a wide variety of functions with the data: aggregation, expression mapping, variable tests, etc. Below is an example of just one of these processes.

The value-map component allowed the IT manager to easily automate the process of translating the month name, represented by three letters of text in the source component, to the numerical format required by the EDI message.



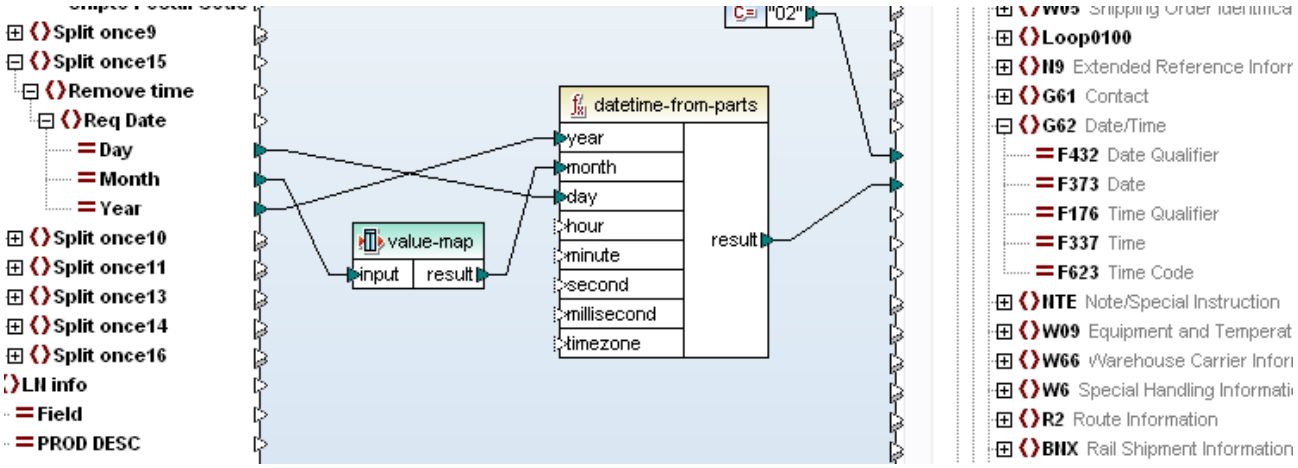
Then, inserting the relevant constants and using MapForce's drag and drop interface to reorder the data,

HD_SO_REQD_DT=14-nov-2007 00:00:00

became

G62*02*20071114

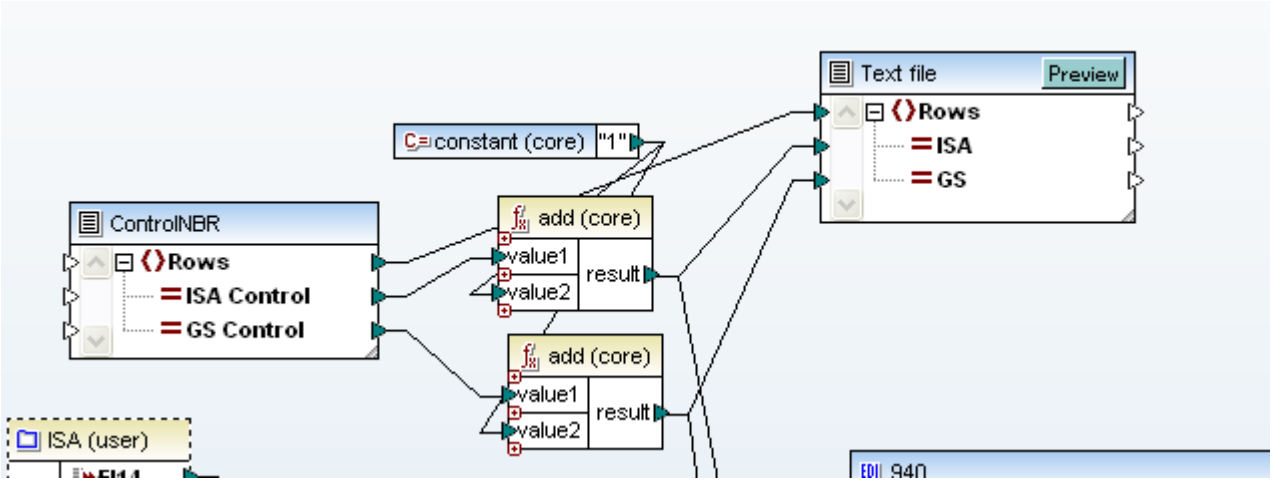
The mapping components used appear below:



