

# **TIBCO® Adapter for Files (IBM i)**

## **Installation and User's Guide**

*Software Release 5.0*  
*January 2016*

## Important Information

SOME TIBCO SOFTWARE EMBEDS OR BUNDLES OTHER TIBCO SOFTWARE. USE OF SUCH EMBEDDED OR BUNDLED TIBCO SOFTWARE IS SOLELY TO ENABLE THE FUNCTIONALITY (OR PROVIDE LIMITED ADD-ON FUNCTIONALITY) OF THE LICENSED TIBCO SOFTWARE. THE EMBEDDED OR BUNDLED SOFTWARE IS NOT LICENSED TO BE USED OR ACCESSED BY ANY OTHER TIBCO SOFTWARE OR FOR ANY OTHER PURPOSE.

USE OF TIBCO SOFTWARE AND THIS DOCUMENT IS SUBJECT TO THE TERMS AND CONDITIONS OF A LICENSE AGREEMENT FOUND IN EITHER A SEPARATELY EXECUTED SOFTWARE LICENSE AGREEMENT, OR, IF THERE IS NO SUCH SEPARATE AGREEMENT, THE CLICKWRAP END USER LICENSE AGREEMENT WHICH IS DISPLAYED DURING DOWNLOAD OR INSTALLATION OF THE SOFTWARE (AND WHICH IS DUPLICATED IN THE LICENSE FILE) OR IF THERE IS NO SUCH SOFTWARE LICENSE AGREEMENT OR CLICKWRAP END USER LICENSE AGREEMENT, THE LICENSE(S) LOCATED IN THE "LICENSE" FILE(S) OF THE SOFTWARE. USE OF THIS DOCUMENT IS SUBJECT TO THOSE TERMS AND CONDITIONS, AND YOUR USE HEREOF SHALL CONSTITUTE ACCEPTANCE OF AND AN AGREEMENT TO BE BOUND BY THE SAME.

This document contains confidential information that is subject to U.S. and international copyright laws and treaties. No part of this document may be reproduced in any form without the written authorization of TIBCO Software Inc.

TIBCO, Two-Second Advantage, TIBCO Adapter, TIBCO Adapter for Files (IBM i), Predictive Business, Information Bus, TIBCO Enterprise Message Service, TIBCO Rendezvous, TIBCO SmartSocket, TIBCO ActiveMatrix BusinessWorks, TIBCO Hawk, TIBCO InConcert, and TIBCO Enterprise either registered trademarks or trademarks of TIBCO Software Inc. in the United States and/or other countries.

Enterprise Java Beans (EJB), Java Platform Enterprise Edition (Java EE), Java 2 Platform Enterprise Edition (J2EE), and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle Corporation in the U.S. and other countries.

All other product and company names and marks mentioned in this document are the property of their respective owners and are mentioned for identification purposes only.

THIS SOFTWARE MAY BE AVAILABLE ON MULTIPLE OPERATING SYSTEMS. HOWEVER, NOT ALL OPERATING SYSTEM PLATFORMS FOR A SPECIFIC SOFTWARE VERSION ARE RELEASED AT THE SAME TIME. SEE THE README FILE FOR THE AVAILABILITY OF THIS SOFTWARE VERSION ON A SPECIFIC OPERATING SYSTEM PLATFORM.

THIS DOCUMENT IS PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT.

THIS DOCUMENT COULD INCLUDE TECHNICAL INACCURACIES OR TYPOGRAPHICAL ERRORS. CHANGES ARE PERIODICALLY ADDED TO THE INFORMATION HEREIN; THESE CHANGES BE INCORPORATED IN NEW EDITIONS OF THIS DOCUMENT. TIBCO SOFTWARE INC. MAY MAKE IMPROVEMENTS AND/OR CHANGES IN THE PRODUCT(S) AND/OR THE PROGRAM(S) DESCRIBED IN THIS DOCUMENT AT ANY TIME.

THE CONTENTS OF THIS DOCUMENT MAY BE MODIFIED AND/OR QUALIFIED, DIRECTLY OR INDIRECTLY, BY OTHER DOCUMENTATION WHICH ACCOMPANIES THIS SOFTWARE, INCLUDING BUT NOT LIMITED TO ANY RELEASE NOTES AND "READ ME" FILES.

Copyright © 2003-2016 TIBCO Software Inc. All rights reserved.

TIBCO Software Inc. Confidential Information

# Contents

<b>Figures</b> .....	<b>vii</b>
<b>Preface</b> .....	<b>ix</b>
Related Documentation .....	x
TIBCO Adapter for Files (IBM i) Documentation .....	x
Other TIBCO Product Documentation .....	x
Typographical Conventions .....	xi
Connecting with TIBCO Resources .....	xiii
How to Join TIBCOCommunity .....	xiii
How to Access TIBCO Documentation .....	xiii
How to Contact TIBCO Support .....	xiii
<b>Chapter 1 Overview</b> .....	<b>1</b>
Architecture .....	2
Transport Mechanisms and Delivery Options .....	3
Wire Format .....	3
Functional Components .....	4
FilePublisher .....	5
FileSubscriber .....	8
Message Delivery Considerations .....	10
Data Transfer Modes .....	10
Transport Options .....	10
Comparing ECM and RVCM for Guaranteed Delivery .....	11
Supported File Types and Data Types .....	13
BusinessEvents™ Messages .....	15
JMS Overview .....	16
Point-to-Point Messaging .....	16
Publish and Subscribe Messaging .....	17
Controlling the Flow of Messages .....	18
JMS Message Structure .....	19
<b>Chapter 2 Installation</b> .....	<b>21</b>
Installation Requirements .....	22
System Requirements .....	22
Storage Requirements .....	22

Software Requirements . . . . .	22
Distribution Media and Contents . . . . .	23
Obtaining the Software . . . . .	23
Installation Files . . . . .	23
Uploading the Software . . . . .	24
Previous Releases of the Software . . . . .	26
Identifying Previous Installations . . . . .	26
Deleting Previous Installations . . . . .	27
Installing the Software . . . . .	28
Single Installation . . . . .	28
Multiple Installations . . . . .	29
Testing Adapter . . . . .	31
Sample File Contents . . . . .	31
Submitting the Initial Sample Using RV . . . . .	32
Submitting the Initial Sample Using EMS . . . . .	35
Sample Results . . . . .	37
Stopping Adapter . . . . .	38
Checking the Trace Logs . . . . .	38
Messages to Operator . . . . .	39
Uninstalling the Software . . . . .	40
<b>Chapter 3 Configuring Adapter . . . . .</b>	<b>41</b>
Overview . . . . .	42
Configuration File Structure . . . . .	42
Configuration Syntax . . . . .	43
Sample Configuration and Data Files . . . . .	44
FilePublisher Configuration . . . . .	47
Trace Section . . . . .	47
Options Section . . . . .	51
Pre-Register Section . . . . .	56
FileType Section . . . . .	57
Sample Configuration Elements . . . . .	70
FileSubscriber Configuration . . . . .	79
Trace Section . . . . .	79
Options Section . . . . .	82
FileType Section . . . . .	92
Sample Configuration Elements . . . . .	106
Guaranteed Delivery for EMS Messages . . . . .	113
<b>Chapter 4 Using Adapter . . . . .</b>	<b>115</b>
Starting and Stopping Adapter . . . . .	116

Starting FilePublisher . . . . .	116
Starting FileSubscriber . . . . .	117
Stopping FilePublisher . . . . .	118
Stopping FileSubscriber . . . . .	119
Block Transfer Mode . . . . .	120
File Transfer Using ECM . . . . .	121
ECM Configuration . . . . .	122
Sending and Receiving Numeric Data Types . . . . .	124
Pre-Processing and Post-Processing Files . . . . .	125
Pre-Processing Files . . . . .	125
Post-Processing Files . . . . .	126
Using Heartbeat Messages . . . . .	127
Working with Batch Messages . . . . .	128
Sending Trigger Messages . . . . .	129
FilePublisher Usage Guidelines . . . . .	130
Sending Numeric Data Types . . . . .	130
Sending Data Untranslated (OPAQUE) . . . . .	131
Constructing a Subject Name from Data . . . . .	131
Publishing Double Values . . . . .	132
Pre-registering Subscribers . . . . .	132
FileSubscriber Usage Guidelines . . . . .	133
Receiving Numeric Data Types . . . . .	133
Adding Header and Trailer Records . . . . .	133
<b>Appendix A Trace Log Examples . . . . .</b>	<b>137</b>
Successful FilePublisher Session . . . . .	138
Failed FilePublisher Session . . . . .	142
Successful FileSubscriber Session . . . . .	145
Failed FileSubscriber Session . . . . .	151
<b>Appendix B Error Messages . . . . .</b>	<b>159</b>
Error Message Format . . . . .	160
Publisher Error Messages . . . . .	161
SXF0000 – SXF0500 . . . . .	161
SXF1000 – SXF1999 . . . . .	178
SXF2000 – SXF2999 . . . . .	186
SXF3000 – SXF3999 . . . . .	211
SXF4000 – SXF4999 . . . . .	214
Subscriber Error Messages . . . . .	216
SXF5000 – SXF5999 . . . . .	216

SXF6000 – SXF6999.....	229
SXF7000 – SXF7999.....	234
SXF8000 – SXF8999.....	263
SXF9000 – SXF9999.....	274
<b>Index .....</b>	<b>277</b>

# Figures

Figure 1 Logical Architecture .....2

Figure 2 Functional Components .....4

Figure 3 FilePublisher Workflow When Polling.....6

Figure 4 Using a Trigger Subject With a File Name .....7

Figure 5 FileSubscriber Workflow.....8

Figure 7 Message Delivery.....16

Figure 8 Point-to-point Message .....17

Figure 9 Publish and subscribe messages.....18

Figure 10 Sections of a FilePublisher Configuration File .....42

Figure 11 Sections of a FileSubscriber Configuration File .....43





# Preface

This manual describes how to install, configure, and use the TIBCO® Adapter for Files (IBM i) software. This manual is primarily intended for users who are involved in setting up and maintaining the system, but may also be useful for application programmers.

## Topics

---

- [Related Documentation, page x](#)
- [Typographical Conventions, page xi](#)
- [Connecting with TIBCO Resources, page xiii](#)

## Related Documentation

---

This section lists documentation resources you may find useful.

### TIBCO Adapter for Files (IBM i) Documentation

The following documents form the TIBCO® Adapter for Files (IBM i) documentation set:

- *TIBCO Adapter for Files (IBM i) Installation and User's Guide*— Read this document for instructions on using the product.
- *TIBCO Adapter for Files (IBM i) Release Notes* — Read this document for a summary of features, changes since the last release, and a description of any issues that may affect installing or using the Adapter.

### Other TIBCO Product Documentation

You may find it useful to read the documentation for the following TIBCO products:

- TIBCO Rendezvous®: This TIBCO's real-time transport layer is supported by the TIBCO Adapter for Files (IBM i) software.
- TIBCO Enterprise Message Service™: This product offers uniform messaging interface among enterprise applications and is supported by the TIBCO Adapter for Files (IBM i) software.
- TIBCO® Adapter for Files z/OS (MVS): This product allows files on an IBM z/OS system to interoperate with applications configured for the TIBCO environment.

# Typographical Conventions

The following typographical conventions are used in this manual.

Table 1 General Typographical Conventions




Convention	Use
code font	Code font identifies commands, code examples, filenames, pathnames, and output displayed in a command window. For example:  Use <code>MyCommand</code> to start the foo process.
<b>bold code font</b>	Bold code font is used in the following ways: <ul style="list-style-type: none"> <li>In procedures, to indicate what a user types. For example: Type <b>admin</b>.</li> <li>In large code samples, to indicate the parts of the sample that are of particular interest.</li> <li>In command syntax, to indicate the default parameter for a command. For example, if no parameter is specified, <code>MyCommand</code> is enabled: <code>MyCommand [<b>enable</b>   disable]</code></li> </ul>
<i>italic font</i>	Italic font is used in the following ways: <ul style="list-style-type: none"> <li>To indicate a document title. For example: See <i>TIBCO ActiveMatrix BusinessWorks Concepts</i>.</li> <li>To introduce new terms For example: A portal page may contain several portlets. <i>Portlets</i> are mini-applications that run in a portal.</li> <li>To indicate a variable in a command or code syntax that you must replace. For example: <code>MyCommand PathName</code></li> </ul>
Key combinations	Key name separated by a plus sign indicate keys pressed simultaneously. For example: <code>Ctrl+C</code> .  Key names separated by a comma and space indicate keys pressed one after the other. For example: <code>Esc, Ctrl+Q</code> .
	The note icon indicates information that is of special interest or importance, for example, an additional action required only in certain circumstances.
	The tip icon indicates an idea that could be useful, for example, a way to apply the information provided in the current section to achieve a specific result.

Table 1 General Typographical Conventions (Cont'd)

Convention	Use
	The warning icon indicates the potential for a damaging situation, for example, data loss or corruption if certain steps are taken or not taken.

## Connecting with TIBCO Resources

---

### How to Join TIBCOCommunity

TIBCOCommunity is an online destination for TIBCO customers, partners, and resident experts. It is a place to share and access the collective experience of the TIBCO community. TIBCOCommunity offers forums, blogs, and access to a variety of resources. To register, go to <https://community.tibco.com>.

### How to Access TIBCO Documentation

Documentation for this and other TIBCO products is available on the TIBCO Documentation site:

<https://docs.tibco.com>

Documentation on the TIBCO Documentation site is updated more frequently than any documentation that might be included with the product. To ensure that you are accessing the latest available help topics, please visit us at <https://docs.tibco.com>.

### How to Contact TIBCO Support

For comments or problems with this manual or the software it addresses, contact TIBCO Support as follows:

- For an overview of TIBCO Support, and information about getting started with TIBCO Support, visit this site:

<http://www.tibco.com/services/support>

- If you already have a valid maintenance or support contract, visit this site:

<https://support.tibco.com>

Entry to this site requires a user name and password. If you do not have a user name, you can request one.



## Chapter 1 **Overview**

This chapter introduces TIBCO Adapter for Files (IBM i) by providing background information about features, components, and the application architecture.

