

Fraser Stream Integration Application Data Sheet

1. FRASER STREAM INTEGRATION	3
1.1 Fraser Stream Output Types	4
1.2 Installation Requirements	5
1.3 Fraser Stream Installation	5
2 FRASER STREAM REPORTS (FRASER STREAM INTEGRATION CLIENT)	5
3 FRASER STREAM REPORTS (STANDALONE VERSION)	7
4 FRASER STREAM INTEGRATION BATCH FUNCTIONALITY	7
5 BATCH TRANFORMATION PROCESSES	9
5.1 Individual Batch Files	9
5.2 Main Batch File	10
 5.3 Using the FSI Batch Creation Interface – Field Explanations 5.3.1 Header Section 5.3.2 Detail Section 5.3.3 Command Buttons 	10 10 11 12
6 DATABASE OUTPUT	12
6.1Defining Database Connection Strings6.1.1Default Connection Strings	14 15
 6.2 Method 1: Extract the data from a text report and load it into a table 6.2.1 Batch Files 6.2.2 Changing Field Types / Enabling the Append Table Option 	15 15 16
 6.3 Method 2: Convert a text report into XML, CSV or Dataset and load it into a table record. 6.3.1 Batch Files 6.3.2 Table Structure to load a report converted into XML, CSV or Dataset into a record 	17 18 18
6.4 Method 3: Extract documents from a text report in XML or Dataset format and load each documents from a text report in XML or Dataset format and load each documents from a text report in XML or Dataset format and load each documents from a text report in XML or Dataset format and load each documents from a text report in XML or Dataset format and load each documents from a text report in XML or Dataset format and load each documents from a text report in XML or Dataset format and load each documents from a text report in XML or Dataset format and load each documents from a text report in XML or Dataset format and load each documents from a text report in XML or Dataset format and load each documents from a text report in XML or Dataset format and load each documents from a text report in XML or Dataset format and load each documents from a text report in XML or Dataset format and load each documents from a text report in XML or Dataset format and load each documents from a text report in XML or Dataset format and load each documents from a text report in XML or Dataset format and load each documents from a text report in XML or Dataset format and load each documents from a text report in XML or Dataset format and load each documents from a text report in XML or Dataset format and load each documents from a text report in XML or Dataset format and load each documents from a text report in XML or Dataset format and load each documents from a text report in XML or Dataset format and load each documents from a text report in XML or Dataset format and load each documents from a text report in XML or Dataset format and load each documents from a text report in XML or Dataset format and load each documents from a text report in XML or Dataset format and load each documents from a text report in XML or Dataset format and load each documents from a text report in XML or Dataset format and load each documents format and load e	ment
into a table record. 6.4.1 Batch Files	19 20
6.4.2 Table Structure to load each document into a table record into XML or Dataset format	20
7 HTML REPORTING STRUCTURE	22

7.1 Using the FSI HTML Reporting Interface	22
 7.2 FSI HTML Reporting Interface – Field Explanations 7.2.1 Header Section 7.2.2 Detail Section 7.2.3 Command Buttons 	23 23 24 24
8 FSI BATCHES AND COMMANDS	25
8.1 Main Batch File	25
8.2Individual Batch File8.2.1FS_Runtime.exe Parameters Description	26 26
8.3 Database Output Method 1 – Extract the data from a text report and load it into a table	27
8.4 Database Output Method 2– Convert a text report into XML, CSV or Dataset and load it into a record 27	table
8.5 Database Output Method 3– Extract documents from a text report in XML or Dataset format a load each document into a table record.	nd 28
8.6 Web HTML Output structure	28
8. PDF OUTPUT	29
8.7 Preconditions	29
 8.8 PDF Output Settings 8.8.1 Change Dataset Fields Types 	33 30
8.9 Create Crystal Reports Format	30

1. Fraser Stream Integration

Fraser Stream Integration (FSI) is a powerful data transformation tool that rapidly transforms structured ASCII text reports and delimited files into various file formats.

- Translate text reports from any source into:
- Excel
- Professional looking PDF
- Word
- XML with style sheets
- Dataset
- CSV
- Extract information from text reports and deposit directly into any database
- Publish information onto the web as HTML, XML with style sheets or PDF
- · Output to XML format for integration with other applications or to share with business partners

Regardless of what the end use will be, the backbone of Fraser Stream technology is the method in which text reports are translated into the aforementioned formats. The *TRANSFORMATION SETTINGS* define how each type of report is to be processed and converted. Mapping transformation settings takes a matter of a few minutes and needs only to be done once. Once the settings are saved, the text report can be converted at the click of a button or scheduled to be processed at any predefined interval. Transformation Settings map information from the text reports header, detail, and footer sections, then translate this information into a single record which in turn can be displayed as a single row of an Excel spreadsheet or exported as unique record into a database.

Transformation Settings map information from the text report's header, detail, and footer sections, then translate this information into a single record which in turn can be displayed as a single row of an Excel spreadsheet or exported as unique record into a database.

Help 🥭 end				
	Tes	st Rep	ort O	000
00 HOME COOLER PRODUCTS				
Description	Site	ABC	UM	Ext GL Cost
BATTERIES, ALKALINE, STD. ENDURACELL	10000	A	EA	20.00 *
CORD, POWER, UK	10000	A	EA	740.00 *
CORD, POWER, USA	10000	A	EA	250.00 *
CORD, POWER, UNIVERSAL ONE SIZE FITS ALL	10000	В	EA	500.00 *
NOMAD(TM) SOLAR POWERED COOLING SYSTEM	15000	A	EA	77,641.79 *
PRODUCT LIN	ie total		_	79,151.79
	Help Send	Help Send Tes OO HOME COOLER PRODUCTS Description Site BATTERIES, ALKALINE, STD. 10000 ENDURACELL CORD, POWER, UK 10000 CORD, POWER, USA 10000 CORD, POWER, UNIVERSAL 10000 ONE SIZE FITS ALL NOMAD (TM) SOLAR POWERED 15000 COOLING SYSTEM PRODUCT LINE TOTAL	Help Mend Test Rep OO HOME COOLER PRODUCTS Description Site ABC BATTERIES, ALKALINE, STD. 10000 A ENDURACELL CORD, POWER, UK 10000 A CORD, POWER, UK 10000 A CORD, POWER, USA 10000 A CORD, POWER, UNIVERSAL 10000 B ONE SIZE FITS ALL NOMAD (TM) SOLAR POWERED 15000 A COOLING SYSTEM PRODUCT LINE TOTAL	Help Send Test Report 0 00 HOME COOLER PRODUCTS Description Site ABC UM BATTERIES, ALKALINE, STD. 10000 A EA ENDURACELL CORD, POWER, UK 10000 A EA CORD, POWER, USA 10000 A EA CORD, POWER, UNIVERSAL 10000 B EA ONE SIZE FITS ALL NOMAD (TM) SOLAR POWERED 15000 A EA COOLING SYSTEM PRODUCT LINE TOTAL

The following example depicts a typical text report with the subsequent FSI output to Excel:

The same report in Excel format:

×	Microsoft Exc	el - Test Report	0000 FS 0000	.CSV				
	Eile Edit Vi	ew <u>I</u> nsert F <u>o</u> rm	at <u>T</u> ools <u>D</u> ata	Window Help 🔗 end				Ту
Ľ	i 📂 🔛 🔒 🤅	🗟 🎒 🗟 🚏 🛛	X 🖹 ∽ +	🍓 Σ 🛛 🛃 📶 😰 🐥 Arial	~ 10	• B .	<u>z</u> <u>u</u> ≣	s = = = = *
	J10 🗸	fx						
	A	В	C	D	E	F	G H	I
1	Product_Line	Prod_Line_Desc	Item_Number	Description	Description2	Site	ABC UM	Ext_GL_Cost I
2	1000	HOME COOLER	30-100	BATTERIES, ALKALINE, STD.	ENDURACELL	10000	A EA	20
3	1000	HOME COOLER	22-100	CORD,POWER,UK		10000	A EA	740
4	1000	HOME COOLER	22-120	CORD,POWER,USA		10000	A EA	250
5	1000	HOME COOLER	22-130	CORD, POWER, UNIVERSAL	ONE SIZE FITS ALL	10000	B EA	500
6	1000	HOME COOLER	10-15001	NOMAD(TM) SOLAR POWERED	COOLING SYSTEM	15000	A EA	77,641.79

Fraser Stream Integration is easy to use, requires very little technical know-how, and costs a fraction of what is currently offered on the market today. This Microsoft Windows based tool can be used as a stand alone desktop application or run on a server as an enterprise level ETL (extract load transform) solution.

1.1 Fraser Stream Output Types

Fraser Stream Integration (FSI) transforms ASCII text reports or delimited files into EXCEL, PDF, CSV, HTML or XML with Style Sheets.



- 1. Click on the Select a Text Report button
- 2. Select the folder: C:\Program Files\Fraser Stream\Fraser Stream Integration\TextReports
- Select the text file: Test Report 0001.prn (Transformation settings will automatically appear in the drop down list)
- 4. Click on the Text button to view the report in its original format in Notepad

- 5. Click on the **CSV Excel** button to see the text report in Excel format. The application will extract report header, details line and footer information and display each record on a single Excel row
- 6. Click on the HTML button to see the text report in HTML format
- 7. Click on the XML button to see the text report in XML format
- 8. Click on the **XML with Style Sheets** button to open the XML file using a Style Sheet. You may customize the Style Sheet to display your company's logo, fonts, colors, background or any other HTML elements
- 9. The PDF, MS Word and MS Excel are created with Crystal Reports. Please see PDF Output chapter

Transformation Settings must be mapped in order for FSI to work. Fortunately this process only needs to be done once. Open FSI and click on the 'Select a text report' button to choose the text file you want to transform. Fraser Stream Integration will scan the text report and check to see if transformation settings have been mapped for this report in the past.

NOTE: FSI does not rely on the report name to determine if transformation settings exist. Instead, FSI looks at the structure of the report and identifies the precise positions of certain key characteristics such as the header, detail, and footer sections to determine if this file has been mapped before.