Generating control numbers

In the 940 format, there exist two fields that serve as unique identifiers and need to be generated for each document instance. These are the Interchange Control Number (FI12) and the Group Control Number (F28). These identifiers are required elements and allow an audit trail to be maintained across all recorded interchanges in the system.

Creating a unique identification number generation function in MapForce is a simple procedure. The IT manager was able to create and insert two CSV text files, one to serve as the input and one as the output. Using the add function from the MapForce function library as intermediary step in the mapping caused a “1” to be added to each of the control numbers in the input text file.



ALTOVA®

The results of the output text file were then mapped to the appropriate nodes in the EDI 940 template.

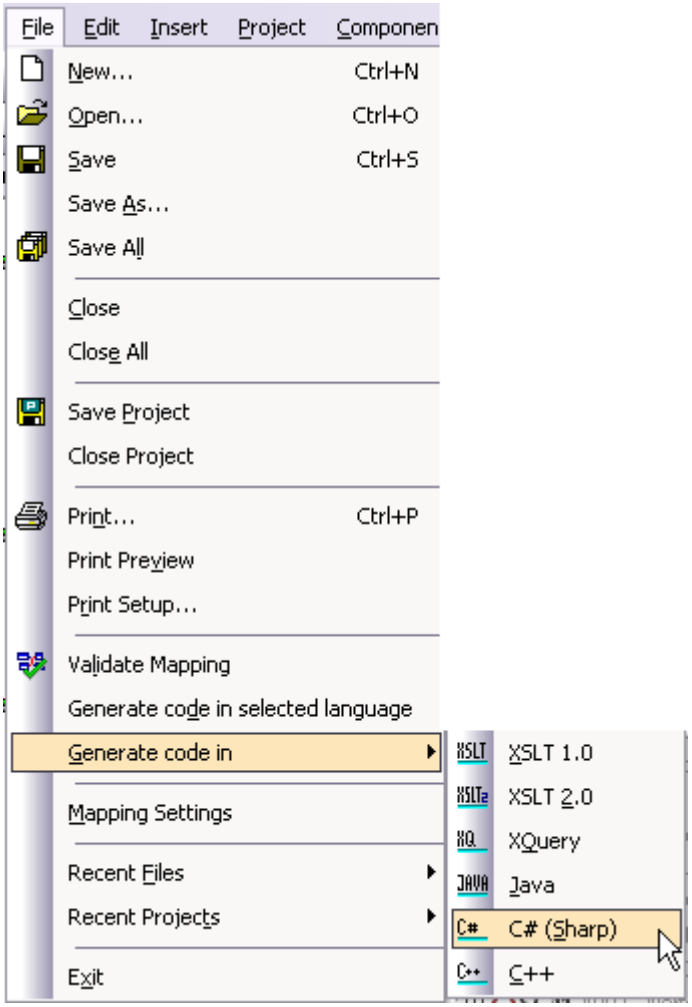
Throughout the mapping process, the IT manager was able to see the EDI data that was being generated on-the-fly by clicking on the MapForce output tab. A sample of this data appears below.

```
ISA*00*          *00*          *01*009262064      *08*311135000T
*080109*0848*U*00501*000000093*1*P*>~
GS*OW*2063228900*311135000T*20080109*084827*93*P*004010UCS~
ST*940*1113126~
W05*N*1113126~
N1*ST*SYSCO MEMPHIS, TN*91*SYS MEMPHI~
N3*4359 BF GOODRICH BLVD~
N4*MEMPHIS*TN*38118-7306~
N1*DE*NATIONAL FROZEN*9*009262064~
N1*SF*AMERICOLD-MOSES LAKE*9*0079091610090~ G62*02*20080115~ G62*10*
20080115~ NTE*WHI*PO Purpose: Original. PO Type: New Order. Note Type: .
Note: .
Transportation Me~
NTE*WHI*thod: Customer Pickup. Ship From WHS: MOSES LAKE/040. LOAD ON
PALLETS~ W66*CC*M********CPU~ LX*1~ W01*1020*CA**VN*74865-00111~ G69*
1/96E SYC COB CORN A~ LX*2~ W01*120*CA**VN*74865-00112~ G69*1/96E SYR
COB CORN B~ LX*3~ W01*200*CA**VN*74865-07843~
G69*12/2.5 SYR CORN B~
W76*1340~
SE*23*1113126~
GE*1*93~
IEA*1*000000093~
```