### Topics •

---

- [Architecture, page 2](#)
- [Functional Components, page 4](#)
- [Message Delivery Considerations, page 10](#)
- [Supported File Types and Data Types, page 13](#)
- [BusinessEvents™ Messages, page 15](#)

## Architecture

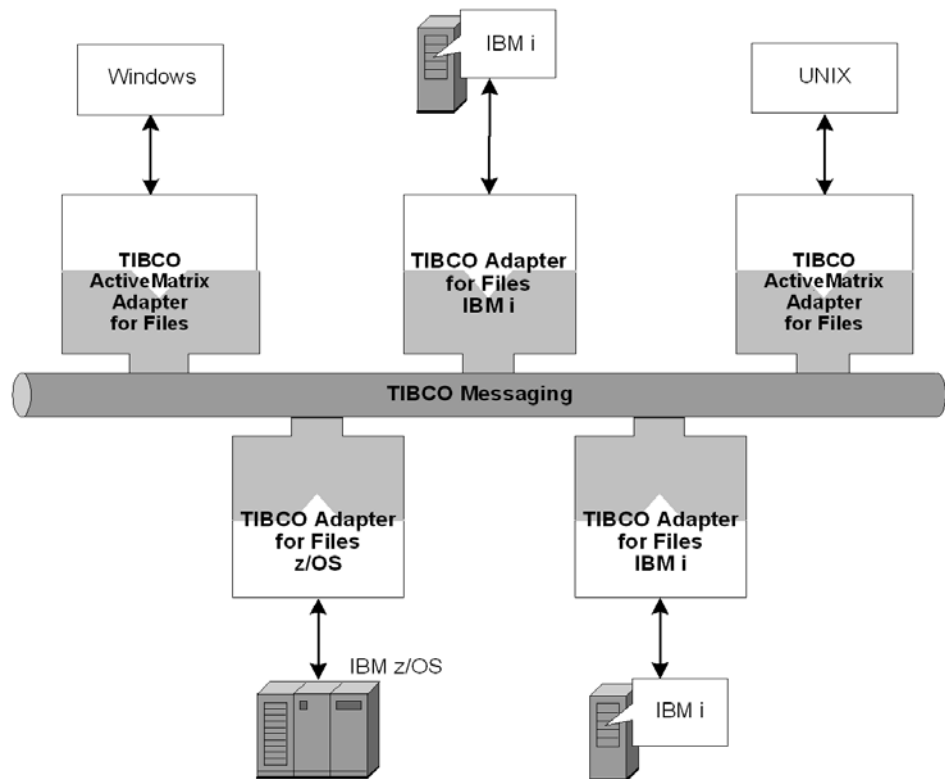
---

TIBCO Adapter for Files (IBM i) is a file handler that supports data files and communicates with other applications enabled for TIBCO Rendezvous or TIBCO EMS.

TIBCO Adapter for Files (IBM i) is typically used in a network operations center or development environment where system administrators, network administrators, and IBM i programmers have to process files between IBM i applications and other operating systems.

Typically, the adapter transfers files from IBM i systems to applications that are located on other operating systems as well as IBM i systems (Figure 1). The adapter is especially useful in batch-oriented handling of files (for example, passing automated input or output files to a billing system) or for doing quick TIBCO Rendezvous or TIBCO EMS message prototypes.

*Figure 1 Logical Architecture*





Users with substantial experience in TIBCO messaging formats can use the adapter to communicate with TIBCO BusinessWorks™. If any transformations are required across platforms, such as EBCDIC to ASCII, or Big Endian to Little Endian, conversions are handled by TIBCO Rendezvous. There are options to add constant fields to a message, and to write constant fields in the output file.



The following import and export subjects are used by ECM for handshakes between Publisher and Subscriber.

- `_TIBCO_AE_ADAPTER_FAFI_SUBTOPUB.<endpoint subject name>`
- `_TIBCO_AE_ADAPTER_FAFI_PUBTOSUB.<endpoint subject name>`

## Transport Mechanisms and Delivery Options

TIBCO Adapter for Files (IBM i) supports the following transport mechanisms:

- TIBCO EMS
- TIBCO Rendezvous Messaging (RV)
- TIBCO Rendezvous Certified Messaging (RVCM)

Users of RVCM have certified delivery with the overhead of ledger files. For simple file transfer, the adapter supports ECM (Explicit Confirmation Mode), which supplies FTP-like functionality with error handling and recovery. For additional information, see [Message Delivery Considerations on page 10](#).

## Wire Format

TIBCO Adapter for Files (IBM i) supports only TIBCO Rendezvous wire format, not AE wire format. Support is for the MapMessage message type only, the sole exception being that nested messages in EMS are nested inside StreamMessages.

## Functional Components

The adapter consists of the following two applications, as illustrated in [Figure 2](#).

- **FilePublisher**

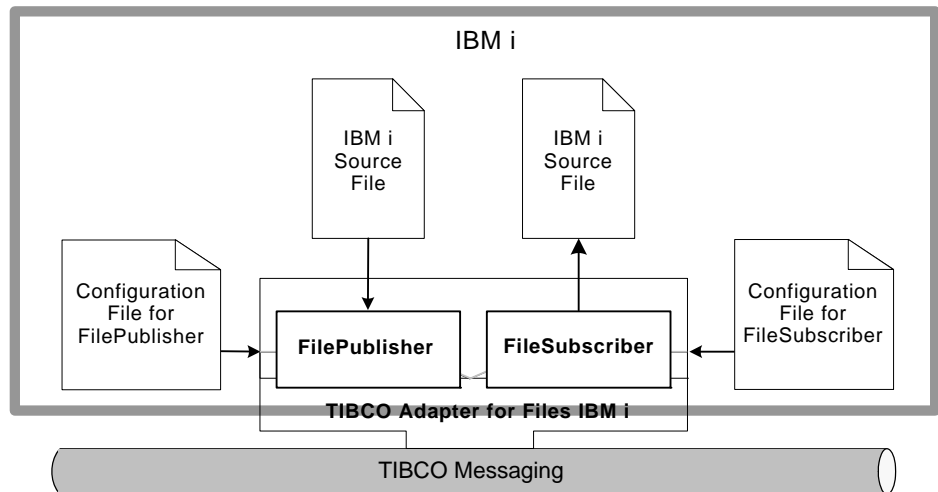
A publication service that processes data from files and publishes the contents as TIBCO Rendezvous or TIBCO EMS messages.

- **FileSubscriber**

A subscription service that subscribes to TIBCO Rendezvous or TIBCO EMS messages, processes received messages, and writes the contents to a file.

For each adapter instance, you can configure the format of the incoming or outgoing TIBCO Rendezvous or TIBCO EMS message, the file format, and many file-handling options.

*Figure 2 Functional Components*



## FilePublisher

FilePublisher is an adapter publication service, you can use it by specifying the following options:

**Data Transfer Mode:** determines whether FilePublisher transfers data in Block mode or Record mode. Each of these modes in turn has a guaranteed mode and a reliable mode. For guaranteed delivery, you can use ECM with the RV message transport, and you can use durable topics for the EMS transport. For details, see [Data Transfer Modes on page 10](#).

**File Processing Options:** determines the service can run once, or can be configured to run continuously. When configured for the latter, it processes and publishes files when it receives an event such as a timer event (polling), or an event generated by a message (trigger).

**Transport Options:** supports TIBCO Rendezvous, TIBCO Rendezvous Certified Messaging (RVCN), and TIBCO EMS message transport.

**Publication Options:** in Record mode, after processing a file, the service provides options to publish the objects generated during the file processing. These options include capabilities for checkpoint restart, to batch objects for publication, delay publication, invoke a callout before publication, or to publish an object only if all of its associations have no errors.

### Continuous File Processing

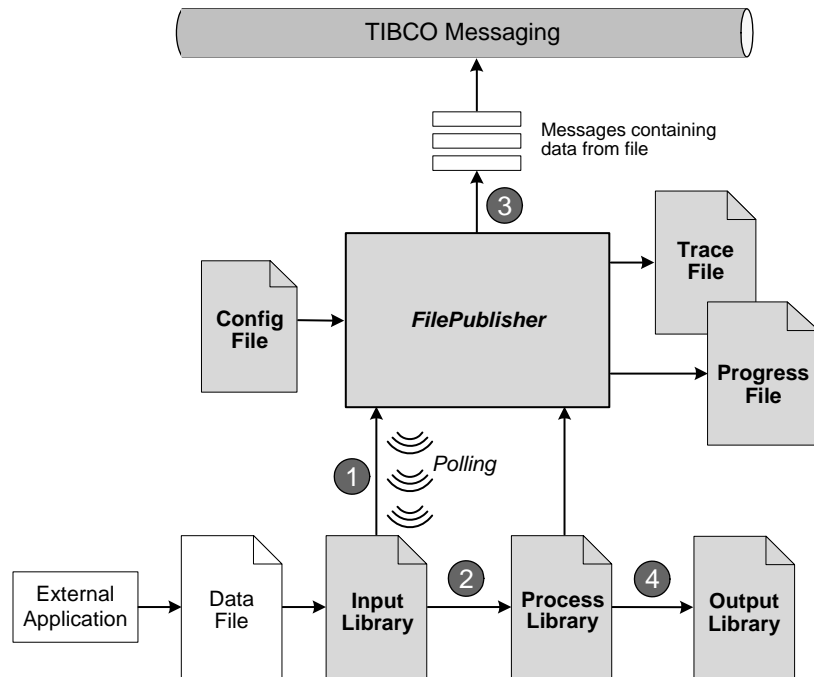
FilePublisher accesses input files either by polling for them or through use of a trigger subject. The polling and trigger-subject methods can operate together. If trigger messages are received while files are being published, the adapter queues files and publishes what is in the queue after the current file. A trigger-subject file takes precedence over a polling-method file.

## Polling Method

In the polling method, FilePublisher performs the following operations:

1. Selects files from an input library.
2. Moves files to a process library.
3. Publishes TIBCO Rendezvous and TIBCO EMS messages from data in the files.
4. Moves files to an output library or removes the files completely.

Figure 3 FilePublisher Workflow When Polling

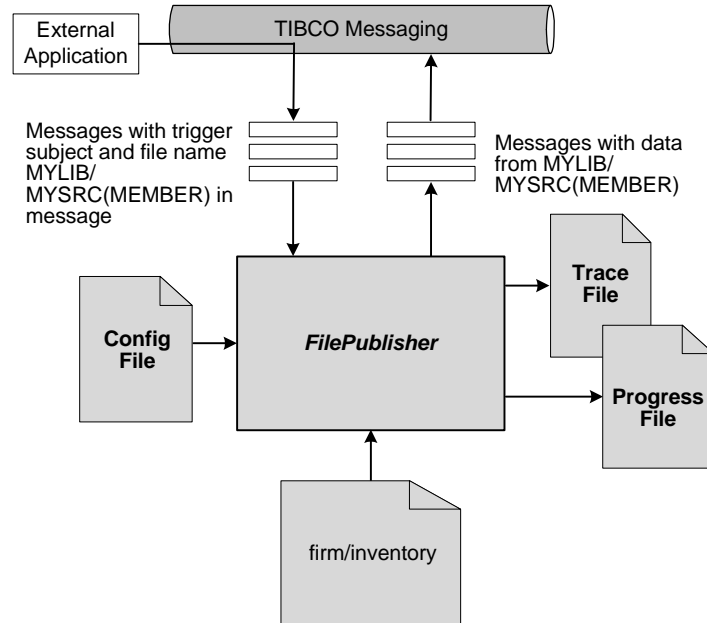


FilePublisher creates a progress file (.PRG) that can be used for viewing the status of file processing. A progress file tracks how far into the file we have published and have received acknowledgements from all subscribers. It is used for recovery.

## Trigger-Subject Method

In the trigger-subject method, FilePublisher subscribes to a subject that triggers publishing. If the trigger subject contains a file name in the message, FilePublisher publishes the file as is from its current location. The file is not moved after it is published.

Figure 4 Using a Trigger Subject With a File Name



## FileSubscriber

As an adapter subscription service, you can use FileSubscriber by specifying the following options:

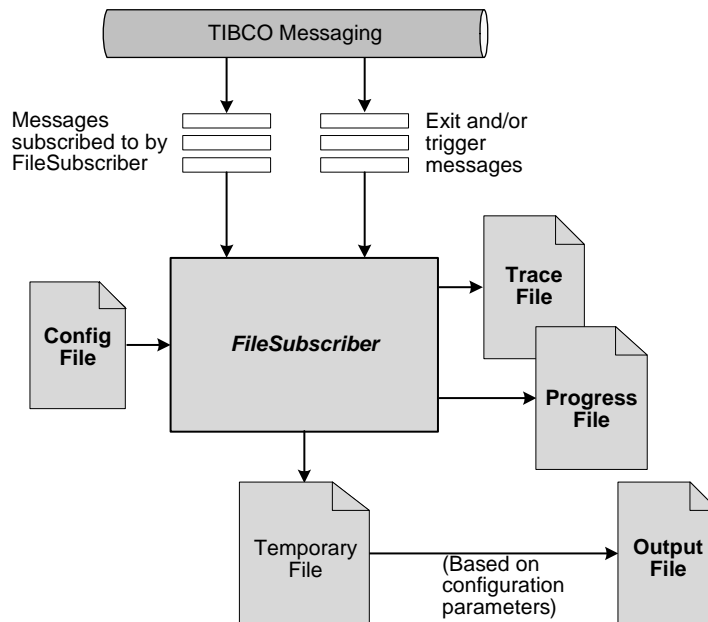
**Data Transfer Mode:** determines whether FileSubscriber transfers data in Block Mode or Record Mode. Each of these modes in turn has a guaranteed mode and a reliable mode. For guaranteed delivery, you can use either ECM with the RV message transport or durable topics for the EMS transport. For details, see [Data Transfer Modes on page 10](#).

**Processing and File Generation Options:** the service can be configured to generate files in a specific library; the service also provides options for executing a command or calling a program before or after generating an output file, and for determining when and how to generate the output file.

**Transport Options:** supports the TIBCO Rendezvous, TIBCO Rendezvous Certified Messaging (RVCN), and TIBCO EMS message transport (reliable and certified).

Depending on the chosen transport option, FileSubscriber listens to either TIBCO Rendezvous or EMS messages on predefined subjects or destinations and generates output records to a file. FileSubscriber keeps the output in a temporary file and maintains a progress file (Figure 5).

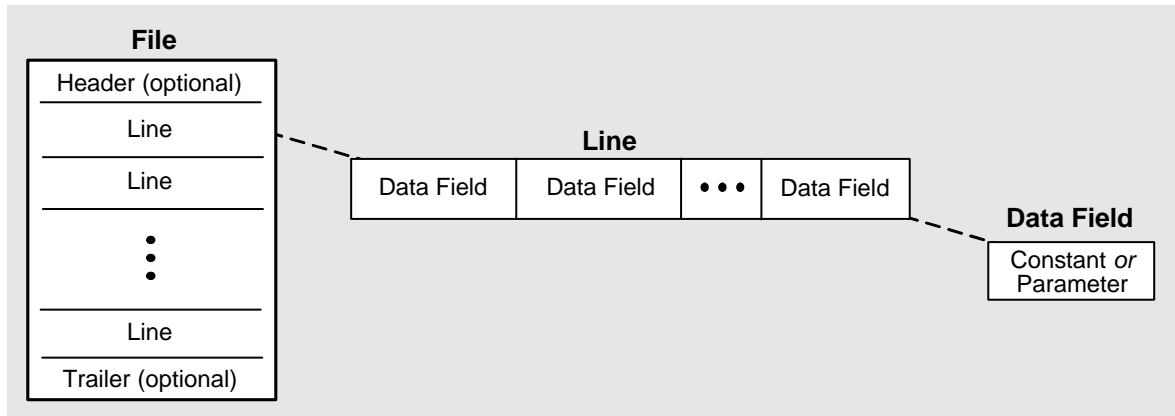
Figure 5 FileSubscriber Workflow



## Generating Output Files

The format of an output file is shown in [Figure 6](#).

Figure 6 Format of an Output File



When FileSubscriber writes the output file, it uses the `filePrefix` parameter as the output file name. You can also optionally append the system date and time to the file name by using the `appendDateTime` parameter.

## Message Delivery Considerations

---

This section describes the types of message delivery provided by TIBCO Adapter for Files (IBM i).