If a specific type of report has already been mapped, a transformation setting will appear in the drop down list (see screen shot). A report can have multiple transformation settings. For example one set may include each detail column i.e. price, quantity etc...A second set of settings may only capture the header section and the sales order number. Fraser Stream Integration will display all previously mapped settings. Select the settings you wish to use from the drop down list. If no settings exist, you will need to create them. Please download the PDF file Fraser Stream Reports Mappings.

Assuming transformation settings exist, simply select the settings from the drop down list and click on any of the output buttons to view the data in the different formats.

Fraser Stream Software offers a free 30 day trial of Fraser Stream Integration. You may download this free version from the *Downloads* section of <u>www.fraserstream.com</u>.

Follow these steps to install the program:

1.2 Installation Requirements

1. Supported Operating Systems:

Windows 2000, Windows 98, Windows ME, Windows NT, Windows Server 2003, Windows XP, Windows Vista

- 2. Processor: Minimum133-MHz Intel Pentium-class processor
- 3. Memory: Minimum128 MB of RAM, 256 MB recommended

1.3 Fraser Stream Installation

- 1. Go to: www.fraserstream.com/downloads
- 2. Click on Download Now For the software you want to install
- 3. Click Run to install the software
- 4. The software will be installed in the folder: .\Program Files\Fraser Stream\Fraser Stream Integration
- 5. To Start Fraser Stream Integration: Start\All Programs\Fraser Stream Integration

2 Fraser Stream Reports (Fraser Stream Integration Client)

Fraser Stream Reports - Client has the same functionality as Fraser Stream Integration except batch functionality. Fraser Stream Reports - Client installation allows a company to create transformation settings for a large number of text reports that are shared by all company's users. Fraser Stream Reports – Client doesn't need a registration code but it has to be registered with Fraser Stream Integration (See step 3 below).

Fraser Stream Reports - Client must have access to the Shared Folder. (See step 1 below).



- 1. Create FSI Shared folder.
 - Create a network folder where Fraser Stream Reports users have access. This will be the Fraser Stream Shared Folder
 - Unzip "FSI_Shared.zip" file from "C:\Program Files\Fraser Stream\Fraser Stream Integration\Deployment" into the network folder
 - Update the Shared Folder path in "Files\Options" menu with "[Network folder]\FSI_Shared and click Save button

Options			_ 0
Close	Options		
Regional Options: Decimal Separator: Date Separator:	dot		
CSV Options:			
CSV Separator: Apostrophe for leading zeroes (yes/no):	yes	Number of Rows in Datagrid	200
Create DTD for XML file:	no	Encoding iso8859-2	
Generate Activity Log	yes	Real Time Processing	no
Runtime XSLT Path:			
C:\Program Files\Fraser Stream	\Fraser Stream	Integration\XMLOutput\XSLT	
Shared Folder			
F:\App_Files\FSI_Shared			
			Save

2. The transformation settings are created using Fraser Stream Integration and published into the Shared Folder on the network. Click "**Publish FSI Settings**" from "**Files****FSI Clients Access Setup**" menu to publish the transformation settings.

Gu	FSI Clients	Access Setup	
_	User_Name	Product_ID	
•	User 1	9347997347988	
*			

- 3. Download and Install Fraser Stream Reports (client version) on a PC and register it with Fraser Stream Integration. To register Fraser Stream Reports:
 - Open "\Help\ Register" menu on Fraser Stream Reports and copy the Product ID from the registration screen.
 - Open Fraser Stream Integration on the computer where it is installed and input the Product ID and the user name in "Files\FSI Clients Access Setup" screen and click "Save" button.
- 4. Download and Install Fraser Stream Reports (client version) on a PC and register it with Fraser Stream Integration. To register Fraser Stream Reports:
 - Open "**Help**\ **Register**" menu on Fraser Stream Reports and copy the Product ID from the registration screen.
 - Open Fraser Stream Integration on the computer where it is installed and input the Product ID and the user name in "Files\FSI Clients Access Setup" screen and click "Save" button.
- 5. Set in Fraser Stream Reports the same Shared Folder as the Shared Folder in Fraser Stream Integration. (See step 1 above).
- 6. After these steps are done Fraser Stream Reports has the same functionality as Fraser Stream Integration. If the transformation settings are created on Fraser Stream Integration then Fraser Stream Report can transform a text report in Excel, PDF, Word, XML or HTML.

3 Fraser Stream Reports (Standalone Version)

Fraser Stream Reports – Standalone Version has the same functionality as Fraser Stream Integration except batch functionality. It has to be registered on each computer where it is installed. The Shared Folder is not mandatory for this version.

4 Fraser Stream Integration Batch Functionality

Fraser Stream Integration (FSI) automates the process of transforming ASCII text reports (or delimited files) into EXCEL, PDF, CSV, HTML, XML or ADO DataSet format. FSI can automatically generate web reports using XML/HTML output or, populate either existing tables or create new database tables altogether in Access, SQL, or Oracle. Use FSI to post sales reports on the web or to populate tables for your Business Intelligence applications.

FSI can integrate Fraser Stream Reports' text file transformation capabilities with other applications and batch processes. FSI contains a simple command structure which can be included in a batch process or integrated into any Windows based application; use the command structure in your Visual Basic, C++, legacy, and ERP applications.

FSI has three distinct functions all of which can be automated using Windows Scheduler or another scheduling tool.

1. Batch Creation

Create batch files that automatically select, transform, and deposit your text reports into any folder in CSV, HTML, XML, PDF, and DataSet format. Your destination folders can be individual user folders or web report folders for your entire organization.

2. Database Output

Text report or CSV files can be output directly into a database table. FSI provides two distinct methods of outputting data to a table. FSI can either pre-populate an existing table or generate a new table on the fly. FSI supports any database format including MS Access, SQL, and Oracle.

3. HTML Reporting Structure

Create HTML page hierarchies that contain links to the folders that contain your newly transformed files. These web pages can be made available through your intranet or on the Web and secured directly through the folder security settings.

Fraser Stream Integration can automate the transformation of potentially thousands of reports daily. The following diagram outlines the various data sources that can feed Fraser Stream Integration. FSI in turn can transform and output the text or delimited file data into a variety of formats for further use throughout the organization.



FSI is able to output the original data into a folder from where it can be accessed directly, or, output it into a database table through one of two methods.

Key Benefits

Fraser Stream Integration has all the benefits of Fraser Stream Reports plus:

- 1. automatically transforms the text or delimited files directly into CSV, PDF, and web formats
- 2. runs as a simple command that can be part of a batch process or can be integrated into other applications
- 3. facilitates information exchange from one application to another using XML (the standard format for application information exchange today)
- 4. transforms ASCII text reports into Microsoft ADO DataSet in XML format this format can be used by ASP applications or can be exported into other databases
- 5. improves business to business data exchange by quickly extracting, transforming, and delivering data to business partners via CSV, XML, PDF, or Data Set format
- 6. creates a unified report format from different data sources
- 7. is quick and easy to implement FSI does not require a great deal of technical knowledge
- 8. outputs data directly into database tables

- a. creates a table that contains the text report's information ideal for preparing data for use with Business Intelligence tools such as Cognos
- b. stores the text report's information within a single record of a database table (as CSV, XML, or dataset) for use with other applications

5 BATCH TRANFORMATION PROCESSES

The Batch interface creates batch files that automate the text or delimited file transformation process. An FSI batch file is able to take a text or delimited file from any folder, transform it into XML, HTML, CSV or Dataset format, and deliver the output into any other folder within your network. A batch file must also be created when exporting to a database table. (Database output will be covered in the next chapter.)

tch Name : Desktop_user1 Ut Folder : CCiFSI\Inbox/Home\user1 tput Folder : CCiFSI\Outbox/Home\user1 ttings XML : CCiFSI\Outbox/Home\user1 ttings XML : CCiFSI\Inbox/Home\user1 tchive Folder : CCiFSI\Inbox/Home\user1\Archive chive Folder : CCiFSI\Inbox/Home\user1 quence : 10 Run As Service Database Record Output Connection: Imput_File coutput Type : PDF PDF PDF PDF PDF PDF PDF PDF PDF PDF pdf*.pm output Options: *							. [
ut Folder : C\FSNIhox/Home/user1 tput Folder : C\FSNIhox/Home/user1 ttings XML : C\FSNIhox/Home/user1 ttings XML : C\FSNIhox/Home/user1 chive Folder : C\FSNIhox/Home/user1\Archive chive Folder : C\FSNIhox/Home/user1 quence : 10 Polete Input Files Database Record Output Connection:	atch Name : Deskto	p_user1	•	Create / Update	Batch	Delete Batc	:h
tyut Folder : C\FSI\Outbox/Home\user1 ttings XML : C\Forgram Files\Fraser Stream\Fraser Stream Integration\SetFiles\Settings_XML.xml chive Folder: C\FSI\Inbox/Home\user1\Archive Active Batch Delete Input Files Database Table Output Connection:	put Folder : C:\FSI\l	1box\Home\user1					
tings XML: C\Program Files\Fraser Stream\Fraser Stream Integration\SetFiles\Settings_XML.xml chive Folder: C\FSN\Inbox\Home\user1\Archive quence: 10 P Run As Service Database Record Output Connection: Coutput Type: PDF PDF pdf.pml Setting Used : Output Options: Save As File or Table	utput Folder : C:\FSI\0)utbox\Home\user1					
chive Folder: CxIFStl(hox/Homel/user1/Archive Active Batch Delete Input Files quence: 10 PRun As Service Database Record Output Connection: Output Type: PDF InputFile: MILXSL web*.pm CSV exced: pdf*.pml PDF POF Pot	ettings XML : C:\Prog	ram Files\Fraser Stream\Fr	raser Stream Integration\Set	tFiles\Settings_XML.xml			
Active Batch quence: 10 Parabase Table Output Connection: ImputFile: PDF ImputFile: Output Type: PDF Setting Used : Output Options: ImputFile: Save As File or Table	rchive Folder: C:\FSI\l	1box\Home\user1\Archive					
quence : 10 Image: Record Output Connection: Image: Record Output Connection: Output Type : PDF Image: Output_Type Input_File Setting_Used Output_Options Save_As_File InputFile : Output_Type Input_File Setting_Used Output_Options Save_As_File Ddf*,pml PDF PO* pm Input_File Input_File Setting Used : * 00* pm Input_File Output Options: * 9D* pd* pm Input_File Save As File or Table * Save As File or Table Save As File or Table	Active Batch	Delete Input Files	Database Table	e Output Connection:			-
Output Type : PDF InputFile : odf*pml PDF Setting Used : Output Options: Save As File or Table Name :	equence 10 _						
Output Type : PDF Output_Type InputFile : >pdf*,pml > PDF > Pdf*,pm > Save As File or Table		Run As Service	Database Reco	ord Output Connection	n: 🗆		•
PDF Output_Type Input_File Setting_Used Output_Options Save_As_File pdf*.prnl XMLXSL web*.prn CSV excel*.prn pdf*.prnl PDF PO*.prn Input_File Setting Used : * PDF pdf*.prn Output_Options: * * *	Output Type :						
Output_Type Input_File Setting_Used Output_Options pdf*,prn XMLXSL web*,pm Input_File pdf*,prn PDF PO*,pm Input_File PDF PO*,pm Input_File PDF pdf*,pm Input_File	PDF	•					
Input File : XMXXL web*pm pd*pm pd*p		Output_Ty	/pe Input_File	Setting_Used	Output_Options	Save_As_File	
odf. pml PDF PD*pm Input_File Setting Used : * # Input_File Output Options: * * *	InputFile :	XMLXSL	web".pm				
Setting Used : * PDF pdf pm	pdf*.pm	PDF	PO*.pm		Input File		
Save As File or Table Name :	Cotting Lload	PDF	pdf*.pm				
Output Options: Save As File or Table Name :	Setting Used .	*					
Output Options: Save As File or Table Name :							
Save As File or Table Name :							
Save As File or Table Name :	Output Options:						
Save As File or Table Name :	Output Options:	_					
	Output Options:						
	Output Options: Save As File or Table Name :	•					
	Output Options: Save As File or Table Name :	•					