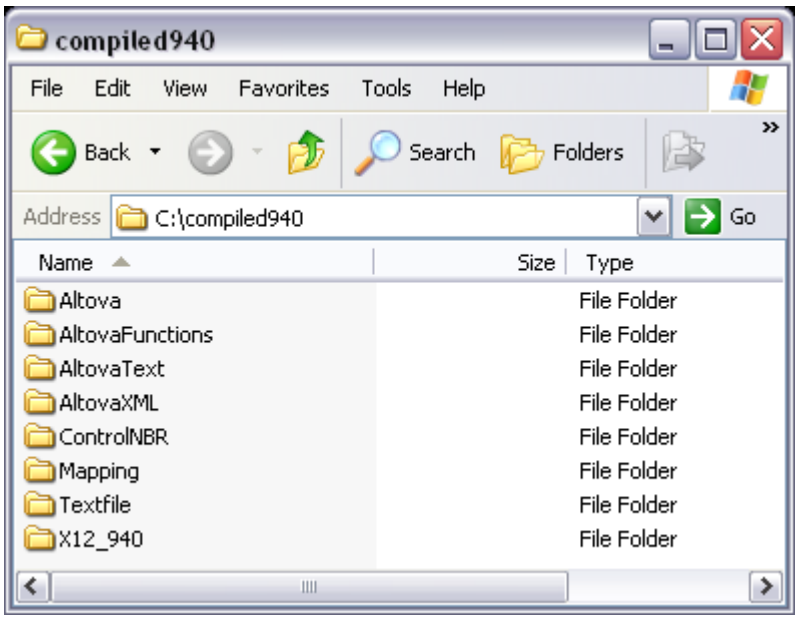
National Frozen Foods now had a valid X12 940 document generated from their original source data. In order to automatically reproduce this process for future iterations, they needed to generate code that would perform these operations.

Exporting, compiling, and executing the code

Once the mapping had been completed, it was time to generate the mapping code so that each shipping order initiated at National Frozen Foods would be instantly available for EDI transmission.



The IT manager chose to generate code for Visual Studio 2003 in C#, which was then compiled and saved as a library to a target directory.



The mapping.sln file was then opened within Visual Studio and compiled as a build solution, integrating all of the various databases and applications needed to complete a warehouse shipping order transaction under one interface.

The Results

Now every time that a warehouse shipping order is initiated by National Frozen Foods, it is run though their MapForce code for immediate and automatic translation to a valid X12 940 format. The EDI document is then imported into TrustedLink for Windows, where it is sent via AS2 (a prevalent security transmission protocol) to the cold storage facility.

National Frozen Foods was able to greatly reduce time-to-market and increase overall efficiency by removing the intermediary step of outsourced custom code development. In addition to eliminating this expensive, time consuming step, using MapForce ensured that data transformation code was written consistently across the entire integration project since it was auto-generated according to industry standards and globally defined parameters, rather than having multiple engineers manually develop the code. This code consistency helped reduce and isolate software bugs while improving overall code readability and reusability.

Using MapForce as their data mapping and code generation tool, National Frozen Foods has succeed in bringing their EDI implementation in-house and now maintain a graphically represented codebase that can be easily modified and re-used for future transactions.

More Information

For the full, detailed MapForce code creation tutorial, see National Frozen Foods IT manager Michael Ellerbeck's blog.

Find out how MapForce can help you with your EDI implementations. Download a free 30-day trial of &mapforce; today!