### Data Transfer Modes

You can use TIBCO Adapter for Files (IBM i) to transfer the data in files using the following modes:

- **Block Mode:** the file is read in blocks without regard to its record or field structure. The data are published as blocks and must be subscribed to in block transfer mode as well.
- **Record Mode:** the records and field structure of the data are read and published.

In addition, each of the above data transfer modes has a guaranteed and reliable mode, as follows:

- **Explicit Confirmation Mode (ECM):** guaranteed mode for RV transport
- **Durable Subscribers:** guaranteed delivery for EMS transport

For additional information, see [Block Transfer Mode on page 120](#).

### Transport Options

You can use TIBCO Adapter for Files (IBM i) to publish or subscribe to data using the following message delivery types:

- **Standard Rendezvous reliable delivery (RV):** reliable message delivery that works well for many programs.
- **Rendezvous certified message delivery (RVCM):** provides stronger assurances of delivery, along with tighter control, greater flexibility, and fine-grained reporting.
- **TIBCO Enterprise Message Service (EMS):** provides guaranteed and reliable delivery

For additional information, see *TIBCO Rendezvous Concepts* and *TIBCO Enterprise Message Service User's Guide*.



ECM cannot be used with RVCM.





For transfers between the adapter and TIBCO ActiveMatrix Adapter for Files, only block mode over RV is supported with and without ECM.

## Comparing ECM and RVCN for Guaranteed Delivery

As noted previously, ECM cannot be used with RVCN, and therefore must be used with standard Rendezvous reliable delivery. ECM provides FTP-like functionality with error handling and recovery. While RVCN supports certified messaging, it can be slow when handling large data sets. ECM provides a simplified confirmation mechanism that is suitable for handling large data sets, and the adapter can handle data sets of any size.

By default, the adapter uses RVCN to guarantee message delivery. With RVCN, all messages are held in the publisher's ledger file, along with each subscriber's state, until they are confirmed by all subscribers. While RVCN guarantees message delivery, its major disadvantage is that it requires significantly more disk space and memory. RVCN works well in environments that do not require high volume data rates. Its main strength is that it sends messages asynchronously and handles confirmations in the ledger without the application having to deal with it.

In block transfer mode, the adapter must support high volumes of data (files of 200MB or more), and its performance in transferring files is critical. If RVCN were used to transfer such large data sets, it would require an impractical and unfeasible amount of system resources. Also the data that the adapter is publishing is already held on disk in the input file, so storing it again in the RVCN ledger adds significantly more overhead.

The following differences between ECM and RVCN are notable:

- With RVCN, slow consumers can cause excessive growth in the memory of FilePublisher. In contrast, memory stays constant with ECM. The only trade-off is that all subscribers must confirm each block, since the state is maintained per file type, not per subscriber.
- Since FilePublisher is aware of when the confirmations arrive, it can send the next set of blocks as soon as all the confirmations arrive. This significantly improves the performance of ECM, and it can operate at the speed of the subscriber.

ECM for block transfer requires that each subscriber confirm each block message. The publisher sends out a set of blocks containing file data, and it does not send the next set until it has received confirmations from each subscriber. If the publisher does not receive a block confirmation within a specified time limit, the publisher resends the block message (or messages). This mechanism ensures that

the process memory does not to grow based on how fast the subscribers can process the data. Also, because the block data is already persisted in the input file and the publisher (FilePublisher) maintains its state, guaranteed delivery and recovery are achieved.



The following import and export subjects are used by ECM for the handshakes between Publisher and Subscriber:

- `_TIBCO_AE_ADAPTER_FAFT_SUBTOPUB` . *<endpoint subject name>*
- `_TIBCO_AE_ADAPTER_FAFT_PUBTOSUB` . *<endpoint subject name>*

For additional information, see [File Transfer Using ECM on page 121](#).

## Supported File Types and Data Types

---

The adapter supports a FileType parameter designation of SEQ only. Keyed or random access is not currently supported.

Both FilePublisher and FileSubscriber support the following numeric (non-text) data types:

- **Zoned Decimal**

The rightmost four bits of a byte are called the numeric bits (N) and normally consist of a code representing a decimal digit. The leftmost four bits of a byte are called the zone bits (Z), except for the rightmost byte of a decimal operand, where these bits might be treated either as a zone or as a sign (S).

- **Binary (COMP and COMP-4)**

A binary number occupying 2, 4, or 8 bytes of storage and is handled for arithmetic purposes as a fixed-point number with the leftmost bit being the operational sign.

- **Packed Decimal (COMP-3).**

Each byte contains two decimal digits (D), except for the rightmost byte, which contains a sign to the right of a decimal digit.

- **Floating Point – Single (COMP-1)**

A 4-byte single-precision, floating point number. COMP-1 fields do not require a precision definition tag.

- **Floating Point – Double (COMP-2)**

An 8-byte double-precision, floating point number. COMP-2 fields do not require a precision definition tag.

### Field Precision and Length

The precision of a field is its length and number of decimals, and is specified using the “precision” tag. The format is `precision="n,p"` where `n` is the number of digits for the number (including decimals) and `p` is the number of decimals.

The length of a field in a record depends on the numeric format and specified precision in the field.

- **Zoned Type:** Zoned numbers occupy 1 byte of storage for each numeric digit specified.

- **Binary Type:** Binary numbers occupy 2, 4, or 8 bytes of storage, depending on the specified precision: 2 bytes: n from 1 to 4; 4 bytes: n from 5 to 9; and 8 bytes: n from 10 to 18.
- **Packed Type:** Packed numbers occupy from 1 to 15 bytes of storage. The number of bytes is determined from the formula  $n/2 + 1$ . For example, a field with `precision="7,2"` would occupy 4 bytes of storage.

## BusinessEvents™ Messages

---

TIBCO Adapter for Files (IBM i) can provide business-event level notifications for TIBCO BusinessEvents.

To configure TIBCO Adapter for Files (IBM i) to create business-event messages, add the following element in the Options section of the FileSubscriber and FilePublisher configuration files:

```
EEM_SUBJECT = "destination_subject_name"
```

where *destination\_subject\_name* is the destination to which the event messages are routed.

## JMS Overview

---

Java Message Service 1.1 (JMS) is a Java framework specification for messaging between applications. Sun Microsystems developed this specification, in conjunction with TIBCO Software and others, to supply a uniform messaging interface among enterprise applications.

You can use the Message Service to integrate the applications within an enterprise. Message-oriented-middleware (MOM) creates a common communication protocol between these applications and you can easily integrate new and existing applications in your enterprise computing environment.

JMS is based on creation and delivery of messages. Messages are structured data that one application sends to another. The creator of the message is known as the *producer* and the receiver of the message is known as the *consumer*. The TIBCO EMS server acts as an intermediary for the message and sends it to the correct destination. The server also provides enterprise-class functionality such as fault-tolerance, message routing, and communication with other messaging systems, such as TIBCO Rendezvous® and TIBCO SmartSockets™.

The diagram below illustrates an application producing a message, sending it by way of the server, and a different application receiving the message.

Figure 7 Message Delivery



JMS supports two messaging models:

- Point-to-point (queues)
- Publish and subscribe (topics)

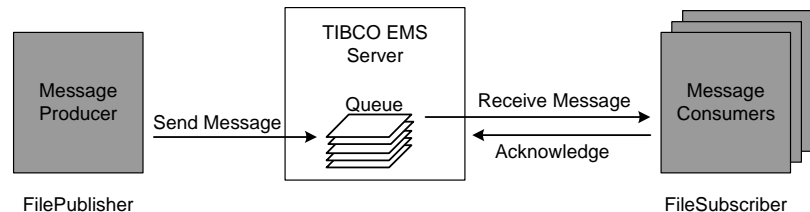
### Point-to-Point Messaging

Point-to-point messaging has one producer and one consumer per message. This style of messaging uses a *queue* to store messages until they are received. The message producer sends the message to the queue; the message consumer retrieves messages from the queue and sends acknowledgement that the message was received.

More than one producer can send messages to the same queue, and more than one consumer can retrieve messages from the same queue. The queue can be configured to be exclusive, if desired. If the queue is exclusive, then all queue messages can only be retrieved by the first consumer specified for the queue. Exclusive queues are useful when you want only one application to receive messages for a specific queue. If the queue is not exclusive, any number of receivers can retrieve messages from the queue. Non-exclusive queues are useful for balancing the load of incoming messages across multiple receivers. Regardless of whether the queue is exclusive or not, only one consumer can ever retrieve each message that is placed on the queue.

The diagram below illustrates point-to-point messaging using a non-exclusive queue. Each message consumer receives a message from the queue and acknowledges receipt of the message. The message is taken off the queue so that other consumers cannot receive it.

Figure 8 Point-to-point Message



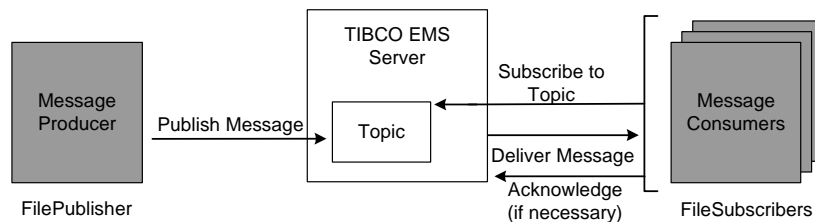
## Publish and Subscribe Messaging

In a publish and subscribe message system, producers address messages to a topic. In this model, the producer is known as a *publisher* and the consumer is known as a *subscriber*.

Many publishers can publish to the same topic, and a message from a single publisher can be received by many subscribers. Subscribers subscribe to topics, and all messages published to the topic are received by all subscribers to the topic. This type of message protocol is also known as *broadcast* messaging because messages are sent over the network and received by all interested subscribers, similar to how radio or television signals are broadcast and received.

The diagram below illustrates publish and subscribe messaging. Each message consumer subscribes to a topic. When a message is published to that topic, all subscribed consumers receive the message.

Figure 9 Publish and subscribe messages



There can be a time dependency in the publish and subscribe model. By default, subscribers only receive messages when they are active. If messages are delivered when the subscriber is not available, the subscriber does not receive those messages. JMS specifies a way to remove part of the timing dependency by allowing subscribers to create durable subscriptions. Messages for durable subscriptions are stored on the server until the message expires or the storage limit is reached. Subscribers can receive messages from a durable subscription even if the subscriber was not available when the message was originally delivered.

## Controlling the Flow of Messages

You can control the flow of messages to a destination. This is useful when message producers send messages much faster than message consumers can receive them.



## JMS Message Structure

---

JMS messages have a standard structure. This structure includes the following sections:

- Header (required)
- Properties (optional)
- Body (optional)

The JMS specification details a standard format for the header and body of a message. Properties are provider-specific and can include information on specific implementations or enhancements to JMS functionality. The adapter supports the following properties:

Property	Description
JMS_TIBCO_COMPRESS	Set this property in senders to request that EMS compress the message before sending it to the server.
JMS_TIBCO_DISABLE_SENDER	Set this property in senders to prevent the EMS server from including the sender name in the message when the server sends it to consumers.
JMS_TIBCO_MSG_TRACE	<p>Specified in the FILE_OPTIONS element of the FileType section of the FilePublisher and FileSubscriber configuration files. Valid values are 'body' and 'null'</p> <ul style="list-style-type: none"><li>• When body is specified, the entire EMS message is tracked on the EMS server.</li><li>• When null is specified, only the header of the message is tracked.</li></ul> <p>See the EMS documentation for additional information on properties.</p>
JMS_TIBCO_PRESERVE_UNDELIVERED	Specifies the message is to be placed on the undelivered message queue if the message must be removed.

The JMS standard specifies two delivery modes for messages, `PERSISTENT` and `NON_PERSISTENT`. TIBCO EMS also includes `RELIABLE_DELIVERY`. This delivery mode eliminates some of the overhead associated with the other delivery modes.

For consumer sessions, you can also specify that consumers do not need to acknowledge receipt of messages, if desired.

More information about properties specific to TIBCO EMS can be found in the *TIBCO Enterprise Message Service Java API Reference*.

## Chapter 2      **Installation**

This chapter explains how to install the TIBCO Adapter for Files (IBM i) software.

### Topics

---

- [Installation Requirements, page 22](#)
- [Distribution Media and Contents, page 23](#)
- [Uploading the Software, page 24](#)
- [Installing the Software, page 28](#)
- [Testing the Adapter, page 31](#)
- [Uninstalling the Software, page 40](#)

## Installation Requirements

---

Before starting the installation procedure, review the topics in this section to verify that your system meets the basic requirements and that you have installed the required software.

For this release, the product is installed under a new Resource ID.

For instructions on uninstalling the current or a previous version of the product, see [Uninstalling the Software on page 40](#).

### System Requirements

TIBCO Adapter for Files (IBM i) requires a minimum version of Power Systems server that runs IBM i V7R1 with TCP/IP installed.

### Storage Requirements

TIBCO Adapter for Files (IBM i) requires approximately 75 MB of auxiliary storage to install.

### Software Requirements

TIBCO Adapter for Files (IBM i) runs in a native IBM i environment. The following software is required and must be installed and configured before starting the installation procedure:

- Operating System -- IBM i V7R1 or higher.
- **Optional.** QShell -- Licensed Program 5722-SS1 Option 30.
- The TCP/IP Connectivity.
- A DNS server that can resolve the hostname and host address of the TIBCO Rendezvous and EMS server.
- \*PRV CL Compiler Support -- Licensed Program 5722-SS1 Option 9.

## Distribution Media and Contents

---

This section describes how to obtain the software, and the installation files included in the distribution media. The software is distributed as three \*SAVF files inside a ZIP file and installed with the RSTLICPGM system command.

### Obtaining the Software

You can download the TIBCO Adapter for Files (IBM i) software from the TIBCO Software web site, or request a CD.

#### Downloading from the Web Site

Completing the following task to download the software from the TIBCO web site:

1. Contact TIBCO Software Inc. for a password, directory information, and so on.
2. Connect to the TIBCO web site with the required information.
3. Download the appropriate ZIP files.

#### Requesting a CD

You can email TIBCO Software to request a CD at:

[fulfillment@tibco.com](mailto:fulfillment@tibco.com)

### Installation Files

The following ZIP file is included in the distribution media:

TIB\_fa\_5.0.0\_ibmi\_power.zip

The ZIP file contains the following three save files:

TIBFA500 – contains the \*BASE option

TIBFA500R – Option 1 for all samples using RV

TIBFA500E – Option 2 for all samples using EMS

## Uploading the Software

---

After obtaining the ZIP file, you have to upload the software to the IBM i host system using the FTP file transfer utility. Perform the following operations:

1. Download or copy the `TIB_fa_5.0.0_ibmi._power.zip` file to a PC that can connect to the IBM i.
2. Unzip this file to a temporary location on the PC. It contains the following files:
  - `TIBFA500.SAV` – TIBCO Adapter for Files (IBM i) \*BASE option
  - `TIBFA500R.SAV` – TIBCO Adapter for Files (IBM i) samples with RV; Option 1
  - `TIBFA500E.SAV` – TIBCO Adapter for Files (IBM i) samples with EMS; Option 2
3. From the temporary location on the PC, open a FTP session with the IBM i:
 

```
ftp <YOUR_IBMi>
```
4. Enter the user ID and password when requested.
5. Specify that the upcoming file transfer locations and names using IFS notation:
 

```
quote site namefmt 1
```
6. Change the default working directory (library) on the IBM i that can receive the transfer:
 

```
cd <TARGET_LIBRARY>.LIB
```
7. Change the local directory on the PC to the location where the product save files reside:
 

```
lcd <WINDOWS_DIRECTORY>
```
8. Specify a binary transfer:
 

```
bin
```
9. Transfer the `TIBFA500.SAV` file as a \*SAVF file, and do the same for the sample files:
 

```
PUT TIBFA500.SAV TIBFA500.SAVF
PUT TIBFA500R.SAV TIBFA500R.SAVF
PUT TIBFA500E.SAV TIBFA500E.SAVF
```

10. Verify that the FTP was successful by signing on to the IBM i as QSECOFR (or a user profile with similar authority) and display the contents of the \*SAVE files:

```
DSPSAVF <TARGET_LIBRARY>/TIBFA500
```

```
DSPSAVF <TARGET_LIBRARY>/TIBFA500R
```

```
DSPSAVF <TARGET_LIBRARY>/TIBFA500E
```

You can see a number of objects on the bottom half of the screen.