The entire FSI batch structure revolves around two kinds of batch files. The **Main** batch file and the **Individual** batch files (which can be run separately or together through the Main batch file).

5.1 Individual Batch Files

The Individual batch files are created through the FSI Batch interface seen above (in the tools/generate batch on the menu bar). Each individual batch file essentially requires 3 elements:

- 1. A source folder
- 2. A transformation setting
- 3. A destination folder

The key is that an individual batch file can process one or thousands of text files provided they all:

- originate in the same input folder, and
- the transformed files are deposited in the same destination folder

An individual batch file can process the same text file multiple times using different transformation settings as well as transforming the file into various file formats (even with the same transformation settings) such as XML, HTML, and Dataset as long as the output is deposited into the same destination folder.

Note: If the input and output destinations change, a new batch file will need to be created.

5.2 Main Batch File

Each time a new individual batch file is generated; the command to run this file is automatically appended to a Main batch file. When the main batch file is run, every single individual batch file is also run. There are provisions to disable individual batch files from running; this will be covered in the subsequent sections.

The Main Batch file *Main_Batch.bat* is stored by default in: C:\Program Files\Fraser Stream\Fraser Stream Integration\Batch\

This file can be scheduled to run, or can be manually run to execute all the individual batches. Individual batches can also be scheduled to run at separate time intervals by copying the specific command from the Main Batch file and pasting into a separate .bat file altogether.

5.3 Using the FSI Batch Creation Interface – Field Explanations

Each individual batch requires an Input Folder that contains the text files, the Transformation Settings to be used, and the Output Folder where the file(s) are to be deposited.

Please refer to the previous screen shot.

5.3.1 Header Section

This section defines the input, transformation, and output paths.

1. Batch Name

Enter a unique batch name in the Batch Name Combo box if you are creating a new batch process. Select an existing batch name is you are editing an existing batch process

2. Input Folder

Provide the path to the folder that contains the text or delimited files slated for transformation. You must have access to this folder.

3. Output Folder

Provide the path to the output folder where the transformed files will be deposited. You must have access to this folder

4. Settings XML

Provide the path including the file name of the file that contains the transformation settings. These setting must first be defined using Fraser Stream Reports. The settings file by default will be found: *C:\Program Files\Fraser Stream\Fraser Stream Integration\SetFiles\Settings XML.xmI* You must copy Settings_XML.xmI file from .\Fraser Stream Reports\SetFiles in the above folder.

5. Archive Folder (optional)

Enter a path if you want to archive each input file after transformation. FSI will automatically move the input text or delimited file into the specified folder.

NOTE: If running FSI as a Service, either an Archive Folder must be specified or the Delete Input File box must be checked.

6. Active Batch

If this box is checked, FSI will write the command to run this batch into the Main Batch file (see previous section for a definition of Individual and Main batch files)

If the box is unchecked, FSI will still write the command to run this batch into the Main Batch file, but the command will be suspended and will not run each time the *Main_Batch.bat* file is run

7. Delete Input File

If this box is checked, the input file will be deleted after each time it is transformed.

NOTE: If running FSI as a Service, either an Archive Folder must be specified or the Delete Input File box must be checked.

8. Run as a Service

If this box is checked, the batch file will be included in FSI's Windows Service. When the service is activated, each batch added to the service is run automatically every 20-30 seconds. The service scans all input folders as defined in each individual batch file and processes any text files found therein. If no files are found the batch does not run. The service continuously scans the input folders for new files. This service may run indefinitely in the background while the server is running.

9. Sequence

This is an integer number which defines in what order the individual batches within the Main Batch file are to be executed.

Note: The Database Record Output Connections and Database Table Output Connection fields will be covered in section 3 Database Output. These fields define the database connection string for the two methods of database output. Outputting to a database is also part of FSI's Windows Service option.

5.3.2 Detail Section

This section defines the input, transformation, and output file details i.e. names, formats, etc...

1. Output Type

Use this combo box to define the type of output file you want generated by the transformation settings. Select from the following list of file types:

- a. XML The FSI batch transforms the input file into XML
- b. XMLXSL The FSI batch transforms the input file into an XML that has a reference to a style sheet located in the folder XSLT. The XSLT folder must be in the same output folder where the XML file is to be stored.
- c. XMLDTD The FSI batch transforms the input file into an XML with a reference to a DTD file stored in the same output folder where the XML file is to be stored.
- d. CSV The input file is transformed by FSI batch into a CSV file. This file can be opened with MS Excel
- e. DATASET The ADO Dataset format is generated from the input file.
- f. PDF The input text file is converted to PDF

2. Input File

Enter the specific name of the input file or specify a pattern if you require multiple files to be transformed. If you enter txt - FSI will transform all the text files in this folder

3. Setting Used (optional)

If the Fraser Stream Reports transformation setting file (specified in the header section) contains more than one set of transformation settings for this particular input file(s) you must specify which settings to use. Otherwise, by leaving the field blank, FSI will use the first set of setting that match the input file(s).

4. Output Options

The output options will allow you to select a naming convention for your output files.

- a. (blank) The output file name is generated using the report name and the name of the transformation settings used.
- b. Extend Use this option if a unique output file name is needed. The Extend option will append the (year, month, day, hour, minute, second, millisecond) to the end of the output file name as define in option *a* thereby creating a unique file name each time.
- c. Input_File Use this option is you want the output file name to be the same as the input file name (only the file extension will differ according output type (i.e. xml instead of txt).
- d. Save_As Use this option for a specific output file name. The file name has to be specified in Save As File parameter.

5. Save As File

Enter a specific name for the output file (including file extension). The Output Option parameter must have the value **Save_As**.

5.3.3 Command Buttons

To update the individual and main batch files:

Add Batch Line

As mentioned in the previous section, an individual batch can process multiple text files provided they originate in the same folder and are deposited in the same place after transformation. Each batch can convert the input files into multiple file formats utilizing different transformation settings and applying different naming conventions to the final output files.

Use this button to add individual command lines to single batch. For example in the screen shot we see that the user has specified certain text files in the input folder to be processed one way, while files with the extension .prn are processed differently. A specific text file can be transformed using a particular set of transformation settings and saved with a unique name that utilizes the time the file was transformed.

Create / Update Batch

This button will either create or update the individual batch to include all the commands as specified in the detail section. The batch is stored in the folder:

C:\Program Files\Fraser Stream\Fraser Stream Integration\Batch\BatchDetail

This function also updates or (first time) creates the **Main_Batch.bat** batch that contains all the individual FSI batches. The Main_Batch.bat is stored in the folder: *C:\Program Files\Fraser Stream\Fraser Stream Integration\Batch*

Delete Batch

This button deletes the individual FSI batch. Be sure that the correct batch has been selected from the Batch Name field.

6 Database Output

As an extension of Fraser Stream Integration's batch creation functionality, FSI can output data directly into a database table. FSI still requires that the original input text file be transformed into CSV, XML, or Dataset and deposited in an output folder. From this point FSI either appends the data to an existing table or generates a new table and populates it with the data from the output file. This process is part of the individual batch file and can be run automatically as part of FSI's Windows Service functionality.

Your data can be made available to virtually any system whether it is a custom application or your Business Intelligence solution.

Fraser Stream Integration provides two methods of exporting information to a database table.

Extract the data from a text report and load it into a database table

FSI creates a table into a database and loads the information extracted from the text file in that table. The fields names are the same as the fields generated with the transformation settings tool. FSI provides the option to index select fields within the table to ensure that only unique records are appended to the table each time the batch is run.

le Edit Frans	st. View Help												
ODOCDUZ. D	en Derr Deite			5.9.3 PU	rchase or	dens by I	тел не	2790					C
age: 1					Your N	iame Here							T
upplier Nar	me o	inder Li	n qey	ordered Qty	open UN	Purchas	e Cost		Ext	Cost	Due	Perfo	nn (
	in cosh	- RWER - HE											
vision: A	A												
000000 Hy	droxco Gas Proce P droxco Gas Broce B	00018 3	1	100.0	30.0 EA 75 0 EA		1.00		4	$\frac{5.00}{1.25}$	03/02/	94 03/02 94 03/02	/94 L /94 L
01000 Ge	neral Supply Cor P	0:020	î	2010.0 1	960.0 EA		1.00		1,96	0.00	03/02/	94 03/02	/14 B
	en Trop Torol			3336 S 3	095 A F1				3.67	4. 56			
6161	se rem local			6699.8 6	900 V EA				6,97	0.23			
ee: 22-31)	0 0040	, Rowen, Euro	ope, w	122									
vision: A	A decide das Broce B	- 8 00 M	2	125.0	100 O IV		1.00		0	0.00	10.000	as na ina	(94. i
1030000 Hy	droxco Gas Proce P	00019	2	100.5	75.0 EA		1.00		7	1.25	93/92/	94 03/02	/94 i
01000 661	neral Supply Cor P	05020	2	1000.3	500.0 EA		1.00		50	0.00	03/02/	94 - 69/402) 64	/94 <u>H</u>
ALAAA	and the second of the second	29	-							101 X 102 X 10	- 1050 M L 1010	New York Concerning Street Stree	
09000 we:	st Coast Electro R	a	2	-500.3 -	-300.0 EA		AL 99						
L.	st Coust Elèctro R	a :	2	-500.3 -	-500. 0 EA		2.94		- 7 4		***		J
••••••	st Colst Electro R	1	2	-500.3 -	-500, 0 EA		2.04				***	-	1
Vicrosoft Acc	st Colst Elèxtro R 2005 - [Sempleilezort : 1	0. ; [ole]	2	-500.3 -	500.0 EA		2.04				***		1
osooo we Crosoft Acc De Ede S - La Par	st Coast Electro R coss (SampleBoast : 1 Yew Just Fynat Br	1. : [auto] sondo [auto [.0] (9.15]	2 	- 500.3 -	200,0 EA	A-10	2.00				Туреас	uestion for he	1
breadt Act	st Coast Electro R coso (SampleRoport) 1 Yem Jaert Fyna B Cal (2, * 1 × 2 = 15) Rem_Descriptio	Taole) sords Isok	Window XIII	- 500.3	No NX ()	∕3 - ⊕ ase_Order I.	n Gty_O	indered	Oty_Open	UM	Type a c	uestion Forthe	, , , , ,
brosoft Acc brosoft Acc ble Edit 5 - La Sal tem Number 2100	st Godst Electro R ccss (SempleRoperti) 것en jeset Fignat Br cg (고, 우) 또 가 가	ia. Intel In	2 windor 21 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2	- 500.3 - 5 tele 5⁄Y3 ∨ A Suppler, Nan Hydroxes Gas P	Po Por Purch	23 - 1 - 0	n <u>Gty_0</u>	indened	Cty_Open 50	UM	Typea o Purchase	uestion Forthe Ext_Cost 45	×
osooo we deresoft Acc ole gate : - La that tem Number 22100 22100	st Coast Electro R cos - SampleRopert - 1 yen Jean Agnat B co Descriptio CORDOWERUK CORDOWERUK	1. : 1. :	2 wrds- 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	- 500.3 - Brie Gr Va V (A) I Suppler Nan Hydroxe Gas P	PE IX 3	23 - 1 00 ase_Order L 18 19	n <u>Gty_0</u> 1	ndered 100 125	City_Open 50 75	UM EA CA	Type a c Purchase 1	Lestion Forthe Ext_Cost 71.25	* •
00000 we:	st Cosst Electro R cos (SempleRoord) Yee Jeet Fond B CoRDPOWERUK CORDPOWERUK CORDPOWERUK	a. sords Iook 1 7 (%,) 24 on Su Su Su Su Su Su Su Su	window 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2	Beb Grant Suppler Nan Hydroxee Gas P Hydroxee Gas P Hydroxee Gas P	Pro BX C	23 - 1 00 ass_Order L 18 19 20	n <u>Oty_0</u> 1	indened 100 125 2010	City_Open 50 75 1960	UM EA EA	Type a c Purchase 1	Lestion Forha Exit_Cont 45 71.25 1,960.00	* • •
01000 we:	st Cost Electro R	a. sends Isek 17 (2), 24 m Su 500 500 PE,WEST 500	2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2	- 500-3 - 500-3 - 500-30-3 - 500-3 - - 500-3 -		23 - 1 00 asse_Order L 18 19 20 18	n Oty_0 1 1 1	rderes 100 125 2010 125	0ty_0pen 50 75 100	UM EA EA EA	Type a c Purchase 1 1 1	Ent_Cont 45 71.25 1.950.00 90	
Kresoft Acc (6 gds : 	St Coast Electro R Coast SampleRoord a 1 Yen Jean Agna B Cal D. C. Romon SampleRoord a 1 Mem Descriptio CORD POWERUK CORD POWERUK CORD POWERUKO CORD POWERUKO CORD POWERUKO	a sardi Jock ordi Jock ordi Jock ordi Jock ordi Jock ordi Statistical Statis	2 2 2 2 2 2 2 2 2 2 2 2 2 2	tele Suppier Nan Hydroxe Gas P General Supply C Hydroxe Gas P Hydroxe Gas P	Mol O EX Mol O EX <t< td=""><td>2 - 00 ase_Order L 18 19 20 18 19</td><td>n Cty_0 1 1 1 2 2</td><td>ndened 100 125 2010 125 100</td><td>City_Open 50 75 1960 75</td><td>UM EA EA EA EA</td><td>Type a c Purchase 1 1 1 1 1</td><td>Est_Cost 45 71.25 90 71.25</td><td></td></t<>	2 - 00 ase_Order L 18 19 20 18 19	n Cty_0 1 1 1 2 2	ndened 100 125 2010 125 100	City_Open 50 75 1960 75	UM EA EA EA EA	Type a c Purchase 1 1 1 1 1	Est_Cost 45 71.25 90 71.25	
Alteresoft Acc be gat 1 - La 12 22100 22100 22110	St Cost Electro R Cost (SampleRoort) Yew Jeet Fynd D Coll 2 C A Corporations Corp	factor secrets jack secrets jack jack in Su 900 900 900 9	2 2 2 2 2 2 2 2 2 2 2 2 2 2	- 500,3 - Beb Suppier_Nan Hydroxce Gas P Hydroxce Gas P Hydroxce Gas P Hydroxce Gas P Hydroxce Gas P	A A	23 - ₩ ase_Order L 19 20 18 19 20 20	n Cty_0 1 1 2 2	ndened 100 125 2010 125 100 1000	0ty_0pen 50 75 100 75 500	UM EA EA EA EA EA	Types of Purchase 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	uestion for he Ext_Cost 45 71.25 1960.00 90 71.25 500	
Afresoft Arc 0e 64 : -1 64 53 : -1 64 54 53 : -1 64 54 54 54 54 54 54 54 54 54 54 54 54 54	St Cost Electro R Cost (SempleRoser) I Yew Jeert Fand Br Coro PowEruk Coro PowEruk Coro PowEruk Coro PowEruk Coro PowEruko Coro PowEruko Coro PowEruko Coro PowEruko Coro PowEruko	foole1	2 2 2 2 2 2 2 2 2 2 2 2 2 2	-500.3 - 500.3 - 500.3 - 500.5 - 500.	Po at 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2	20 - 1 00 ase_Order L 19 20 18 19 20 20	n Cty_O 1 1 1 2 2 2	ndened 100 125 2010 125 100 1000 -500	2ty_Open 50 75 100 100 75 500 800	UM EA EA EA EA EA	Type a c Purchase 1 1 1 1 1	Est_Cost 45 71.25 90 71.25 500 600	
Koresoft Acc (e gat : 1	St Cost Electro R Cost (SempleRoord) Yee Jeet Fond B CoRDPOWERUK CORDPOWERUK CORDPOWERUKO CORDPOWERUROF CORDPOWERUROF CORDPOWERUROF CORDPOWERUROF CORDPOWERUROF CORDPOWERUROF CORDPOWERUROF	Table] sends Iask m Sa 500 2000 2000 2000 2000 2000 2000 2000	2 window pplier 00000 00000 00000 00000 00000 00000 00000	-500.3 - 500.3 - 500.3 - Supple Nan Hydroxe Gas P Hydroxe Gas P General Supply G West Coast Elec Hydroxe Gas P	>000,0 EX >> >	23 - 1 00 18 19 20 20 19 20 20 20 20	n Cty_0 1 1 1 2 2 2 3	ndened 100 125 2010 125 100 1000 -500 200	Cty_Open 50 75 100 100 200 200	UM EA EA EA EA EA EA EA	Type a c Purchase 1 1 1 1 1 1 1 1	Ext_Cost 45 71.25 90 71.25 500 4500	* • • • • • • • • • • • • • • • • • • •

Convert a text report into XML, CSV or Dataset and load it into a database record

The second method of exporting file contents to a database is far more complex and is more suitable for more advanced development. This method actually takes the entire contents of a single CSV, XML, or Dataset file and stores them in a single field within a table.

The remaining fields in each record will store information such as original text file name, file transformation date, the original contents of the text file, output file paths etc... In this scenario each report is appended to a single table record. The table structure is more suitable for staging data for further development, tracking historical changes or, for managing web content. (It is far easier to manage data within a database as compared to a folder structure)

Fraser Stream Integration can export data directly into this format; however the following table structure must be strictly observed including the exact field name convention.

	Field Name	Data Type
8	ID	AutoNumber
	date	Date/Time
	infilename	Text
-	infilepath	Text
	infileload	Memo
Ĩ	outfiletype	Text
	outfilename	Text
	outfilepath	Text
U	outfileload	Memo
m	fsifield1	Text
	fsifield2	Text
	fsifield3	Text
	01-101000-0000	
111		
233		

6.1 Defining Database Connection Strings

In order to output your data to database table, you must define the connection string for the database regardless of the database output method you use.

To define the connection string, select **Database Connection** from the **Tools** menu on the main Fraser Stream Integration page.