# Previous Releases of the Software

If previous versions of TIBCO Adapter for Files (IBM i) have been installed on the system, you can choose to delete the previous releases of the product, or install the new release in its own library.

## Identifying Previous Installations

If TIBCO Adapter for Files (IBM i) has been previously installed on the system, you can obtain information about the existing installations by executing the DSPSFWRSC command.

For example:

Resource ID	Option	Feature	Description
1TIBFA1	*BASE	5001	TIBCO Adapter for Files (IBM i) V4R7M0 GA
1TIBFA1	1	5002	TIBCO Adapter for Files (IBM i) V4R7M0 GA RV samp
1TIBFA1	2	5003	TIBCO Adapter for Files (IBM i) V4R7M0 GA EMS samp
1TIBFA1	*BASE	5001	TIBCO Adapter for Files (IBM i) V4R7M0 HF3
1TIBFA1	1	5002	TIBCO Adapter for Files (IBM i) V4R7M0 HF3 RV samp
1TIBFA1	2	5003	TIBCO Adapter for Files (IBM i) V4R7M0 HF3 EMS sam

If you press F11, you can also see the product library name and release:

Resource ID	Option	Feature	Feature Type	Library	Release
1TIBFA1	*BASE	5001	*CODE	TIBFALIB	V4R7M0
1TIBFA1	1	5002	*CODE	TIBFASMPRV	V4R7M0
1TIBFA1	2	5003	*CODE	TIBFASMPER	V4R7M0
1TIBFA1	*BASE	5001	*CODE	TIBFA4703	V4R7MC
1TIBFA1	1	5002	*CODE	TIBFA4703R	V4R7MC
1TIBFA1	2	5003	*CODE	TIBFA4703E	V4R7MC

You can choose to either delete the previous release of the product, or install the new release alongside the existing release. To install the new release alongside any existing releases, see [Multiple Installations on page 29](#).



## Deleting Previous Installations

To delete a previous installation of TIBCO Adapter for Files (IBM i), you can execute the `DLTLICPGM` command specifying the resource ID for the installation to be removed. Note that you can obtain the resource ID using the `DSPSEWRSC` command, as described in [Identifying Previous Installations](#).



If you do not want to delete all releases associated with a resource ID, make sure that you release-qualify the operation. Furthermore, for options other than `*BASE`, you can specify a specific option to be deleted. For example, to delete option 1 for TIBCO Adapter for Files (IBM i) release 4.7.0 GA only, run:

```
DLTLICPGM LICPGM(1TIBFA1) OPTION(1) RLS(V4R7M0)
```

To delete all options for TIBCO Adapter for Files (IBM i) release 4.7.0 GA, including `*BASE`, run:

```
DLTLICPGM LICPGM(1TIBFA1) RLS(V4R7M0)
```

To delete all GA and hotfix releases of TIBCO Adapter for Files (IBM i) release 4.7.0, run:

```
DLTLICPGM LICPGM(1TIBFA1)
```

## Installing the Software



If no previous installations of TIBCO Adapter for Files (IBM i) exist on the system, you can follow one path of the RSTLICPGM command. See [Single Installation](#) for more details.

If a previous installation of TIBCO Adapter for Files (IBM i) exists on the system, you must be careful as to which parameters you specify with RSTLICPGM. See [Multiple Installations](#) for more details.

### Single Installation

Perform the following steps to install the TIBCO Adapter for Files (IBM i) software. All program objects are located within \*SAVF TIBFA500.

1. Sign-on as QSECOFR or as a user profile with similar authority.
2. Use the Restore Licensed Program (RSTLICPGM) command to load the software:

```
RSTLICPGM LICPGM(2TIBFA0) DEV(*SAVF)
SAVF(<TARGET_LIBRARY>/TIBFA500)
```

Parameters:

- LICPGM(2TIBFA0) – specifies the Resource ID of the product being restored.
- DEV(\*SAVF) – the device name where the product resides; in this case, the device is a \*SAVF whose name is specified in the SAVF parameter.
- SAVF(<TARGET\_LIBRARY>/TIBFA500) – \*SAVF TIBFA500 resides in library <TARGET\_LIBRARY>.

3. **Optional.** Use the RSTLICPGM command to load the samples that use RV:

```
RSTLICPGM LICPGM(2TIBFA0) DEV(*SAVF) OPTION(1)
SAVF(<TARGET_LIBRARY>/TIBFA500R)
```

Parameters:

The same syntax as the preceding step, with OPTION(1) specified. This parameter specifies which of the optional parts of the licensed program (specified in the LICPGM parameter) is to be restored.

4. **Optional.** Use the RSTLICPGM command to restore the samples that use EMS:

```
RSTLICPGM LICPGM(2TIBFA0) DEV(*SAVF) OPTION(2)
SAVF(<TARGET_LIBRARY>/TIBFA500E)
```

Parameters:

The same syntax as the preceding step, with `OPTION(2)` specified. This parameter specifies which of the optional parts of the licensed program (specified in the `LICPGM` parameter) is to be restored



After becoming familiar with the product and the samples, you can omit installing the samples on subsequent machines.

The above `RSTLICPGM` command restores the product components into the appropriate default library names, as follows:

- `TIBFALIB` for `*BASE` option
- `TIBFASMPRV` for `Option 1`
- `TIBFASMPER` for `Option 2`

## Multiple Installations

Perform the following steps to install the TIBCO Adapter for Files (IBM i) Software.

1. Sign-on as `QSECOFR` or as a user profile with similar authority.
2. If you are installing this release alongside previous installations, perform the following command:

```
— RSTLICPGM LICPGM(2TIBFA0) DEV(*SAVF)
   SAVF(<TARGET_LIBRARY>/TIBFA500) RLS(VvRrMm) REPLACERLS(*NO)
   LIB(yourbaselib)
```

This command restores the `*BASE` option for the product to library *yourbaselib*.

Parameters:

- `LICPGM(2TIBFA0)` – specifies the Resource ID of the product being restored.
- `DEV(*SAVF)` – the device name where the product is located; in this case, the device is a `*SAVF` whose name is specified in the `SAVF` parameter.
- `SAVF(<TARGET_LIBRARY>/TIBFA500)` – `*SAVF TIBFA500` resides in library `<TARGET_LIBRARY>`.
- `RLS(VvRrMm)` – specifies the version, release, and modification level of the licensed program being restored. `VvRrMm` refers to the release level where *v* is the version number, *r* is the release number, and *m* is the modification level. For example, for release 5.0.0, `VvRrMm` is `V5R0M0`.
- `REPLACERLS(*NO)` – specifies the version, release, and modification level of the licensed program being replaced. `*NO` indicates that the licensed program currently installed on the system should not be replaced if the release version is the same. The licensed program being restored must be a different release

than the one currently installed. If the licensed program being restored exists in the same library as the installed program, an override parameter must be specified indicating to which library the licensed program is restored.

- **LIB** – specifies the library into which the licensed program is being restored.
3. **Optional.** Use the **RSTLICPGM** command to restore the samples that use **RV**:
    - **RSTLICPGM LICPGM(2TIBFA0) DEV(\*SAVF) OPTION(1)**  
**SAVF(<TARGET\_LIBRARY>/TIBFA500R) RLS(VvRrMm) REPLACERLS(\*NO)**  
**LIB(*youropt1lib*)**

This command restores option 1 for the product to library *youropt1lib*.

Parameters:

The same syntax as the preceding step, with **OPTION(1)** specified. This parameter specifies which of the optional parts of the licensed program (specified in the **LICPGM** parameter) is to be restored.

4. **Optional.** Use the **RSTLICPGM** command to restore the samples that use **EMS**:

```
RSTLICPGM LICPGM(2TIBFA0) DEV(*SAVF) OPTION(2)
SAVF(<TARGET_LIBRARY>/TIBFA500E) RLS(VvRrMm) REPLACERLS(*NO)
LIB(youropt2lib)
```

This command restores option 1 for the product to library *youropt2lib*.

Parameters:

The same syntax as the preceding step, with **OPTION(2)** specified. This parameter specifies which of the optional parts of the licensed program (specified in the **LICPGM** parameter) is to be restored.

## Testing the Adapter

---

After installation, you can test the adapter using RV or EMS according to the sample that you have installed. The sample provides sample configuration files for the FilePublisher and FileSubscriber, and sample data files. This section explains how to test the adapter using the samples.

### Sample File Contents

Option 1 of TIBCO Adapter for Files (IBM i) is for the RV samples, which are restored in library TIBFASMPRV by default. Similarly, Option 2 is for the EMS samples, which are restored in library TIBASMPM. Each set of the samples provides the following files:

- File README with member README providing information that is specific to the sample set.
- File INIFPUB with sample configuration members PUB001, PUB002 and so on as appropriate for the sample set. Each member contains the configuration parameters for FilePublisher to do a specific type of work. See the README file for details, or the prologue at the top of each member.
- File INIFSUB with sample configuration members SUB001, SUB002 and so on as appropriate for the sample set. Each member contains the configuration parameters for FileSubscriber to do a specific type of work. See README file for details, or the prologue at the top of each member.
- File QCLSRC with members RUN001, RUN002 and so on as appropriate for the sample set. Each member contains CL code to start up a FilePublisher/FileSubscriber pair. Member SNDTRG002 contains the CL code to send a trigger to PUB002. Member COMPILE contains CL that can be used to compile the SNDTRG002 program, RUNFA001, RUNFA002 and so on.
- The following data files are used with the sample configuration files above: BIN2, FT11, FT21, and FT31.
- Backup copies of the data files above: BKP\_BIN2, BKP\_FT11, BKP\_FT21, and BKP\_FT31.
- Program COMPILE matching the member in QCLSRC.
- Programs RUNFA001, RUNFA002 and so on, and SNDTRG002 matching the members in QCLSRC. These programs were created when the option was restored using the COMPILE program.
- Trace log files PUBLOG and SUBLOG, into which trace logging information can be written.

Each FilePublisher/FileSubscriber pair (except PUB002/SUB002) utilizes file polling at timer intervals to query whether there are any *matching* files in INPUT\_LIBRARY to publish. PUB002/SUB002 starts publishing when FilePublisher receives an external trigger.

Note that each executing FilePublisher/FileSubscriber pair can be configured to handle multiple FileType definitions concurrently. So all non-conflicting INIFPUB file members could be combined into one member, and FilePublisher can process all of these FileType definitions at the same time. The same is true for the Subscriber. For details on FileType definitions, see [FileType Section on page 57](#).

## Submitting the Initial Sample Using RV

To submit the initial sample, perform the following steps:

1. Start TIBCO Rendezvous.

```
SBMJOB CMD(CALL PGM(TIBRV/RVD) PARM('-listen' 'tcp:7500'))
ALWMLTTHD(*YES)
```

This example assumes that the RV daemon is running on the same machine as the adapter. If communicating with a RV daemon running on a remote machine, the remote host name has to be specified. For example, remote\_host1:7500. For additional details, see the TIBCO Rendezvous documentation.

2. Add the following libraries to the library list:

```
ADDLIB TIBFASMPRV
ADDLIB TIBFALIB
```



The publishing process requires a PROCESS\_LIBRARY for holding intermediate data and a FilePublisher OUTPUT\_LIBRARY for holding the file after it is published. In addition, there must be a FileSubscriber OUTPUT\_LIBRARY for holding progress files, intermediate results, and the final published file. These three libraries are required before starting FilePublisher and FileSubscriber.

3. Create the required process and output libraries using the names provided by either of the methods below:

- Accept the FAPUBPROC, FAPUBOUT, and FASUBOUT library names which are already specified as defaults in the sample configuration files provided.

or

- Modify TIBFASMPRV/INIFPUB(PUB001) by changing names provided for PROCESS\_LIBRARY and OUTPUT\_LIBRARY to libraries of your choice, and

modify TIBFASMPRV/INIFSUB(SUB001) by changing the OUTPUT\_LIBRARY to a library of your choice.



Before proceeding, note that file FT11 exists in library TIBFASMPRV. If this file is not there because of a previous sample run, duplicate the file from BKP\_FT11.

4. Start FilePublisher and FileSubscriber using either of the methods below:

- Call the sample program RUNFA001.

```
CALL RUNFA001
```

or

- Submit the jobs to batch individually from a command entry screen.

```
"SBMJOB      CMD(CALL PGM(SXF3RSUB) PARM('-config'
'TIBFASMPRV/INIFSUB(SUB001)')) ALWMLTTHD(*YES)
```

```
"SBMJOB      CMD(CALL PGM(SXF3RPUB) PARM('-config'
'TIBFASMPRV/INIFPUB(PUB001)')) ALWMLTTHD(*YES)
```



You could also start FilePublisher and FileSubscriber interactively in the QShell environment, as follows:

Display 1:

```
ADDLIBLE TIBFALIB
qsh
cd /usr/tibco/tibfa/PRODUCT-LIBRARY-NAME
sxf3rsub -config 'TIBFASMPRV/INIFSUB(SUB001)'
```

where *PRODUCT-LIBRARY-NAME* is the name of the library in which the product option is installed, usually TIBFALIB.

Display 2:

```
ADDLIBLE TIBFALIB
qsh
cd /usr/tibco/tibfa/PRODUCT-LIBRARY-NAME
sxf3rpub -config 'TIBFASMPRV/INIFPUB(PUB001)'
```

where *PRODUCT-LIBRARY-NAME* is the name of the library in which the product option is installed, usually TIBFALIB.

When the FilePublisher and FileSubscriber jobs are submitted to batch, ensure that the batch job queues are active and that the programs are running.

Note also that to execute multithreaded applications from QShell, you must ensure that the environment variable QIBM\_MULTI\_THREADED has a value of Y. This might be done before starting the QSHELL, as follows:

```
ADDENVVAR ENVVAR(QIBM_MULTI_THREADED) VALUE(Y) REPLACE(*YES)  
LEVEL(*SYS)
```

You can also set the environment variable at `LEVEL(*JOB)` so that the variable applies to the current job only.



## Submitting the Initial Sample Using EMS

To submit the initial sample, perform the following steps:

1. Before submitting the samples edit the `EMS_SESSION` parameter in the publisher and subscriber configuration files to reference your EMS Server. For example if your EMS Server is running on a machine with IP address `10.98.32.103` and is using the default port the entry, would look like this:

```
EMS_SESSION = { providerURL="tcp://10.98.32.19:7222",
name="EMS_P" }
```

For additional details, see the TIBCO Enterprise Message Service documentation.