🗾 Database Connection			
Close Datab	ase Connection Strings		
Connection:	Connection1	Save Connection	Delete Connection
Connection Description :	Connection String for Table Output		
Encrypted Connection	Saved encrypted connection strings are not displayed by F to retype the connection string if you need to change it.	FSI. You will have	
Connection String : Provider=Microsoft.Jet.OLEDB.4.0;	Data Source=C:\Program Files\Fraser Stream\Fraser Stream Ii	Integration\DB\FSIDbTables.m	db;
-			
Z Database Connection			
Close Database	ase Connection Strings		
Database Connection Close Database Connection:	ase Connection Strings	Save Connection	Delete Connection
Database Connection Close Database Connection: Connection Description :	ase Connection Strings SQL_Con1 SQL Database Connection Example	Save Connection	Delete Connection
Database Connection Close Database Connection: Connection Description : Encrypted Connection	ase Connection Strings SQL_Con1 SQL Database Connection Example Saved encrypted connection strings are not displayed by F to retype the connection string if you need to change it.	Save Connection FSI. You will have	Delete Connection
Database Connection Close Database Connection: Connection Description : Encrypted Connection Connection String :	ase Connection Strings	Save Connection	Delete Connection

Field Definitions:

Connection

Select a connection for your database. You can setup as many connections as necessary. By default there will be three connections along with SQL_Example connection. Connections 1 and 2 by default point to a pair of MS Access Databases included with Fraser Stream Integration and are ideal for testing purposes. (see section 3.1.1 for details)

To add more connections, simply type in a connection name i.e. MyConnection

Connection Description

Enter a description for you connection

Encrypted Connection

You may encrypt your connection string by selecting this check box. Your connection string will be encrypted in an xml file.

Connection String

Enter the actual connection string – see screenshot above Use the Save Connection and Edit Connection command buttons to manage your database connections. Once your connection has been setup the next step involves creating a batch file as described in Chapter 2 with a few minor differences.

6.1.1 Default Connection Strings

This section identifies the default database connection strings which correspond to the two different methods of outputting information to a database. Refer to section 3.2 and 3.3 for a detailed explanation of the two methods.

Connection 1: This connection string is for database output method 1 (section 3.2). By default it points to an MS Access database in the root directory under .\Fraser Stream Integration\DB\FSITable.db

Connection 2: This connection string is for database output method 2 (section 3.3). By default it points to an MS Access database in the root directory under .\Fraser Stream Integration\DB\FSIDatabase.db

6.2 Method 1: Extract the data from a text report and load it into a table

This is the standard method of importing data into a database table. Once in a database, you can connect to this information with any range of applications ranging from in house custom solutions to enterprise level Business Intelligence tools such as Cognos and Crystal Reports.

Whether exporting data to a table or to a folder, Fraser Stream Integration still requires that a batch file be created.

6.2.1 Batch Files

Refer to the previous section (Chapter 2) on the FSI Batch Creation Interface. The key difference is that you must now check the box titled:

DATABASE TABLE OUTPUT CONNECTION

Select the database connection string as defined in section 3.1. You may use Connection 1 as a default for this method of output.

Note: For method 1, you do <u>not</u> need to create a table in your database beforehand – as long as the <u>database</u> is created, FSI will create the <u>table</u> automatically. If you use Connection 1 as the default, the string will point to FSI's MS Access database. (found in the root directory under *Fraser Stream Integration**DB**FSITable.db*)

Click on the Save Batch button once you have finished filling in all the necessary fields.

Once the batch has been created two things will occur:

1. In the .*Fraser Stream Integration* \Batch folder in your root directory you will find a file called: TABLE_Batch.bat

This is a master batch file that executes every single individual batch process that involves outputting data to a single table. Note: the process outlined in this section can be repeated multiple times for any number of text reports being outputted to table.

With this file you can de-activate any of the individual batch commands.

 In the .\Fraser Stream Integration \Batch\BatchTB folder you will find the individual batch files that contain the commands to select, transform, deposit, then export to a table a single or a series of like text reports. For example, you may have created a batch called Sales_Order. Within this folder you will find a batch file called TABLE_Batch_Sales_Order.

Each time this batch file is executed the following will occur:

- a) Select input text file
- b) Transform the file to the specified format (ie. CSV, XML, etc...)
- c) Deposit the new file into an output folder
- d) Delete any existing table
 * <u>Note</u> the option exists to continuously append new records to an existing table each time the batch is run, you may create a unique index in the output table to avoid duplicate records see section 3.2.2
- e) Create a table in the specified database (see previous comment)
- f) Import the file contents into the table

6.2.2 Changing Field Types / Enabling the Append Table Option

Under Tools on the main FSI page, there exists an option called Change Fields Types - Create Append Table.

This page addresses two table options:

1. FSI creates a table with the information extracted from the text report. When the batch runs again the table is deleted and a new table is created. FSI will by default assign field types when the database table is created. These field types can be changed after the table has been created.

Select the connection string and the corresponding table you wish to modify from the Select Table list. A list of fields will be listed for the selected table, simply change the field properties by keying in a different field type (a list of permissible field types is listed at the top of the screen i.e datetime, varchar, integer, decimal... see screen shot on next page).

Next time when the table will be created it will have the new assigned field types.

2. Fraser Stream ETL solution (Export, Transformation and Load)

FSI creates a second table called "append table" that has the exact name as the first table but it has "_APP" at the end (the append table will not be deleted and recreated each time the batch is run, it will be created only if the table does not exist). FSI provides the option to create a unique index within the append table. This index will allow you to append only those records that are unique to the index you assign.

If the **Active Append Table** check box (see screen shot on next page) is left <u>unchecked</u>, FSI will create a new table each time the batch is run (deleting the old table and re-creating it along with the field type settings that have been defined in point 1).

If the **Active Append Table** is <u>checked</u>, FSI will create a <u>second table</u> within the specified database connection string. This second table will contain the same field type settings as the first table, the key difference being that the second table will not be deleted and recreated each time the batch is run. Instead, records will be appended to it or overwritten depending on the index you create (or primary key) you create. This second table will have the exact name as the first table with the addition of _*APP* at the end of the name.

<u>Note:</u> The first table will still be deleted and recreated each time the batch is run. It will contain a snapshot of only the latest text file. This table can serve as a validation table that will allow a comparison between your last transformation and the records added to your second Append Table.

Unique Index or Primary Key

If the Active Append Table checkbox is selected, you <u>must</u> create a unique field index to control how the records are appended to the table. Note the column in the screen below called *Unique_Index_Append_Table*. Simply check the field, or combination of fields you want to use for your unique index (or primary key). Each time the batch is run, only unique records will be added to your table while existing records will be overwritten.

EXAMPLE: Suppose you have a sales order report that you are exporting to a table. The index you set is a combination of the Sales Order Number and the Line Number. The first time you run the batch all the contents of the text file will be added to your newly created table. The second time the batch is run FSI will check to see if the combination of sales order number and line number already exist in the table. If they exist, FSI will simply overwrite those records to ensure the most current information is in table. For example the ship quantity for a given item within sales order may have changed that day.

If the combination of sales order number and line number does not exist in the table, FSI will append the records as a new transaction to the table.

<u>Note:</u> After the Append Table has been created you will be unable to change Field Types for that table through FSI unless you first either change the field types manually through the database table or delete the table altogether. Otherwise the batch could fail if you change the field types after the table has already been created.

FS_Date Field

When the Append table is created for the first time, FSI will automatically create a field in the table called FS_Date. This is a date field that captures the exact time that the records are either appended or overwritten in the Append Table. This is a useful field for guerying database activity and may be used to indicate when data was last refreshed.

Field	s Type and Append Table		_	
Clos	^e Change Fie	elds Types - Crea	ate Append Table	
Data	base Connection: SQL Select Table Invoid	Con1 SQL C	atabase Connection Example e_0008	·
S	ave Table Properties	Active Append Table		
The	fields type must be: var	:har, integer, decimal, date		
	Field_Name	Field_Type	Unique_Index_Append_Table	
•	FS_Date	date		
	Invoice	varchar	v	
	Customer	varchar		
	Customer_Name	varchar		
	Purchase_Order	varchar		
	_			
	Ship_To	varchar		
	Ship_To Ship_Date	varchar date		
	Ship_To Ship_Date Inv_Date	varchar date date		
	Ship_To Ship_Date Inv_Date Order	varchar date date varchar		
	Ship_To Ship_Date Inv_Date Order Ln	varchar date date varchar integer		
	Ship_To Ship_Date Inv_Date Order Ln Item_Number	varchar date date varchar integer varchar		l
	Ship_To Ship_Date Inv_Date Order Ln Item_Number Item_Description	varchar date varchar integer varchar varchar		
	Ship_To Ship_Date Inv_Date Order Ln Item_Number Item_Description UM	varchar date varchar integer varchar varchar varchar		
	Ship_To Ship_Date Inv_Date Order Ln Item_Number Item_Description UM Qty_Ordered	varchar date varchar integer varchar varchar varchar varchar		
	Ship_To Ship_Date Inv_Date Order Ln Item_Number Item_Oescription UM Qty_Ordered Qty_Invoiced	varchar date date varchar integer varchar varchar varchar varchar decimal		_

Transaction Log

In .*Fraser Stream Integration* *Log*\ you will find a log file called *Database_Create_Table_App.log*. This file contains a record of every Append Table transaction which includes how many records were updated (overwritten), added, or failed. An explanation is also provided for all records that fail to be processed.

Once all the field types and indexes have been set – simply click on the **Save Table Properties** command button. Then run the batch to begin the process of populating the table(s).

6.3 Method 2: Convert a text report into XML, CSV or Dataset and load it into a table record.

The second method of storing data is more complex in that the entire file output is stored in a single field within a single record in the table. This solution is more suitable for the following scenarios:

Data needs to be staged for various applications. This table structure will allow a developer to quickly access the XML, CSV, or Dataset contents of a file for further utilization.

Historical data needs to be tracked for future analysis – accessing point in time information is far easier through a table than via a complex folder structure

Web reporting is far easier to manage when the data is stored in a table. A web application need only query a record with a specific file ID – then display the XML contents stored within a specific field in that specific record.

This approach greatly reduces development time and steers clear of the traditional approach of generating a series of complex queries to return a record set.

A database connection string must be defined for both database output options. This is the first step – please refer to section 3.1.

6.3.1 Batch Files

Outputting data to a single record requires a batch file.

Refer to the previous section (Chapter 2) on the FSI Batch Creation Interface. The key difference is that you must now check the box titled:

DATABASE RECORD OUTPUT CONNECTION

Select the database connection string as defined in section 3.1. You may use Connection 2 as a default for this method of output.

<u>Note:</u> You need to create a table in your database beforehand - refer to section 6.3.2 for a description of the required table structure. After the initial creation, the batch will append new records each time the batch file is run.

Note: For method 2, you <u>must</u> create a <u>table</u> in your database beforehand – refer to section 3.3.2 for a description of the required table properties. After the initial creation, the batch will append new records each time the batch file is run.

Click on the Save Batch button once you have finished filling in all the necessary fields.

Once the batch has been created two things will occur.

Each time this batch file is executed will follow these steps:

- Transform the file to the specified format (XML, Dataset)
- Append new record to the specified database
- Loads the CSV , XML or Dataset format in the new record.

6.3.2 Table Structure to load a report converted into XML, CSV or Dataset into a record

Table Structure:

Field Name	Data Type
₿▶ID	AutoNumber
date	Date/Time
infilename	Text
infilepath	Text
infileload	Memo
outfiletype	Text
outfilename	Text
outfilepath	Text
outfileload	Memo
fsifield1	Text
fsifield2	Text
fsifield3	Text

This table may be created in any database including: MS Access, MS SQL, and Oracle.

Note: For SQL Server you may import the database and table structure from: .\Fraser Stream Integration \SetFiles\FSDbRecord_export.sql

Note: You must use the appropriate field types for the specific database you are using. For example, if you are using a SQL database, you must ensure that a memo field is set to *VARCHAR* etc...

Field Description:

- 1. ID
 - Record ID this field must be the sole primary key
- 2. date
- The date and time that the text file transformation took place when the actual batch was run **3.** infilename
- The file name of the original text file
- 4. infilepath The input location of the original text file
- 5. infileload
- The entire contents of the original text file
- outfiletype The file type of the transformed file (XML, CSV, Dataset)
- outfilename
 The file name of the transformed file
- 8. outfilepath The output location of the transformed file
- 9. outfileload
 - The entire contents of the transformed file in XML, CSV, or Dataset format
- 10. fsifield1
- Unused field 11. fsifield2
- Unused field
- 12. fsifield3 Unused field

6.4 Method 3: Extract documents from a text report in XML or Dataset format and load each document into a table record.

The documents from a text report are converted into XML or Dataset format and each document is loaded into a database record. The documents stored in the database can be published onto the web in HTML or PDF format using a web interface. Fraser Stream Web interface publishes these documents onto the web with security per document type or per customer or supplier code. Using Fraser Stream document publishing functionality no programming is required to publish any documents from an ERP system on the web **Transformation setting for documents extraction:**

The individual documents are extracted from the text reports and loaded into the database only if a **special transformation setting** is defined for the report.

The transformation settings contain following field names from the report header area:

1. doc_nbr (mandatory)

Document Number extracted from the text file (has to unique per document like invoice, purchase order or sales order number)

- 2. addr_code (mandatory)
 - Address Code customer code, supplier code
- 3. fs_field1 (mandatory)
- Field from document header 4. fs field2 (optional)
- Field from document header
- 5. fs_field3 (optional)

Field from document header

- 6. fs_field4 (optional)
 - Field from document header
- 7. fs_field5 (optional) Field from document header

The transformation settings can contain any other fields from the text document.

6.4.1 Batch Files

To output the data to a single record a batch file has to be created.

Refer to the FSI Batch Creation Interface. The key difference is that you must now check the box titled:

DATABASE RECORD OUTPUT CONNECTION

Select the database connection string as defined in section 3.1.

<u>Note:</u> You need to create a table in your database beforehand - refer to section 6.4.2 for a description of the required table structure. After the initial creation, the batch will append new records each time the batch file is run.

Note: For method 2, you <u>must</u> create a <u>table</u> in your database beforehand – refer to section 6.4.2 for a description of the required table properties. When the batch is executed it will append new records for each document extracted from the text report.

Click on the Save Batch button once you have finished filling in all the necessary fields.

Within the .\Fraser Stream Integration \Batch folder in your root directory you will find a file called: DB_Batch.bat

Copy the DB_Batch.bat and customize it to run only the batches that are needed

Example : The batch called Batch_WebDoc.bat

```
REM Fraser Stream Integration - Create and execute a batch
ECHO ON
CALL cd "C:\Program Files\Fraser Stream\Fraser Stream Integration"
CALL ".\bin\FS_CreateBatchDB.exe" "Batch_WebDoc" ""
"C:\FSI\Inbox\Database\DB_Web_Doc\Archive"
CALL ".\Batch\BatchDB\DB_Batch_WebDoc.bat"
```

Each time this batch file is executed will follow these steps:

- Transform the file to the specified format (ie. CSV, XML, Dataset)
- Append new record to the specified database
- Load each document from the text file into a new record in XML or Dataset format.

6.4.2 Table Structure to load each document into a table record into XML or Dataset format

Table structure:

	fsidoc : Table			×
	Field Name	Data Type	Description	
	îs_date	Date/Time	FSI Create Date	T
P	doc_name	Text	Document Description	
Ŷ	doc_nbr	Text	Document Number extreacted from the text file	
	addr_code	Text	Address Code - customer code, supplier code or other code	
	load_type	Text		
	file_load	Memo	The field where the XML file is loaded	
	file_dataset	Memo	The field where the Dataset is loaded	
	file_pdf	Memo	The field where the PDF is loaded	
	fs_field1	Text	Extra field 1 - field from document header	
	fs_field2	Text	Extra field 2 - field from document header	
	fs_field3	Text	Extra field 3 - field from document header	
	fs_field4	Text	Extra field 4 - field from document header	_
	fs_field5	Text	Extra field 5 - field from document header	~
		I	Field Properties	

This table may be created in any database including: MS Access, MS SQL, and Oracle.

The table needs a unique index on doc_name and doc_nbr fields

Note: You must use the appropriate field types for the specific database you are using. For example, if you are using a SQL database, you must ensure that a memo field is set to *VARCHAR* etc...

Field Description:

- 8. fs_date
- The date and time that the text file transformation took place when the actual batch was run **9. doc_name**
- Document Description (The "Save As" name from the Batch screen)
- 10. doc_nbr
- Document Number extracted from the text file
- 11. addr_code
- Address Code customer code, supplier code or other code
- 12. load_type
 - The conversion type (XML or Dataset)
- **13. file_load** (field type can be: Text, XML, Memo) The field where document converted into XML is loaded
- **14. file_dataset** (field type can be: Text, XML, Memo) The field where the document converted into Dataset is loaded
- **15. file_pdf** (field type can be: Text, XML, Memo)
- The field where the document converted into PDF format is loaded **16. fs field1**
- Field from document header
- 17. fs_field2
- Field from document header **18. fs_field3**
- Field from document header
- 19. fs_field4
- Field from document header
- 20. fs_field5

Field from document header

If any other fields are need then they can be added to the table but Fraser Stream doesn't use them.

7 HTML Reporting Structure

The third key component to Fraser Stream Integration is the HTML Reporting Interface. This component creates a comprehensive series of web pages that group and categorize you newly transformed files based on your folder structure.

With <u>FSI's Batch Interface</u>, you can transform literally thousands of files daily, depositing them into any combination of file folders on your network. With FSI's HTML Interface these file folders and the contents within them can now be made available to personnel in your company through any web browser. Security issues can be addressed at the folder level ensuring only those with appropriate access can view a folder's contents.

7.1 Using the FSI HTML Reporting Interface

When building the HTML page structure, you must begin at the lowest level, creating those pages that reside at the bottom of the hierarchy. The last page to be created will be the main or home page. This rule must be followed; otherwise, any links to pages lower in the page hierarchy will fail.

The following diagram is an example of simple HTML structure that can be generated by FSI.



	Page Nume :	C:\Program Files\Fraser !	Stream\Fraser Stream Integration\Web				_
eque	nce:	20	HTML Page Background :	#C6FFC6	green_01	•	Colors
		Active Page	HTML Page Title :	Fraser Stream Inte	gration Fyellow_02 yellow_03 green 01		
File	es Pattern :	*50.html	-		green_02 green_03		
File	es Folder :	C:\Program Files\Fr	aser Stream\Fraser Stream Integration\W	eh\WebDetail	blue 01	-	
1000		-		op in opp ordin		-	
Lin	ks Path	C:\Program Files\Fr	aser Stream\Fraser Stream Integration\W	eb\WebDetail	blue_02 blue_03	~	
	ks Path Add Files S	C:\Program Files\Fr	aser Stream \Fraser Stream Integration \W Files_Folder	eb/WebDetail	inks_Path	~	
Lin	ks Path Add Files S Files_Pattern *50.html	C:\Program Files\Fr	aser Stream\Fraser Stream Integration\\\/ Files_Folder C:\Program Files\Fraser Stream\f	eb/WebDetail	inks_Path	tream	Fraser Stream I

The following interface is used to generate the HTML Reporting Interface:

7.2 FSI HTML Reporting Interface – Field Explanations

The interface is divided into two sections: Header and Detail

7.2.1 Header Section

The Header section defines the properties of the HTML page itself including the name, page path, colour, and page generation order.

- 1. HTML Page Name Input the name of the HTML page to be created. NOTE: Do not include the .html extension.
- 2. HTML Page Path

Input the path of the folder in which this HTML file will reside.

3. Sequence

Enter an integer value to specify the order sequence for this HTML file. Pages with a lower number will be created first.

4. HTML Page Background Select a background color from the drop-down list. It is advisable to utilize different colors to identify the various levels in the HTML hierarchy.

5. HTML Page Title

Enter a title to identify the types of links/files that are accessible from this page. For example, a page containing links to different production reports may be called *ABC Inc. Production Reports*

7.2.2 Detail Section

The Detail Section defines the <u>contents of each HTML page</u> such as links to other HTML pages further in the hierarchy, or, directly to transformed files i.e. XML, CSV, etc...