2. Add the following libraries to the library list:

```
ADDLIB TIBFASMPPEM
ADDLIB TIBFALIB
```



The publishing process requires a `PROCESS_LIBRARY` for holding intermediate data and a `FilePublisher OUTPUT_LIBRARY` for holding the file after it is published. In addition, there must be a `FileSubscriber OUTPUT_LIBRARY` for holding progress files, intermediate results, and the final published file. These three libraries are required before starting `FilePublisher` and `FileSubscriber`.

3. Create the required process and output libraries using the names provided by either of the methods below:

- Accept the `FAPUBPROC`, `FAPUBOUT`, and `FASUBOUT` library names which are already specified as defaults in the sample configuration files provided.

or

- Modify `TIBFASMPPEM/INIFPUB(PUB001)` by changing names provided for `PROCESS_LIBRARY` and `OUTPUT_LIBRARY` to libraries of your choice, and modify `TIBFASMPPEM/INIFSUB(SUB001)` by changing the `OUTPUT_LIBRARY` to a library of your choice.



Before proceeding, note that file `FT11` exists in library `TIBFASMPPEM`. If this file is not there because of a previous sample run, duplicate the file from `BKP_FT11`.

## 4. Start FilePublisher and FileSubscriber using either of the methods below:

- Call the sample program RUNFA001.

```
CALL RUNFA001
```

or

- Submit the jobs to batch individually from a command entry screen.

```
"SBMJOB      CMD(CALL PGM(SXF3ESUB) PARM('-config'
'TIBFASMPPEM/INIFSUB(SUB001)')) ALWMLTTHD(*YES)

"SBMJOB      CMD(CALL PGM(SXF3EPUB) PARM('-config'
'TIBFASMPPEM/INIFPUB(PUB001)')) ALWMLTTHD(*YES)
```



You can also start FilePublisher and FileSubscriber interactively in the QShell environment, as follows:

## Display 1:

```
qsh
cd /usr/tibco/tibfa/PRODUCT-LIBRARY-NAME
sxf3esub -config 'TIBFASMPPEM/INIFSUB(SUB001)'
```

where *PRODUCT-LIBRARY-NAME* is the name of the library in which the product option is installed, usually TIBFALIB.

## Display 2:

```
qsh
cd /usr/tibco/tibfa/PRODUCT-LIBRARY-NAME
sxf3epub -config 'TIBFASMPPEM/INIFPUB(PUB001)'
```

where *PRODUCT-LIBRARY-NAME* is the name of the library in which the product option is installed, usually TIBFALIB.

When the FilePublisher and FileSubscriber jobs are submitted to batch, ensure that the batch job queues are active and that the programs are running.

Note also that to execute multithreaded applications from QShell, you must ensure that the environment variable QIBM\_MULTI\_THREADED has a value of Y. This might be done before starting the QShell, as follows:

```
ADDENVVAR ENVVAR(QIBM_MULTI_THREADED) VALUE(Y) REPLACE(*YES)
LEVEL(*SYS)
```

You can also set the environment variable at `LEVEL(*JOB)` so that the variable applies to the current job only.

## Sample Results

Prior to starting the adapter, file FT11 was in library TIBFASMPRV or TIBFASMPM. When the adapter was started, FilePublisher searched for matches for files with a prefix of FT1 specified in PUB001. File FT11 matched the prefix and was published. The file was copied to the PROCESS\_LIBRARY, transmitted to FileSubscriber, and then moved to the FilePublisher OUTPUT\_LIBRARY. FileSubscriber received the file and placed it in its OUTPUT\_LIBRARY. FilePublisher then continues to poll for files with the prefix FT1.

## Stopping the Adapter

FilePublisher and FileSubscriber can be stopped by sending termination messages with appropriate subject names, as shown below (QShell environment):

1. Start QShell:

```
QSH
```

2. Change to the following directory:

```
CD /usr/tibco/tibfa/PRODUCT-LIBRARY-NAME
```

Which is the name of the library in which the product is installed, usually TIBFALIB.

3. Follow these steps to end the sample:

- a. Send a termination message to end the RV sample:

```
sxftrigr -service 7500 -daemon tcp:host.example.com:7500  
_FILEADAPTER.FILE_PUB_001.TERMINATE 'any message'
```

Make sure the RV daemon parameter is updated to reflect the host where the RV daemon is running.

- b. Send a termination message to end the EMS sample:

```
sxftrige -server 'tcp://host.example.com:7222' '-queue'  
_FILEADAPTER.FILE_EMS_PUB_001.TERMINATE 'any message'
```

Make sure the EMS server parameter is updated to reflect the host where it resides.

The termination message contains the name of the FilePublisher or FileSubscriber instance that you want to stop. For details on the adapter instance names, see the ADAPTER\_NAME element in either FilePublisher or FileSubscriber configuration in Chapter 3.

## Checking the Trace Logs

FilePublisher and FileSubscriber can create trace logs in the files specified by their respective configuration files. You can browse these trace logs for any error, warning, or trace messages.

Examples of trace logs from a successful session and a failed session are shown in [Appendix A, Trace Log Examples](#). Samples PUB001 and SUB001 have been configured to write the trace logs in TIBFASMPRV/PUBLOG(PUB001) and TIBFASMPRV/SUBLOG(SUB001), respectively.



More configuration files and data files are supplied along with the adapter software for your reference. For more information about these files see [Sample Configuration and Data Files on page 44](#).

## Messages to Operator

Any major check point messages, such as start, stop, and error messages, are sent to QSYSOPR \*MSGQ. The command to view the messages is as follows:

```
DSPMSG QSYSOPR
```

## Uninstalling the Software

---



Before uninstalling, you have to back up any files that were created during installation and that you have modified, such as configuration files. The uninstallation program deletes all files that were originally created during installation. If you have modified any installed files, the files are removed.

Use the IBM i command `DLTLICPGM` to remove the software from the system:

To delete the current version (5.0), use the `DLTLICPGM LICPGM(2TIBFA0)` command.

To delete the previous version (4.7), use the command `DLTLICPGM LICPGM(1TIBFA1)`.

## Chapter 3 **Configuring the Adapter**

This chapter describes the choices you have for configuring the adapter and how to create the required configuration files.

### Topics

---

- [Overview, page 42](#)
- [FilePublisher Configuration, page 47](#)
- [FileSubscriber Configuration, page 79](#)
- [Guaranteed Delivery for EMS Messages, page 113](#)

# Overview

This section describes the structure of the configuration files, configuration element syntax, and the supported files and data types.

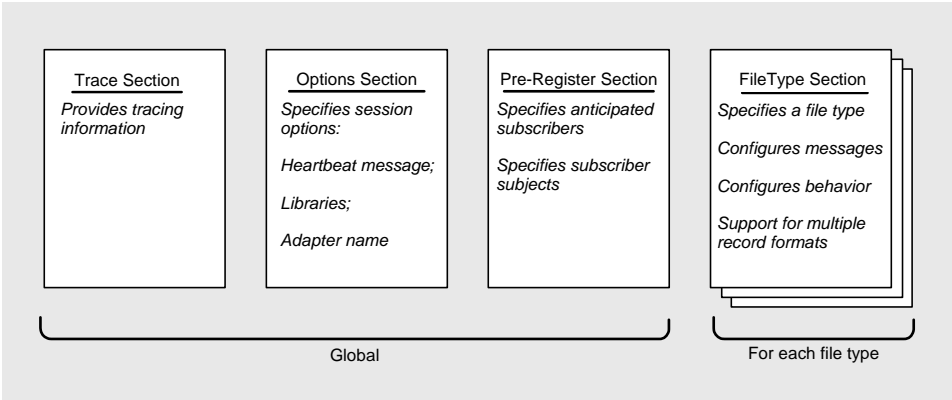
## Configuration File Structure

The FilePublisher and FileSubscriber components of TIBCO Adapter for Files (IBM i) have separate configuration files.

### FilePublisher

The FilePublisher configuration file consists of the sections shown in the following figure.

Figure 10 Sections of a FilePublisher Configuration File



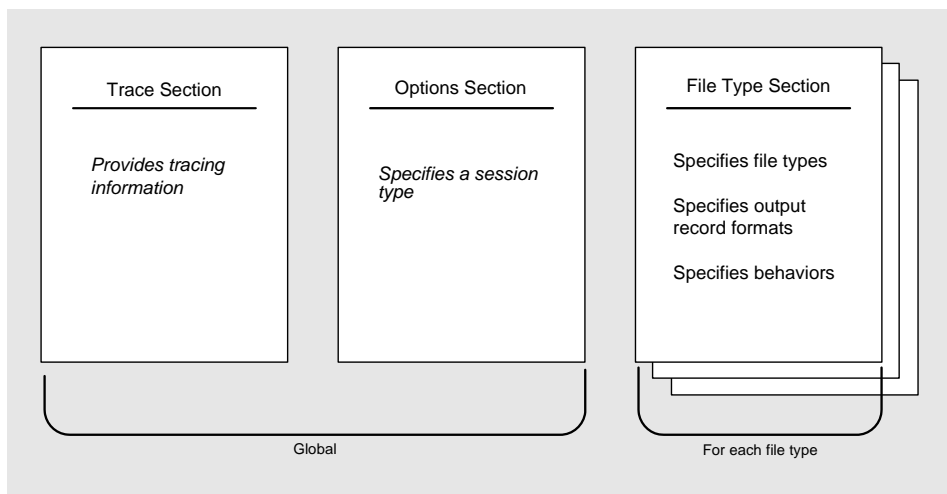
The Trace, Options, and Pre-Register sections contain global configuration values, and there can be one or more FileType sections.



## FileSubscriber

The FileSubscriber configuration file consists of the sections shown in the following figure.

Figure 11 Sections of a FileSubscriber Configuration File



The Trace and Options sections contain global configuration values, and there can be one or more FileType sections.

## Configuration Syntax

Each section of the FilePublisher or FileSubscriber configuration files contain the following organizing entities:

- **element** – the basic configuration unit. Elements are written in upper case letters.
- **parameter** – a sub-unit of an element
- **tag** – a sub-unit of a parameter

Each section consists of elements, and elements consist of parameters surrounded by curly brackets. Each parameter has either a value or consists of tags, surrounded by curly brackets.

For example, the `FILE_OPTIONS` element from the `FileType` section shown below consists of parameters with values, separated by commas:

```
FILE_OPTIONS = {    filePrefix="ft1",
                   useFilePolling="true",
                   pollInterval="5000",
                   datasetType="SEQ",
                   ...
                 }
```

The `FILE_LINE` element shown below consists of multiple `field` parameters that have as their value a set of tags with values (separated by commas).

```
FILE_LINE = {
  field = { description = "Part Number", fromMessage="true",
            length = "6", type="STRING", value="PartNo", length = "11",
            position="0"},

  field = { description = "Description", fromMessage="true",
            type="STRING", value="Desc", length = "15",
            position="11" },
  ..... }
```



Enclose each element, parameter set, and tag set in curly brackets {}, as illustrated in the examples.

## Sample Configuration and Data Files

The adapter includes both sample configuration files and sample data sets for your use. You have several choices when configuring the adapter.

- Use the sample configuration files supplied with the adapter.
- Copy one of the sample configuration files supplied with the adapter and change it to reflect your particular environment. A detailed discussion of the individual parameters is provided in [FilePublisher Configuration on page 47](#) and [FileSubscriber Configuration on page 79](#).
- Write your own configuration files from scratch. See [FilePublisher Configuration on page 47](#) and [FileSubscriber Configuration on page 79](#) for a discussion of available options and parameters.

### Sample Configuration Files

The adapter requires two configuration files to operate: one for `FilePublisher` and one for `FileSubscriber`. The sample configuration files and data files in the list that follows are supplied along with the adapter software for your reference. They show the usage of many of the features provided by the adapter software.

## **Publisher**

- PUB001 — Configuration file with example of publishing a delimited text file in Record Mode.
- PUB002 — Configuration file with example of publishing a fixed length text file in Record Mode.
- PUB003 — Configuration file with example of publishing different header and detail formats.
- PUB004 — Configuration file with example of publishing a file with numeric data types, such as packed, zoned, binary, and so on.
- PUB005 — Configuration file with example of publishing a text file in Record Mode with ECM.
- PUB006 — Configuration file with example of publishing a text file in Block Mode irrespective of the record structure.
- PUB007 — Configuration file with example of publishing a text file in Block Mode with ECM.
- PUB008 — Configuration file with example of publishing a text file in Block Mode with Guaranteed EMS delivery feature.
- PUB009 — Configuration file with example of publishing a text file with Record Mode Guaranteed EMS delivery feature.

## **Subscriber**

- SUB001 — Configuration file with example of subscribing and generating a delimited text file in Record Mode.
- SUB002 — Configuration file with example of subscribing and generating a fixed length text file in Record Mode.
- SUB003 — Configuration file with example of subscribing supporting different header and detail formats.
- SUB004 — Configuration file with example of subscribing to a file with numeric data types like packed, zoned, binary, and so on.
- SUB005 — Configuration file with example of subscribing and generating a text file in Record Mode with ECM.
- SUB006 — Configuration file with example of subscribing and generating a text file in Block Mode irrespective of the record structure.
- SUB007 — Configuration file with example of subscribing and generating a text file in Block Mode with ECM.

- SUB008 — Configuration file with example of publishing a text file in Block Mode with Guaranteed EMS delivery feature.
- SUB009 — Configuration file with example of subscribing and generating a textile with Record Mode Guaranteed EMS delivery feature.

### **Sample Data Files**

The following sample data files are included for testing the adapter.

- FT11 — Sample data file with delimited text.
- FT21 — Sample data file with fixed length records.
- FT31 — Sample data file with different header and detail formats.
- BIN2 — Sample data file containing numeric data types.

## FilePublisher Configuration

This section describes the FilePublisher configuration file. The sections are listed in the order in which the sections appear in the file; the elements that comprise each section are listed in alphabetical order. In addition, sample elements are provided at the end of the section.

### Trace Section

You can use the Trace section to control the trace logging behavior of FilePublisher. This is the first section in the configuration file. The elements specified in this section are listed in the following table; all are required (except those used at the request of TIBCO Support).

Element	Description
FILE_COUNT	Specifies how many log files to keep. Each saved old log file name has a number (from 1 to the FILE_COUNT minus 1) suffixed to it. For example, if FILE_NAME is specified as FPLOG, and FILE_COUNT is set to 5, four log files are saved, named FPLOG1 through FPLOG4, with FPLOG4 being the earliest (or oldest) file.
FILE_LIMIT	Specifies the size of the log file in bytes. When this limit is exceeded, the current log file is saved, and a new log file is created.
FILE_NAME	Identifies the log file to be used for trace messages. This is a source physical file. If the file does not exist, then FilePublisher creates one. The default record length is 266 bytes. Default trace log files, PUBLOG and SUBLOG, are available in TIBFASMPRV and TIBFASMPER depending on which product options were installed. If you require a larger record length, you can create your own file; for example:  CRTSRCPF FILE(TIBFALIB/TIBLOGNEW) RCDLEN(1000)
PRINT_STDOUT	Specifies whether to send the trace log messages to standard output (STDOUT) when set to true. This setting is in addition to the trace log file specified with FILE_NAME. If the adapter jobs are submitted to batch, STDOUT appears in the QPRINT printer file. If QSHELL is used to submit the adapter jobs, STDOUT appears on screen.