1. Files Pattern

Input a pattern (i.e. *.html or *.xml) or specify an exact name of the link(s) to be included on this page. NOTE: The criteria will only be applied to a single folder as specified in the next field

2. Files Folder

Specify the path of the folder that contains the file(s) to be displayed on this page

3. Links Path

Specify the path used to create the HTML links to a list of files.

7.2.3 Command Buttons

Add Selection Criteria

Use this button to add a new line in the selection criteria window. Each HTML page can have multiple selection criteria. For example, a single HTML page can display links to all XML files in folder A, all CSV files in folder A, and all HTML pages in folder B. The newly generated page will group the links by file extension as well as by path.

Update Web Files Batch

This command will save the HTML page with all the specified page definitions.

Delete Page

This command will delete the selected page.

Refreshing Pages

All web pages are automatically refreshed each time the Main_Batch.bat file is run. This file not only reruns each individual batch process, it also refreshed the HTML page structure.

The following screenshots are examples of a pages created with FSI's HTML Interface. Note the two different kinds of files that are displayed (XML and CSV).

Home page:



Production Files HTML page:

🖄 HTML OUTPUT - Microsoft I	nternet Explore	r		. dī	>
Elle Edit View Favorites Iook	s <u>H</u> elp 🛷	nd		4	ł
🌀 Back 🔹 🕥 - 💌 😰	🏠 🔎 Searc	h 👷 Favorites 🜒 Media 🤣 🍰 - 🍑 🔤 - 🧾 📷 🎼 🕼 🥸			
ddress 🙋 C:\Program Files\Fraser Si	ream/Fraser Stream	n Integration/Web/WebDetail/Production Files 50.html	🛩 🔁 Go	Links	-
Date: 8/22/2004 5:45:40	РМ				
Fraser Stream Sof	tware				
Production Fil	es				
Folder: C:\Program Files\Fr	aser Stream\	Fraser Stream Integration\Deployment\XMLXSL			
Date	File Size	File Name			
8/22/2004 4:51:38 PM	77562	Delimited by comma.csv			
8/22/2004 5:14:07 PM	15045	Delimited by semicolon.csv			
8/22/2004 5:14:11 PM	15045	Delimited by Tab.csy			
Folder: C:\Program Files\Fr	aser Stream\	Fraser Stream Integration\Deployment\XMLXSL			
Date	File Size	File Name			
8/22/2004 5:14:15 PM	41845	Efficiency by Work Center Report Example 0010.xml			
8/22/2004 5:14:23 PM	31843	Invoice History Report Example 0008.xml			
8/22/2004 5:14:26 PM	52891	Purchase Orders by Item Report Example 0009.xml			
8/22/2004 5:14:58 PM	52891	Purchase Orders by Item Report Example 0009 2004 8 22 17 14 55 468.xml			
8/22/2004 5:44:44 PM	52891	Purchase Orders by Item Report Example 0009 2004 8 22 17 44 41 828.xml			
8/22/2004 5:15:01 PM	43594	Sales Orders Report 0011.xml			
8/22/2004 5:14:33 PM	32150	Test Report 0001 Example 01.xml			
8/22/2004 5:14:36 PM	48093	Test Report 0002 Example 02.xml			
8/22/2004 5:14:40 PM	39014	Test Report 0003 Example 03.xml			
8/22/2004 5:14:43 PM	40771	Test Report 0004 Example 04.xml			
8/22/2004 5:14:46 PM	72361	Test Report 0005 Example 05.xml			
8/22/2004 5:14:19 PM	558107	Transaction Register Example 0006.xml			
8/22/2004 5:15:02 PM	264999	WO Dataset.xml			
0			My Computer		

8 FSI Batches and Commands

The following section covers the logic behind the batch files that drive Fraser Stream Integration. These files may be created and modified directly through the code by the developer to create customized processes if FSI's Windows Service does not suffice. It is crucial that the command structure be followed exactly to ensure that all steps in the transformation process succeed when executed.

8.1 Main Batch File

The entire FSI batch structure revolves around two kinds of batch files, the Main batch file and the Individual batch files. Each time a new individual batch file is generated the command to run it is automatically appended to a Main batch file. This file when run, will initiate every single individual batch file.

Note: The Individual Batch file essentially contains the series of commands which instructs FSI where to select the input file from, how to transform, which file output to use, and where to deposit the outputted file.

The Main Batch file Main_Batch.bat is stored by default in:

C:\Program Files\Fraser Stream\Fraser Stream Integration\Batch\

The following two commands are contained in the Main_Batch.bat:

```
CALL ".\bin\FS_CreateBatch.exe" "Batch_Production"
CALL ".\Batch\BatchDetail\Batch_Production.bat"
```

The first command "FS_CreateBatch.exe" creates the individual batch "Batch_Production.bat" in the Batch\BatchDetail folder.

The second command executes this newly generated individual batch file.

8.2 Individual Batch File

The Individual Batch file essentially contains the series of commands which instructs FSI where to select the input file from, how to transform, which file output to use, and where to deposit the outputted file.

The Individual Batch file is stored by default in:

C:\Program Files\Fraser Stream\Fraser Stream Integration\Batch\BatchDetail\

Example of an individual FSI batch file:

cd "C:\Program Files\Fraser Stream\Fraser Stream Integration"

CALL ".\bin\FS_Runtime.exe" "XMLXSL" ".\Deployment\TextReports\Work Order Status Report 0007.prn" ".\SetFiles\Settings_XML.xml" ".\Deployment\XML" "" "" ""

CALL ".\bin\FS_Runtime.exe" "XMLXSL" ".\Deployment\TextReports\Invoice History Report 0008.prn" ".\SetFiles\Settings_XML.xml" ".\Deployment\XML" "Example 0008" "" ""

CALL ".\bin\FS_Runtime.exe" "XMLXSL" ".\Deployment\TextReports\Purchase Orders Report 0009.prn" ".\SetFiles\Settings_XML.xml" ".\Deployment\XML" "" "Extend" ""

CALL ".\bin\FS_Runtime.exe" "XML" ".\Deployment\TextReports\Sales Orders Report 0011.prn" ".\SetFiles\Settings_XML.xml" ".\Deployment\XML" "" "Input_File" ""

CALL ".\bin\FS_Runtime.exe" "DATASET" ".\Deployment\TextReports\Work Order Status Report 0007.prn" ".\SetFiles\Settings_XML.xml" ".\Deployment\XML" "" "Save_As" "WO Dataset.xml"

The individual batch file is an executable that contains all necessary parameters for selecting, transforming and depositing the text report or delimited file. The FS_Runtime.exe command structure occurs as follows:

"FS_Runtime.exe" "Output type" "Input file" "Settings file" "Output folder" "Setting used" "Output Options" "Save As File"

8.2.1 FS_Runtime.exe Parameters Description

1. FS_Runtime.exe

This executable file is the command transforms the text report. If Fraser Stream Integration has been registered on a Windows server then the command can be used in a batch or can be integrated into any other application running on that server.

2. Output type

This parameter defines the type of output created by the command. The output can be CSV, XML, XML with DTD, XML with Style Sheet and ADO Data Set in XML format.

3. Input file

This is an ASCII text report created by a UNIX or Windows application. Fraser Stream Integration needs read access to this file. The file can be anywhere on the server or on the network where Fraser Stream Integration has reading rights.

4. Settings file

This is a file created by Fraser Stream Reports that contains the transformation rules for a list of text reports. The file name is Settings_XML.xml; Fraser Stream Integration needs read access to this file. The file can be anywhere on the server or on the network where Fraser Stream Integration has reading rights.

5. Output folder

The output file created by the FS_Runtime command will be stored in this folder. The folder can be anywhere on the server or on the network where Fraser Stream Integration has writing rights.

6. Setting used (optional parameter)

If this parameter is not specified then Fraser Stream Integration will use the first set of transformation settings that is valid for the input file. If the transformation has to be done with a specific set of transformation settings then this parameter will identify that particular set of transformation settings to be used. This parameter is "My title" value chosen by the user when the transformation settings are generated by Fraser Stream Reports.

7. Output Options (optional parameters)

Extend

If the parameter "Extend" is specified then the file name will be extended with the date and the time when the output file is generated (the time format include the milliseconds). The extended file name is useful when the user needs to generate unique file names for the output files. Example of output file name: Test Report 0001 Example 01_2004_3_4_20_29_25_750.csv

Save_As

if the parameter "Save_As is specified then the output file name will be the same as the file name from the "Save_As_File" parameter.

Input_File

If this parameter is used then the output file name is the same as the input file but the file extension is changed according with the output file type.

8. Save_As_File (optional Parameter)

If the "Save_As" option is used in the previous parameter (Save_As under the Output Options) then this parameter specifies the output file name.

8.3 Database Output Method 1 – Extract the data from a text report and load it into a table

The following section describes the command structure for outputting data using Method 1 where a new table is generated containing the contents of the text report record by record.

The Main Batch file TABLE_Batch.bat is stored by default in:

C:\Program Files\Fraser Stream\Fraser Stream Integration\Batch\

The table output batch is an executable that contains all necessary parameters for selecting, transforming, creating and importing the text report or delimited file into a database table. The **FS_Runtime_DB_TB.exe** command structure occurs as follows:

"FS_Runtime_DB_TB.exe" "Output type" "Input file" "Settings file" "Output folder" "Setting used" "Output Options" "Save As File" "Connection"

Each time the **TABLE_Batch.bat** is run, FSI creates a table in the specified database and imports the source data into it. If the table already exists, it will be deleted and replaced with the newly created one.

The command has the all parameters from standard individual batch file FS_Runtime.exe plus one additional parameter - "Connection".

"Connection" parameter

The "Connection" parameter is the connection code that specifies which database connection string is to be used by the command. The connection codes are created in the "Database Connection" menu.

8.4 Database Output Method 2– Convert a text report into XML, CSV or Dataset and load it into a table record

The following section describes the command structure for outputting data using Method 2 where a new record is generated in an existing table and the contents of the text report converted into XML, CSV or Dataset is loaded into a field.