Element	Description
TRACE_EMS_ EPM_ERROR_ MSGs	<p>Specifies if JMS tracing enabled for ERROR or EPM messages when using EMS. The valid values are <code>none</code>, <code>epm</code>, <code>error</code>, <code>both</code>. The default value is <code>none</code>.</p> <p>A value of <code>none</code> maps to null for both EPM and ERROR messages.</p> <p>A value of <code>epm</code> turns on JMS Message body tracing for all EPM messages sent by the adapter to the EEM_DESTINATION.</p> <p>A value of <code>error</code> turns on JMS Message body tracing for all ERROR messages sent by the adapter to the ERROR_DESTINATION.</p> <p>A value of <code>both</code> turns on JMS Message body tracing for all EPM and ERROR messages sent by the adapter.</p>
TRACE_EMS_ HEARTBEAT_ MSGs	<p>Specifies if JMS tracing enabled for HEARTBEAT messages when using EMS. The valid values are <code>true</code>, <code>false</code>. The default value is <code>false</code>.</p> <p>A value of <code>true</code> turns on JMS Message body tracing for all Heartbeat messages sent by the adapter. A value of <code>false</code> maps to null.</p>
TRACE_HEAP	Used for diagnostic tracing in cases of memory exhaustion. Only used at the request of TIBCO Support. Valid value: <code>true</code>

Element	Description
TRACE_LEVEL	<p>Specifies the kind of information that FilePublisher logs. FilePublisher generates trace messages according to the trace level that you specify with this element.</p> <ul style="list-style-type: none"> <li>• <b>Trace level 1</b> generates the following session-level messages: <ul style="list-style-type: none"> <li>— Initialization of TIBCO Rendezvous, including the version number</li> <li>— TIBCO Rendezvous sessions created</li> <li>— Poll timers and publish trigger subjects set for each file type</li> <li>— Initialization of the TIBCO EMS session created</li> <li>— Initialization of TIBCO EMS session created</li> <li>— Publish trigger destinations for each file type</li> </ul> </li> <li>• <b>Trace level 2</b> generates level 1 messages plus all the values loaded from the config file as below: <ul style="list-style-type: none"> <li>— All values loaded from the Options section of the configuration file</li> <li>— All values loaded from the Pre-Register section of the configuration file</li> <li>— All values loaded from the FileType sections of the configuration file</li> </ul> </li> <li>• <b>Trace level 3</b> generates level 1 and 2 messages plus the following field-related messages: <ul style="list-style-type: none"> <li>— Field processing information (label type, and so on.)</li> <li>— Certified Messaging publisher information (name, sequence number, and so on.)</li> <li>— Message buffer reallocation</li> <li>— Traces all key I/O Open/Close actions against S/390 files</li> <li>— Traces flow through key publishing modules</li> <li>— Traces flow of re-Publishing requests</li> </ul> </li> <li>• <b>Trace level 4</b> for detailed tracing. Usually used only at the request of TIBCO Support. <ul style="list-style-type: none"> <li>— Traces detailed I/O calls for reading/writing data.</li> <li>— Traces detailed processing of Container fields</li> </ul> </li> <li>• <b>Trace level 5</b> for detailed tracing. Usually used only at the request of TIBCO Support. <ul style="list-style-type: none"> <li>— Traces all TIBCO RV and EMS and EMS received message callbacks</li> </ul> </li> </ul>

Element	Description
TRACE_LEVEL (Contd)	<ul style="list-style-type: none"> <li>• <b>Trace Level 6</b> generates level 1 through 5 messages. Usually used only at the request of TIBCO Support. <ul style="list-style-type: none"> <li>— Traces all detailed timer callbacks used for handling ECM admin messages, re-publish messages, file lock retry, heartbeat messages, and so on.</li> </ul> </li> <li>• <b>Trace Level 10</b> generates Level 1 through 6 messages. Usually used only at the request of TIBCO Support. <ul style="list-style-type: none"> <li>— Detailed trace of all TIBCO RV and EMS and EMS message allocates and frees</li> <li>— Detailed trace of heap usage during message processing</li> </ul> </li> <li>• <b>Trace Level 15</b> generates Level 1 through 10 messages. Usually used only at the request of TIBCO Support. <ul style="list-style-type: none"> <li>— Detailed message content trace (traces the actual contents of the messages, up to 32K bytes)</li> </ul> </li> </ul>
TRACE_SWITCHES	Special trace switches used by TIBCO support. Do not set this unless explicitly told to by TIBCO Support. The valid values are an 8 byte text string. The default value is NNNNNNNN.
UNIT_TRACE	Allows non-SMS (System-Managed Storage) sites to specify the associated UNIT parameter to be used with the VOLSER for trace (.LOG) files. See VOLSER_TRACE, below.
VOLSER_TRACE	Allows non-SMS (System-Managed Storage) sites to specify a specific MVS volume (or set of volumes) to be used for storing trace (.LOG) files. Must be used with the UNIT_TRACE element. See UNIT_TRACE, above.



## Options Section

You can configure the following options in the Options section:

- Establish the type of TIBCO EMS/Rendezvous or TIBCO EMS session to be used by FilePublisher for publishing
- Name the adapter instance
- Control heartbeats
- Define certain global elements that apply to all file types defined in the configuration file. These global elements can be overridden in the file type definitions by using the appropriate tags.

Element	Description
ADAPTER_NAME	<p>Specifying this element uniquely identifies the adapter instance. This element:</p> <ul style="list-style-type: none"> <li>• Used to set the terminate subject or destination (for EMS) to <code>_FILEADAPTER.&lt;adapter name&gt;.TERMINATE</code>. Sending a message to this subject can stop FilePublisher.</li> <li>• Used to set the heartbeat subject or destination (for EMS) to <code>_FILEADAPTER.&lt;adapter name&gt;.HEARTBEAT</code>. This is used by FilePublisher to send heartbeat messages, but does not mean that heartbeats are always published. To publish (send) heartbeats, see <code>PUBLISH_HEARTBEAT</code>. For terminate messages, the default destination type is <code>QUEUE</code>. For heartbeat messages, the default destination type is <code>TOPIC</code>.</li> </ul>
CONTINUE_ON_CONFIG_ERROR	<p>Specifies whether to abend the adapter when it encounters an initialization error for any FileType section. If set to <code>true</code>, and if a configuration error is found for a file, the publisher generates a message to the log and goes on to the next file. If set to <code>false</code>, abend the adapter.</p> <p>Valid values: <code>true</code>, <code>false</code>. The default value is <code>false</code>.</p>
DELETE_PUB_PRG_FILES	<p>Specifies whether to remove the progress file at the end of a file transfer. The valid values:</p> <ul style="list-style-type: none"> <li>• <code>none</code>—do not remove the progress file.</li> <li>• <code>single</code>—remove the progress file when it is a 1:1 relationship between the publisher and subscriber.</li> <li>• <code>multiple</code>—remove the progress file.</li> </ul> <p>The default value is <code>none</code>.</p>

Element	Description
ECM_NESTED_TEXT_LINES	<p>Specifies if ECM messages have to be sent to the Distributed File Adapter in extended format.</p> <p>The valid values are <code>true</code>, <code>false</code>. The default value is <code>false</code>.</p>
EEM_DESTINATION	<p>Identifies the destination for messages that the adapter passes to TIBCO BusinessEvents. The value is the destination to which the event messages is routed.</p>
EEM_SUBJECT	<p>Identifies the subject for messages that the adapter passes to TIBCO BusinessEvents. The value is the destination to which the event messages is routed.</p>
EMS_SESSION	<p>Specifying this element along with the following required parameters establishes reliable mode publishing:</p> <ul style="list-style-type: none"> <li>• <code>providerURL</code> — The URL of the TIBCO EMS server</li> <li>• <code>name</code> — Unique name for the connection (optional)</li> <li>• <code>username</code> — The user name used to connect to the EMS server</li> <li>• <code>password</code> — The password for the user.</li> </ul>
ERROR_DESTINATION	<p>Specifies where the error messages publish to. If this parameter is not blank, all <code>traceErr</code> and <code>FatalErr</code> messages are copied and sent to that destination.</p> <p>Valid values: blank or a string. The default value is blank <code>""</code>.</p> <p><b>Note:</b> Specify this field at the end of the Options section. Since the Rendezvous transport/EMS connection is not created until the end of the Options section, any error that occurs cannot be published.</p>
ERROR_EXIT_CC	<p>Specifies the return code that FilePublisher returns when exiting with an error condition.</p> <p>Valid values: 4, 8. The default value is 8.</p>
ERROR_SUBJECT	<p>Specifies where the error messages publish to. If this parameter is not blank, all <code>traceErr</code> and <code>FatalErr</code> messages are copied and sent to that subject.</p> <p>Valid values: blank or a string. The default value is blank <code>""</code>.</p>

Element	Description
ERROR_DESTINATION	<p>Specifies where the error messages publish to. If this parameter is not blank, all traceErr and FatalErr messages are copied and sent to that destination.</p> <p>Valid values: blank or a string. The default value is blank "".</p> <p><b>Note:</b> Specify this field at the end of the Options section. Since the Rendezvous transport or EMS connection is not created until the end of the Options section, any error that occurs can't be published.</p>
HEARTBEAT_FILE_INFO	<p>If set to <code>true</code>, additional file type information is also published in the heartbeat message. For example, file prefix, file extension, and number of messages published are included.</p> <p>Valid values: <code>true</code>, <code>false</code>. The default value is <code>true</code>.</p>
HEARTBEAT_TIME	<p>Specifies the interval between heartbeat messages.</p> <p>The value must be greater than or equal to 100 milliseconds.</p> <p>The default value is 60000 milliseconds, or 60 seconds.</p>
HOST_CODEPAGE	<p>Works with the codepage support in TIBCO Rendezvous 7.1 and TIBCO EMS 4.2 and above.</p> <p>Specifies the translation table to use on the IBM i mainframe side. Defaults to the CCSID of the job in which the Publisher or Subscriber is running and which is usually tied to the profile of the user who initiated the job.</p>
INPUT_LIBRARY	<p>Default library data set to search for publishing files.</p>
KILL_PUBLISHER	<p>If set to <code>true</code>, enables a publisher to shut down when the Rendezvous daemon shuts down.</p> <p>If not specified, defaults to <code>false</code>, which means that the publisher continues to run even if the Rendezvous daemon shuts down.</p> <p><b>Note:</b> This functionality is provided by default in TIBCO EMS. Whenever an exception occurs in a client connection to an EMS server, it is handled by default.</p>
MAX_CONCURRENT_JOBS	<p>This limits the number of files that are published at the same time (concurrently). This is used to limit excessive memory and CPU consumption.</p> <p>The default value is 12.</p>

Element	Description
NETWORK_CODEPAGE	Works with the codepage support in TIBCO Rendezvous 7.1 or TIBCO EMS 4.2 and above. Specifies the expected codepage sent by any subscribers. Defaults to codepage 819.
OUTPUT_LIBRARY	Default output library to move files to when they have been published using polling.
PERSIST_EMS_ECM_MSGS	<p>Specifies if ECM messages that are sent by EMS are tagged as PERSISTENT or NON-PERSISTENT.</p> <p>Valid values: <code>true</code> or <code>false</code>. The default value is <code>true</code>.</p> <p>A value of <code>true</code> causes ECM messages sent by EMS to be tagged as PERSISTENT. A value of <code>false</code> causes them to be marked as NON-PERSISTENT.</p>
PRINT_FILE_OPTIONS	If <code>true</code> , prints all configuration information at the adapter startup. Valid values: <code>true</code> , <code>false</code> . The default value is <code>true</code> .
PROCESS_LIBRARY	Default process library to be used for holding intermediate and status information while a file is being published.
PUBLISH_HEARTBEAT	A setting to <code>true</code> for this element enables publishing of heartbeat messages. If not specified it defaults to <code>false</code> , meaning heartbeat messages are not published.
RV_SESSION	<p>Specifying this element along with the following required parameters establishes a reliable mode of publishing:</p> <ul style="list-style-type: none"> <li><code>name</code> — A unique alphanumeric name identifying the TIBCO Rendezvous session</li> <li><code>service</code> — The service group for this session</li> <li><code>network</code> — Network to initialize a TIBCO Rendezvous session</li> <li><code>daemon</code> — The name of the TIBCO Rendezvous daemon for this session</li> </ul> <p>If you include this element, omit the <code>RVCN_SESSION</code> element.</p>

Element	Description
RVCM_SESSION	<p>Specifying this element establishes a certified messaging session. Include all parameters listed for RV_SESSION, plus the following options:</p> <ul style="list-style-type: none"> <li>• <code>ledgerFile</code> — The name of the file-based ledger for Certified Messaging. This file is created in the Integrated File System.</li> <li>• <code>requireOldMessages</code> — Indicates whether a persistent correspondent requires delivery of messages sent to a previous session with the same name for which delivery was not confirmed. A setting to <code>true</code> enables delivery of old unacknowledged messages; a setting to <code>false</code> does not.</li> <li>• <code>defaultTimeLimit</code> — Sets the default message time limit for all outbound certified messages. The time is specified in seconds and the default value is 60.</li> <li>• <code>syncLedger</code> — If you want to use a synchronous ledger file, set this parameter to <code>true</code>. The default value for this is <code>false</code>, meaning an asynchronous ledger file is used.</li> </ul> <p>If you include this element, omit the <code>RV_SESSION</code> element.</p>
EMS_SESSION	<p>Specifying this element along with the following required parameters establishes reliable mode publishing:</p> <ul style="list-style-type: none"> <li>• <code>providerURL</code> — The URL of the TIBCO EMS server</li> <li>• <code>name</code> — Unique name for the connection (optional)</li> <li>• <code>username</code> — The user name used to connect to the EMS server</li> <li>• <code>password</code> — The password for the user.</li> </ul>
TERMINATE_ON_RV_SEND_FAIL	<p>Specifies that the adapter abandons if it cannot successfully call the <code>tibrvsend</code> API. This flag is only applicable to record mode processing. If the file is in block mode, the flag is ignored and the adapter is abandoned. Upon recovery, the adapter re-syncs to the last check point. In the case of standard block mode, the entire file is retransmitted. In ECM, retransmission begins with the last block that was successfully acknowledged. Valid values: <code>true</code>, <code>false</code>. The default value is <code>false</code>.</p>
WRITE_TO_SYSLOG	<p>If <code>true</code>, major checkpoint log messages are sent to <code>QSYSOPR *MSGQ</code>. Each message includes date and time information. Valid values: <code>true</code>, <code>false</code>. The default value is <code>false</code>.</p>

Element	Description
EOL_CHARACTER	<p>Specifies the character that is used as a delimiter between records in a block. This is necessary when the IBM i platform and the platform that is being targeted do not share a translation table for translating an end-of-line character properly.</p> <p>For example, to communicate the proper line breaks to a Microsoft Windows platform from X'15' to X'0A' there is no system translation table that does this. You can use this parameter to specify the character that translates correctly to X'0A'. Therefore, you would specify <code>EOL_CHARACTER = "25"</code>.</p> <p>The <code>EOL_CHARACTER</code> is specified as a two-character literal that represents the EBCDIC hexadecimal value.</p>

Pre-Register Section

You can use the Pre-Register section to configure and specify the list of anticipated subscribers for the subjects that are published. When the anticipated subscribers are pre-registered, the delivery of Certified Messages is guaranteed regardless of when the subscriber is started, before or after FilePublisher, or when FileSubscriber goes down and is restarted. To use this feature, both FilePublisher and FileSubscriber must establish an `RVCN_SESSION` with the `requireOldMessage` parameter set to `true`.