The Main Batch file **DB_Batch.bat** is stored by default in:

C:\Program Files\Fraser Stream\Fraser Stream Integration\Batch\

The Database output batch is an executable that contains all necessary parameters for selecting, transforming, creating and importing the text report or delimited file into a record within an existing database table. The **FS_Runtime_DB.exe** command structure occurs as follows:

"FS_Runtime_DB.exe" "Output type" "Input file" "Settings file" "Output folder" "Setting used" "Output Options" "Save As File" "Connection"

Each time the **DB_Batch.bat** is run, FSI creates a record into a table called "**filelist**" within the database specified in the connection string. The original text report or delimited file is stored in it's entirety within a single field in the record. The content of the output file whether it be XML, CSV, or Dataset is stored in a completely separate field within that record.

The command has the all parameters from standard individual batch file FS_Runtime.exe plus one additional parameter - "Connection".

"Connection" parameter

The "Connection" parameter is the connection code that specifies which database connection string is to be used by the command. The connection codes are created in the "Database Connection" menu.

8.5 Database Output Method 3– Extract documents from a text report in XML or Dataset format and load each document into a table record.

The following section describes the command structure for outputting data using Method 3 where a new record is generated into an existing table for each document extracted from a text report and loads it into that record in XML or Dataset format.

The Main Batch file **DB_Batch.bat** is stored by default in:

C:\Program Files\Fraser Stream\Fraser Stream Integration\Batch\

The Database output batch is an executable that contains all necessary parameters for selecting, transforming, creating and importing the text report or delimited file into a record within an existing database table. The **FS_Runtime_DB.exe** command structure occurs as follows:

"FS_Runtime_DB.exe" "Output type" "Input file" "Settings file" "Output folder" "Setting used" "Output Options" "Save As File" "Connection"

Each time the **DB_Batch.bat** is run, FSI creates a record for each document into a table called "**fsidoc**" within the database specified in the connection string. Each document is loaded into a table record along with other fields from the document header.

"Connection" parameter

The "Connection" parameter is the connection code that specifies which database connection string is to be used by the command. The connection codes are created in the "Database Connection" menu.

8.6 Web HTML Output structure

The main **Web_Files.bat** batch file creates the HTML Reporting Interface. This batch allows you to in a matter of minutes, create a comprehensive series of web pages that group and categorize you newly transformed files based on your folder structure. The batch always runs when the Main_Batch.bat is executed.

The Web_Batch.bat file is stored by default in:

C:\Program Files\Fraser Stream\Fraser Stream Integration\Batch

Web_Files.bat contains the command "FS_FilesList.exe" that genereates the HTML pages:

CALL ".\bin\FS_FilesList.exe" "Production Files 50" ".\Web\WebDetail\Production Files 50.html"

Example of Web_Files.bat :

REM Fraser Stream Integration - Web Batch: Web_Files.bat cd "C:\Program Files\Fraser Stream\Fraser Stream Integration" CALL ".\bin\FS_FilesList.exe" "Production Files 50" ".\Web\WebDetail\Production Files 50.html" CALL ".\bin\FS_FilesList.exe" "CSV Files 50" ".\Web\WebDetail\CSV Files 50.html" CALL ".\bin\FS_FilesList.exe" "Datasets Files 50" ".\Web\WebDetail\Datasets Files 50.html" CALL ".\bin\FS_FilesList.exe" "Fraser Stream Integration Files" ".\Web\Fraser Stream Integration Files.html"

8. PDF Output

Fraser Stream Integration converts text reports into PDF format. Initially the PDF report must be designed using Crystal Reports and the structure saved as an *.RPT* file. Once the structure has been created, Fraser Stream Integration inserts the contents of any text report into the predefined report structure, and then outputs the results into PDF format.

The whole process can be automated ensuring the user need only output the original text file; Fraser Stream software will do the rest, opening the PDF automatically on the user's PC.

8.7 Preconditions

The following preconditions must be met to generate PDF reports with Fraser Stream Integration:

- 1. Transformation Settings must be created for the text report (Please refer to the Fraser Stream Report Mapping Documentation)
- 2. Within Fraser Stream Integration, the path to the Shared Folder has to be set

Options			_ 🗆 🔀
Close	Options		
Regional Options: Decimal Separator: Date Separator: CSV Options: CSV Separator: Apostrophe for leading	dot 7 slash		
zeroes (yes/no): Create DTD for XML file: Generate Activity Log	no yes	Number of Rows in Datagrid Encoding iso8859-2 Real Time Processing	200
Runtime XSLT Path: C:\Program Files\Fraser Stream Shared Folder	u\Fraser Stream	Integration\XMLOutput\XSLT	
F:\&pp_Files\FSI_Shared			Save

- 3. Click the PDF Button on the Main Screen. If the Crystal Report is not found then a message is displayed.
- 4. A Dataset file having the report name is created into :
- C:\Program Files\Fraser Stream\Fraser Stream Integration\Dataset
- 5. Verify and change the database fields types using "Change Dataset Fields Types" menu and Click the PDF Button again to re-generate the dataset

- 6. Copy the generated dataset into a working folder
- 7. Create a Crystal Report having the Dataset as data source
- 8. Save the Crystal Report with the same name as the Dataset name but with the extension .rpt
- 9. Copy the Crystal Report into [Shared Folder]\CrystalReports folder
- 10. Click the PDF Button to generate the PDF format

8.7.1 Change Dataset Fields Types

Fraser Stream Integration applies Dataset fields types according with the type of data extracted from the text file. For example if the Sales Order numbers in a given text report are numeric (Example: 12345) then the Dataset field type for the Sales Order field will be automatically set to numeric. If however the same text report is generated for a different division or customer where Sales Order numbers are strings (Example: SO1234) then Crystal Reports will run into trouble. To avoid data type conflicts in Crystal Reports, ensure that fields such as Sales Order are defined as strings with the Dataset file.

Using Fraser Stream Integration, select Tools\Change Dataset Fields Types from the menu to predefine the field types for a given text report.

Select a dataset from the output folder. The current Dataset field types are displayed and the field type can be changed overwriting the field type. Save the new fields types.

	usiyp	62		
SelectDataSet				
C\ESI\CrystalDatasets\DataSet_Sales Ord	ers by Order	Report Example 0	011 xml	
	0.0 2, 01401	. topon Example o	0113011	
DataSet Fields Types List:				
String		Field	Туре	
Sung	•	Header_ID	Integer	
Integer		Sales_Order	String	
Date		Sold_To	String	
Date		Sold_To_Name	String	
Double		Ship_To	String	
		Order_Date	Date	
		Status	String	
		Quote	String	
		Credit_Terms	String	
		Currency	String	
		Detail_ID	Integer	
		Ln	Integer	
		Item_Number	String	
		Item_Descriptio	String	
		UM	String	
		Qty_Ordered	Double	
		Qty_Allocated	Double	
		Qty_Picked	Double	
		Qty Open	Double	

Once the dataset field types have been properly defined, the Dataset file can be opened with Crystal Reports to create the report structure for the final PDF output.

8.8 Create Crystal Reports Format

Please use Crystal Reports documentation that shows how to design the report. We cover here few basic steps about how to create the report from a Dataset source.

1. Choose the data source type. Select ADO.NET (XML) option



2. Choose the data source Dataset file:

ADO.NET (XML)		<u>></u>
Connection Please enter connection informa	lion	
<u>F</u> ile Path :	C:\FSI\Cryst	alDatasets\DataSet_Sales Orders by
Use Classes from Project		
<u>C</u> lass Name:		
Use DataSet from Class:		
< <u>B</u> ack <u>N</u> ext>	Fin	ish Cancel Help
Standard Report Creation Wizard		
Data Choose the data you want to report on.		ب [_]
Available Data Sources:		Selected Tables:
Current Connections	^	WewDataSet
History		E FooterTable
Create New Connection		I HeaderTable
Access/Excel (DAO)	>>	
Make New Connection		
NewDataSet	<	
E FooterTable	<<	
HeaderTable		
Database Files Exchange 5.5 Message Tracking Log	,	
🗄 🧰 Exchange Message Tracking Log	, 	
Legacy Exchange Mailbox Admin		
ODBC (RD0)		
H Olap	~	

3. Link Dataset Tables – HeaderTable, DetailTable and FooterTable

Next> Finish Cancel Help

< Back

Lini	k together the tables you added to	the report.	
	Header Table Header ID Sales_Order Sold_To Sold_To_Name	DetailTable	Auto-Arrange Auto-Link By Name By Key Link
		Footer_ID Total_Extended_Price	Order Links
			Link Options

4. Select the fields from Dataset Tables



5. Crystal Report Design Example:



Ensure the newly created Crystal Report file name is the same as the Dataset file name but with rpt extension. Example: If the Dataset file name is DataSet_Purchase Orders Report 0009.xml then Crystal Reports report name will be DataSet_Purchase Orders Report 0009.rpt

Copy the newly generated Crystal Reports file into the Crystal Reports folder in the Shared Folder



PDF Format Example:

8.9 Batch Settings for PDF Output

Once the transformation settings have been mapped Fraser Stream Integration can be configured for PDF output.

	Desktop_user1	-		Create / Updat	e Batch	Delete Bate	ch
ut Foldor -	C\FSI\Inbox\Home	e\user1					
tout Foldor :	C\FSI\Outbox\Home\user1						
CAProgram Eiled Ergeor Stream Ergeor			troom Integration Sof	ilos/Sottings VMI vml			
ttings XML :	our rogrammest	Taser Stealing Taser S	seean integration(See	nestoemigs_vivic.viii			
chive Folder:	C.(FSI(InDox(Home	eluseri parchive					_
Active Batch	Delete I	nput Files	Database Table	Output Connection	:		•
quence : 10	Run As	Service	Database Reco	rd Output Connectio	n		-
Output Type :							
PDF	•	0.1.1.7	Inc. 171	Online Hard	0.1.1.0.1	One As Ells	
InnutFile ·		XMLXSI	input_rile	Setting_Used	Output_Options	Save_As_File	
ndf ord		CSV	excel".pm				
poir .prig		PDF	PO*.pm		Input_File		
Setting Used :		PDF	pdf*.pm				
-							
	к.						
Output Options							
Output Option:	•						
Output Options							
Output Options	Table						

Once the original Dataset file is generated, the Dataset Field Types must be defined to ensure the report is properly compile by Crystal Reports.