Not supported for EMS.

Use of this section in your configuration file is optional. If it appears, you have to define it after the Options section. Use the following element to specify the anticipated subscribers for each subject.

Element	Description
SUBSCRIBER	<p>Specify the following required parameters within braces ( { } ) for each anticipated subscriber for each subject. Up to 126 SUBSCRIBER elements can be defined in the configuration file.</p> <ul style="list-style-type: none"><li>listenerName — Specifies the name of the subscribing session.</li><li>listeningSubject — Specifies the name of the subscription subject.</li></ul>

## FileType Section

The FileType section contains two elements:

- FILE\_OPTIONS element
- MESSAGE\_FIELDS element

### FILE\_OPTIONS Element

Use the FILE\_OPTIONS element to specify the following options:

- The type of file to publish
- How you want FilePublisher to process the file
- Where the files are located
- How the data in the file is to be accessed
- How the file is to be published

The FILE\_OPTIONS element is also used to identify those files you want to be published in certified mode if you have started a certified FilePublisher session. A config file can have multiple [FILE OPTIONS] definitions in a single file. See the File PublisherExamples section for details.

The following table lists the parameters in the FILE\_OPTIONS elements.

Parameter	Description
filePrefix (required)	Used to select files from the input library data set for publishing. While polling, any files with a name that starts with this file prefix are selected for publishing. This prefix can be up to 5 characters.  When set to the string NULL, and if trigger mode is used, and the full path of the file is specified, then the publisher cannot ignore this file type. It generates the temporary file without using the file prefix.
publishSubjectName (required)	The subject name to which messages are sent to publish a file.
publishDestinationName	Specifies the destination name to which messages are sent to publish a file.
publishDestinationType	Specifies the type of the destination to which message are published. For additional details, see <a href="#">JMS Overview on page 16</a> .  Valid values: TOPIC, QUEUE . The default value is TOPIC.

Parameter	Description
deliveryMode	<p>Specifies the delivery mode for messages. For additional details, see <a href="#">JMS Message Structure on page 19</a>.</p> <p>Valid values: PERSISTENT, NON_PERSISTENT, RELIABLE_DELIVERY.</p> <p>The default value is PERSISTENT.</p>
JMS_TIBCO_MSG_TRACE	<p>Specifies whether the entire message, or only the header of the message, is traced. For additional details, see <a href="#">JMS Message Structure on page 19</a>.</p> <p>Valid values: body, null.</p>
EMS_ACK_TYPE	<p>Specifies the acknowledge mode for EMS when durable subscribers are used.</p> <p>Valid values are TIBEMS_AUTO_ACKNOWLEDGE, TIBEMS_CLIENT_ACKNOWLEDGE, TIBEMS_DUPS_OK_ACKNOWLEDGE, TIBEMS_EXPLICIT_CLIENT_ACKNOWLEDGE, TIBEMS_EXPLICIT_CLIENT_DUPS_OK_ACKNOWLEDGE, TIBEMS_NO_ACKNOWLEDGE.</p> <p>The default value is TIBEMS_AUTO_ACKNOWLEDGE.</p>
publishDestinationName	<p>Specifies the destination name to which messages are sent to publish a data file.</p>
publishDestinationType	<p>Specifies the type of destination to which messages are published.</p> <p>The valid values are TOPIC, QUEUE. The default value is TOPIC.</p>
deliveryMode	<p>Specifies the delivery mode for messages.</p> <p>The valid values are PERSISTENT, NON_PERSISTENT, RELIABLE_DELIVERY. The default value is PERSISTENT.</p>
JMS_TIBCO_MSG_TRACE	<p>Specifies whether the entire message, or only the header of the message, is traced.</p> <p>Valid values: body, null.</p>



Parameter	Description
<b>Publishing Correlation Identifier</b>	
useTrackingId	<p>Specifies whether a GUID-like transaction id is to be used. If the file is configured as needing one, and if no trackingId string is received as part of the trigger message, then the publisher automatically generates a GUID-like string for the trackingId. If the trackingId string comes with the trigger message, the id is forwarded to the subscriber. This information is sent after the publisher receives the final EOF message from the subscriber.</p> <p>Valid values: true, false (default). If true, the following fields are attached to the message that is published on the subject specified by trackinIdSubject:</p> <ul style="list-style-type: none"> <li>• AdapterName</li> <li>• FileName</li> <li>• GdgTriggerName (N/A for IBM i, z/OS only for GDG files)</li> <li>• FileExtension (N/A for IBM i, NT only, null if z/OS)</li> <li>• FilePrefix (FileType Prefix)</li> <li>• FileSize</li> <li>• fileTransferDuration</li> <li>• DateTime</li> <li>• TransferMode (BLOCK or RECORD)</li> <li>• PubLocalResult (Publisher's Local result = 0 or 8)</li> <li>• Result (0 or 4 or 8 = OK or Fail) overall transfer result from Subscriber</li> <li>• status (Succeeded or Failed)</li> <li>• HostName</li> <li>• Subject</li> <li>• trackingId</li> </ul>
trackingIdSubject	Name of the subject to which messages containing the trackingId information are published.
trackingIdDestination Name	Name of the destination to which messages containing the trackingId information are published.

Parameter	Description
<b>Publishing Method (Polling or Trigger Subject)</b>	
<code>useFilePolling</code>	If set to <code>true</code> , <code>FilePublisher</code> polls for file names to publish from <code>INPUT_LIBRARY</code> whose names begin with the <code>filePrefix</code> value. A setting to <code>false</code> disables file polling. The default value is <code>false</code> .
<code>pollInterval</code>	If file polling is enabled, specifies the time delay between file polls. The time is in milliseconds and the default value is 1000.
<code>triggerSubjectName</code>	Subject to which <code>FilePublisher</code> listens in order to publish a file.
<code>triggerDestinationName</code>	Destination to which <code>FilePublisher</code> listens.
<code>triggerDestinationType</code>	Specifies the type of destination to which trigger messages are to be sent.  Valid values: <code>TOPIC</code> , <code>QUEUE</code> . The default value is <code>TOPIC</code> .
<code>triggerFieldName</code>	Use this parameter if the file name is contained in the trigger subject message as a named field.
<b>Message Publishing</b>	
<code>messagesPerTransaction</code>	Number of messages to publish at one time (as one transaction). The default value is 10.
<code>transactionDelay</code>	Acts as a pause between transmission of records or blocks.  For ECM, this parameter is used for an additional purpose in conjunction with <code>retransmissionDelayTicks</code> . In this case, the publisher checks for any pending confirmations and re-transmit any unacknowledged data messages. See <code>retransmissionDelayTicks</code> . Also used to set up the Admin Timeout timer for resending startup messages to the subscribers.  The default value is 2000 msec.
<code>startPublishSubject</code>	The subject on which <code>FilePublisher</code> sends a message when it starts to publish the file. The published message contains the name of the file that is published.
<code>startPublishDestinationName</code>	Specifies the destination to which <code>FilePublisher</code> sends messages when publishing a file. The published message contains the name of the file that is published.  <b>Note:</b> Destination type is the same as the <code>publishDestinationType</code>

Parameter	Description
endPublishSubject	<p>The subject on which FilePublisher sends a message after it has completed publishing the file. The end publish message contains the name of the file that is published and the number of messages published.</p> <p>This parameter has a dual purpose. The first is to communicate to the FileSubscriber an end-of-file condition. The second is to communicate that the file has been published.</p> <p>The durable subscribers in EMS require that endPublishSubject equal publishSubject (or endPublishDestinationName equal publishDestinationName).</p> <p>Note that publishing an end-of-file indication on a different subject than publishSubject in RV might not maintain the correct sequence of messages.</p>
endPublishDestinationName	<p>Specifies the destination to which FilePublisher sends a message after it has completed publishing a file. The end publish message contains the name of the file that is published and number of messages published.</p> <p><b>Note:</b> Destination type is the same as the publishDestinationType</p>
generateFileFieldName	<p>Most usages of generateFileSubjectName require that it match the subscribeSubjectName.</p> <p>The adapter uses the generateFileFieldName parameter as a message differentiator. A message arriving on a subject name can be checked to see whether it is a regular “Data” message, or whether it is a “Generate File” trigger.</p> <p>When an incoming message has a field matching the value specified in the generateFileFieldName parameter, then the message is assumed to be a “Generate File” trigger message, and the subscriber generates the final target output file.</p> <p>The default value for this field is “filename”.</p>

Parameter	Description
<b>Block Transfer Mode</b>	
transferType	<p>Specifies the data transfer mode (Record Mode or Block Mode) and whether to use ECM or non-ECM with the specified data transfer mode. Valid values includes:</p> <p>BlockModeECM – use Block Mode with ECM.</p> <p>BlockModeSFT – use Block Mode with non-ECM mode.</p> <p>RecordModeECM – use Record Mode ECM.</p> <p>RecordMode – use Record Mode with non-ECM mode</p> <p>For additional information, see <a href="#">Message Delivery Considerations on page 10</a>.</p>
blockTransferMode	The adapter reads the file in blocks, without regard to its record or field structure. Valid values: true, false (default).
blockTransferSize	The block size (the number of bytes) that the adapter reads and publishes each time it reads the file. The default value is 65536 bytes.
<b>ECM Mode</b>	
useExplicitConfirmation	Use ECM mode. Valid values: true, false (default). If true, block mode is used automatically.
retransmissionDelayTicks	<p>Only used for ECM Block Mode. This parameter is used to co-ordinate the startup handshake between the Publisher and the Subscriber.</p> <p>For the Publisher, this parameter represents the number of times the transactionDelay timer must pop before checking for any unacknowledged Message Blocks. The default value for the Publisher is (retransmissionDelayTicks*transactionDelay) sec.</p> <p>Also, this represents the number of times the transactionDelaytimer pops before the Publisher attempts to re-synchronize with the Subscriber.</p> <p>For Subscriber, this parameter represents the number of seconds that FileSubscriber has to wait before trying to connect to the Publisher after a startup. The default value for the Subscriber is (retransmissionDelayTicks * 1000) sec.</p>

Parameter	Description
<code>noWaitAfterConfirmations</code>	<p>Specifies how a publisher proceeds upon receiving all the acknowledgements from all the ECM subscribers. If set to <code>true</code>, the next transaction is processed immediately after FilePublisher receives all the confirmations. If set to <code>false</code>, it uses the normal <code>transactionDelay</code> parameter to process transactions.</p> <p>Valid values: <code>true</code> (default), <code>false</code>. Set to <code>true</code> for ECM publishers to speed up delivery.</p>
<code>ECMSubscriberName</code>	<p>Identifies the ECM subscriber with which this publisher is exchanging acknowledgements in block mode. Repeat this entry for every participating ECM subscriber. This name must be the same as the <code>ECMSubscriberName</code> parameter that is specified in the subscriber's configuration file.</p> <p><b>Warning:</b> You cannot specify both an <code>ECMSubscriberName</code> and a <code>confirmationSubject</code>.</p>
<code>confirmationSubject</code>	<p>Confirmation subject name used by FilePublisher to exchange block confirmation messages in record mode.</p> <p><b>Warning:</b> You cannot specify both a <code>confirmationSubject</code> and an <code>ECMSubscriberName</code>.</p>
File Type and Location	
<code>dataSetType</code>	Specifies that only sequentially organized data is allowed. Specify <code>SEQ</code> .
<code>inputLibrary</code>	Name of the input library. The default value is the <code>INPUT_LIBRARY</code> specified in the Options section. This overrides the value specified in <code>INPUT_LIBRARY</code> .
<code>processLibrary</code>	Name of the process library for this file type. The default value is the <code>PROCESS_LIBRARY</code> specified in the Options section. This overrides the value specified in <code>PROCESS_LIBRARY</code> .
<code>outputLibrary</code>	Name of the output library for this file type. The default value is the <code>OUTPUT_LIBRARY</code> specified in the Options section. This is used in conjunction with polling. This overrides the value specified in <code>OUTPUT_LIBRARY</code> .
<code>lineLength</code>	Record length of the file being published. If <code>lineLength</code> is not specified, the adapter uses the catalog service to obtain the file attributes.

Parameter	Description
<b>Certified Publishing</b>	
<code>isCertified</code>	If the FilePublisher session is not a certified session, this parameter is ignored. If the FilePublisher session is specified as a certified session, the default value for this parameter is <code>true</code> , meaning the file is published in certified mode. If you want to have a specific file published in non-certified (Reliable) mode, you can specify <code>false</code> .
<code>rvcmTimeLimit</code>	Specify a value for this parameter to override the default value specified in the Options section of the configuration file.
<b>Preprocessing and Postprocessing</b>	
<code>removeAfterProcess</code>	If you want to delete a file after it is published, use this parameter. If this parameter is set to <code>true</code> , FilePublisher deletes the file after it is successfully published. The default value is <code>false</code> .
<code>executeBeforeProcess</code>	Causes FilePublisher to execute a command or call a program before the file is published. See <a href="#">Pre-Processing and Post-Processing Files on page 125</a> .
<code>executeAfterProcess</code>	Causes FilePublisher to execute a command or call a program after generating an output file. See <a href="#">Pre-Processing and Post-Processing Files on page 125</a> .
<b>Record Handling</b>	
<code>removeLeadingBlanks</code>	Applies to STRING fields. Set this parameter to <code>true</code> to have FilePublisher strip leading blanks from a STRING.
<code>removeTrailingBlanks</code>	Applies to STRING fields. Set this parameter to <code>true</code> to have FilePublisher strip trailing blanks.
<code>keepTrailingBlanks</code>	If this parameter is set to <code>true</code> , pad records with blanks when publishing a file. Valid values: <code>true</code> , <code>false</code> (default).
<code>useFieldWidth</code>	Identifies the record structure to FilePublisher. If this value is <code>true</code> , the fields in the records are assumed to be fixed length.
<code>delimiter</code>	Identifies the record as having variable-length fields. The delimiter can be any unique character. If <code>useFieldWidth</code> is set to <code>false</code> , the default for this field is <code> </code> (vertical bar).

Parameter	Description
<code>startAtLine</code>	Used by FilePublisher to identify which record from which to start publishing. It can be used to skip over header records in a file. If this parameter is not specified, FilePublisher starts at the first record in the file.
<code>isBinary</code>	When set to <code>true</code> , specifies that the file data is in binary format. Binary data includes zoned decimal, binary, packed decimal, and floating point data types.
<code>logMatched</code>	If the values are set to <code>true</code> , these parameters specify that a log file have to be created to contain a copy of all the input records which were either matched or unmatched respectively according to the constraint definitions for multi-record-format data.  The default value for both the parameters is <code>false</code> .
<code>logUnmatched</code>	

### MESSAGE\_FIELDS Element

Use the MESSAGE\_FIELDS element of the FileType section to configure any messages that are to be published.

- To publish from a file with single record format (either delimited or fixed-width), define one MESSAGE\_FIELDS element.
- To publish from a file with multi-record format (a file with header and detail records), define one MESSAGE\_FIELDS element for each record type using the constraint parameter.



If you have enabled block transfer mode, the adapter ignores the MESSAGE\_FIELDS element.

The MESSAGE\_FIELDS element describes the format of the message to be published. It can be formed by using one or more of the following parameters.

Parameter	Description
<code>messageItem</code>	Identifies the field. You can specify different tags inside this parameter. See <a href="#">Tags in the messageItem Parameter on page 66</a> .
<code>messageContainer</code>	Contains message items. See <a href="#">Tags in the messageContainer Parameter on page 68</a> .
<code>constraint</code>	Must be used in the case of multi-record format. See <a href="#">Tags in the constraint Parameter on page 68</a> .

Tags in the messageItem Parameter

The `messageItem` parameter can identify a field with a constant value, a field from a fixed-width input record, or a field from a delimited input record. In any case, it uses the following tags:

Tag	Description
label	Name of the field.
type	<p>Data type of this field.</p> <p>Valid values: <code>STRING</code>, <code>INTEGER</code>, <code>UNSIGNED INTEGER</code>, <code>SHORT</code>, <code>UNSIGNED SHORT</code>, <code>FLOAT</code>, <code>DOUBLE</code>, <code>BOOLEAN</code>, and <code>TIME</code>. The default value is <code>STRING</code>.</p> <p>Valid values for EMS: <code>STRING</code>, <code>INTEGER</code>, <code>SHORT</code>, <code>FLOAT</code>, <code>DOUBLE</code>, <code>BOOLEAN</code>.</p> <p><b>Note:</b> When you specify <code>TIME</code> as the data type, <code>FilePublisher</code> gets the current system time and adds it to the TIBCO message. When <code>FileSubscriber</code> receives the message, it converts the time to a string that represents the time of the publishing system.</p> <p>Specify <code>OPAQUE</code> to send a field untranslated. See <a href="#">Sending Data Untranslated (OPAQUE) on page 131</a>.</p> <p>For COBOL numeric datatypes only, specify one of <code>ZONED</code>, <code>BINARY</code>, <code>PACKED</code>, <code>COMP</code>, <code>COMP-1</code>, <code>COMP-2</code>, <code>COMP-3</code>, or <code>COMP-4</code> for numeric data. See <a href="#">Sending Numeric Data Types on page 130</a>.</p> <p><b>Note:</b> Make sure <code>isBinary</code> is set to <code>true</code> for both <code>OPAQUE</code> and COBOL numeric data.</p>
precision	For COBOL numeric datatypes only, specifies the size of the field and the number of decimals.
convertToString	<p>For COBOL numeric datatypes only, specifies an alternate conversion.</p> <p>The default value is <code>false</code>, meaning the input numeric field is published as a <code>FLOAT</code> or <code>INTEGER</code> value depending on the type of the field. The datatypes <code>COMP-1</code>, <code>COMP-2</code>, and <code>COMP-3</code> are published as <code>FLOAT</code> and <code>COMP-4</code> as <code>INTEGERS</code>.</p> <p>For a more accurate conversion, specify <code>true</code>, meaning a <code>STRING</code> data type is published.</p>



Tag	Description
value	<p>You can use this tag to emulate the COBOL equivalent of LOW-VALUES and HIGH-VALUES. Valid values:</p> <p>LOW-VALUES -- Sets field's value to binary zeros.</p> <p>HIGH-VALUES -- Sets field's value to binary X'FFFFFFFF'.</p>

For a field with a constant value, add the following tags along with `label` and `type`.

Tag	Description
constant	Set this tag to <code>true</code> to specify that the field is a constant.
value	Use this tag to provide a value for a constant field.

For a field from a fixed-width input record, add the following tags along with `label` and `type` to specify the field's position within the record and the length of the field.

Tag	Description
position	Identifies the position, starting at 0, of the field within the record. Use <code>position</code> or <code>fieldStart</code> but not both.
fieldStart	Used for binary files to specify the start of a field. The first byte of an input or output record is "1." Use <code>fieldStart</code> or <code>position</code> but not both.
length	Identifies the length of the field.

For a field from a delimited input record, add the following tag along with `label` and `type` to specify the field's location within the record.

Tag	Description
location	Identifies the location of the field within the record, starting with location 1.

Tags in the messageContainer Parameter

You can group message items into containers to form a nested message using the messageContainer parameter. Include the following tags along with the messageItem or other messageContainer that you want to group together.

Tag	Description
containername	Name of the container. The name is mandatory. See Example 3, <a href="#">Record Mode Publisher – Different Header and Detail Formats on page 72</a> for an example of usage.

Tags in the constraint Parameter

The adapter supports files with multiple record formats. If there is more than one record format, include the constraint parameter using the following tags:

Tag	Description
containerName	Specifies and groups the entire message field with this name.
startNewMessage	<p>Specifies that this message field definition has to start a new message. If this tag is set to false, the TIBCO message created by the message field is combined with the previous TIBCO message.</p> <p>The default value is true for EMS and false for RV. Refer to <a href="#">Example 8, Nesting of Mapped Messages, on page 77</a>.</p>
value	Specifies the value of the record identifier field. While publishing, this MESSAGE_FIELD format is used only if this value matches the record identifier field of the input record.

For a fixed width input record, add the following tags to the containerName, startNewMessage, and value tags.

Tag	Description
position	Position, starting at 0, of the record identifier field within the record.
length	Length of the field.

For a delimited input record, add the following tag to the containerName, startNewMessage and value tags:

Tag	Description
location	Location of the record identifier field within the record, starting with location 1. The example configuration files ( <a href="#">Sample Configuration Elements on page 70</a> ) illustrate how this tag is used.

## Sample Configuration Elements

This section contains several configuration file examples that you can use as models for your own custom configuration file.

### *Example 1 Record Mode Publisher – Delimited Text File*

Configuration file name – PUB001, input file – FT11.

A portion of the file shows how the data looks:

```
115-01-0500|MONITOR|SONY|VIEWSONIC|01|350.50
115-15-6542|CPU-PIII750MHZ|COMPAQ|PRESARIO|01|900.00
115-67-7356|HDD20GB|SEAGATE|ST500|01|276.60
115-34-8767|FDD54|HP|T24333|01|86.00
115-77-5555|CDRW|HP|T75668|01|350.00
115-78-4646|KEYBOARD|COMPAQ|EASYKEY101|01|32.00
115-88-4454|MOUSE|MICROSOFT|M323|01|26.00
115-36-2727|WINDOWS2000|MICROSOFT|MSW2000|01|400.00
```

The FileType section of the configuration file:

```
[FileType]
FILE_OPTIONS = { filePrefix="FT1",
                  dataSetType = "SEQ",
                  isBinary = "false",
                  useFieldWidth = "false",
                  delimiter = "|",
                  messagesPerTransaction = "10",
                  transactionDelay = "2000",
                  publishSubjectName = "SUB.PUB001.FT1",
                  useFilePolling = "true", pollInterval="5000",
                  removeAfterProcess = "false",
                  transferType = "RecordMode"
                }

MESSAGE_FIELDS = {
  messageItem = { location="1", label="PartNo", type="STRING" },
  messageItem = { location="2", label="Desc", type="STRING" },
  messageItem = { location="3", label="Manufacturer", type="STRING"
},
  messageItem = { location="4", label="Model", type="STRING" },
  messageItem = { location="5", label="Quantity", type="INTEGER" },
  messageItem = { location="6", label="Price", type="FLOAT",
  precision="7,2" },
  messageItem = { location="7", label="Warranty", type="STRING",
  constant="true", value="1 Year" }
}
```

*Example 2 Record Mode Publisher – Fixed Length Text File*

Configuration file name – PUB002, input file – FT21.

A portion of the file shows how the data looks:

```
115-01-0500 MONITOR SONY VIEWSONIC 2 250.50
115-15-6542 CPU-PIII750MHZ COMPAQ PRESARIO 04 12.30
115-67-7356 HDD20GB SEAGATE ST500 25 15
115-34-8767 FDD54 HP T24333 1 045.22
115-77-5555 CDRW HP T57213 5 290.00
115-78-4646 KEYBOARD COMPAQ EASYKEY1 10 121.21
```

The FileType section of the configuration file:

[FileType]

```
FILE_OPTIONS = { filePrefix="FT2",
                  dataSetType = "SEQ",
                  isBinary = "false",
                  lineLength = "100",
                  startAtLine = "1",
                  useFieldWidth = "true",
                  messagesPerTransaction = "5",
                  transactionDelay = "2000",
                  publishSubjectName = "SUB.PUB002.FT2",
                  triggerSubjectName = "TRIGGER.PUB002.FT2",
                  endPublishSubject = "ENDPUB.PUB002.FT2",
                  startPublishSubject = "STARTPUB.PUB002.FT2",
                  useFilePolling = "false", pollInterval="5000",
                  removeAfterProcess = "false",
                  transferType = "RecordMode"
                }

MESSAGE_FIELDS = {
  messageItem = { position="0", length = "11", label="PartNo",
                  type="STRING" },
  messageItem = { position="11",length = "15", label="Desc",
                  type="STRING" },
  messageItem = { position="26",length = "11", label="Vendor",
                  type="STRING" },
  messageItem = { position="37",length = "12", label="Model",
                  type="STRING" },
  messageItem = { position="49",length = "2", label="Quantity",
                  type="INTEGER" },
  messageItem = { position="51",length = "8", label="Price",
                  type="FLOAT" },
  messageItem = { position="59", label="Warranty", type="STRING",
                  constant="true", value="1 Year" }
}
```

*Example 3 Record Mode Publisher – Different Header and Detail Formats*

As an example of how to use the constraint parameter, suppose a file contains both header and detail records that must be combined into one output TIBCO Rendezvous message, but the records have different formats. You could create a TIBCO Rendezvous message as follows:

1. Determine whether the record is a header or detail record. Do this by selecting a field from the record that contains unique data that identifies the field as a header record or as a detail record.
  - a. For fixed length files, identify the field by using both the position and length parameters.
  - b. For delimited files, identify the field by using the location parameter.
2. Provide the data that identifies the field type. Use the value parameter to provide the data.
3. If the record is a header record, specify that TIBCO Rendezvous must start a new message. Do this by specifying the startNewMessage parameter to true. For detail records, specify the startNewMessage parameter to false if they are to be published with the header record as a single message, otherwise specify true, which causes the records to be published as separate messages.
4. The containerName parameter is used to identify the container. For this example, the values supplied are containerName="Recfm1\_Quote" and containerName="Recfm2\_Items".

Configuration file name – PUB003, input file – FT31.

A portion of the file shows how the data looks:

```
QUOTE 15004MC001
ITEMS 115-01-0500 MONITOR SONY VIEWSONIC 2 250.50
ITEMS 115-15-6542 CPU-PIII750MHZ COMPAQ PRESARIO 04 12.30
ITEMS 115-67-7356 HDD20GB SEAGATE ST500 25 15
QUOTE 15004MC002
ITEMS 115-34-8767 FDD54 HP T24333 1 045.22
ITEMS 115-77-5555 CDRW HP T57213 5 290.00
ITEMS 115-78-4646 KEYBOARD COMPAQ EASYKEY1 10 121.21
```

The FileType section of the configuration file:

```
[FileType]
FILE_OPTIONS = { filePrefix="FT3",
                  dataSetType = "SEQ",
                  isBinary = "false",
                  lineLength = "100",
                  startAtLine = "1",
                  useFieldWidth = "true",
                  messagesPerTransaction = "10",
                  transactionDelay = "2000",
                  publishSubjectName = "SUB.PUB003.FT3",
                  useFilePolling = "true", pollInterval="5000",
                  removeAfterProcess = "false",
```

```

        transferType = "RecordMode"
    }

MESSAGE_FIELDS = {
    constraint = { position="0", length = "5", value="QUOTE",
        containerName="Recfm1_Quote",
        startNewMessage="true" },
    messageItem = { position="5", length = "5", label="QuoteRequestNo",
        type="INTEGER" },
    messageItem = { position="10", length = "5", label="SupplierCode",
        type="STRING" }
}

MESSAGE_FIELDS = {
    constraint = { position="0", length = "5", value="ITEMS",
        containerName="Recfm2_Items",
        startNewMessage="false" },
    messageItem = { position="5", length = "11", label="PartNo",
        type="STRING" },
    messageItem = { position="16",length = "15", label="Desc",
        type="STRING" },
    messageItem = { position="31",length = "11", label="Vendor",
        type="STRING" },
    messageItem = { position="42",length = "12", label="Model",
        type="STRING" },
    messageItem = { position="54",length = "2", label="Quantity",
        type="INTEGER" },
    messageItem = { position="56",length = "6", label="Price",
        type="FLOAT", precision="6,2" },
    messageItem = { label="Warranty", type="STRING", constant="true",
        value="1 Year" }
}

```

*Example 4 Record Mode Publisher - Supporting Numeric Data Types*

This example shows a FilePublisher configuration file that supports different numeric data types. It also demonstrates the use of "OPAQUE" keyword. For the FilePublisher sample configuration file, three messageItems are specified.

- The FILE\_OPTIONS element of the FileType section contains the parameter isBinary, specified as true.
- The numeric field location is specified using the position tag in the messageItem parameter. This value specifies the byte number, starting at 1, of the starting position of the field within the record.
- The precision tag in the messageItem parameter is used to identify the relative length of the field and the number of decimal positions. FilePublisher calculates the length of the field within the record depending on the data type. For example, if the field is identified as BINARY, and the precision is specified as 9, 2, then the field occupies 8 bytes of data in the record.

Configuration file name – PUB004, input file – BIN2.

A portion of the file shows how the data looks:

CHAR	PACKED	ZONED	BINARY	FLOAT	DOUBLE
abc	1,231,234.12	12,345,678.12	1,234,567	4501.00E-002	123456.7912E+005
def	1,234,567.00	98,989,898.00	1,234,567	15670.10E-001	123456.7000E+001
abc	1,231,234.12	12,345,678.12	1,234,567	14501.00E-002	123456.7912E+005
def	1,234,567.00	98,989,898.00	1,234,567	15670.10E-001	123456.7000E+001
ghi	123.00	999.00	123	12300.00E-002	123123.4567E+002
jkl	123.20	999.11	345,999	34599.90E-001	345999.1121E+004

[FileType]

```
FILE_OPTIONS = { filePrefix="BIN2",
                  dataSetType = "SEQ",
                  isBinary = "true",
                  lineLength = "95",
                  startAtLine = "1",
                  useFieldWidth = "true",
                  messagesPerTransaction = "10",
                  transactionDelay = "2000",
                  publishSubjectName = "SUB.PUB004.BIN2",
                  useFilePolling = "true", pollInterval="5000",
                  removeLeadingBlanks = "true",
                  removeTrailingBlanks = "true",
                  removeAfterProcess = "false",
                  transferType = "RecordMode"
                }
```

```
MESSAGE_FIELDS = {
  messageItem = { fieldStart="01", label="Char", length="5",
                  type="STRING" },
  messageItem = { fieldStart="06", label="Packed", precision="10,2",
                  convertToString="false", type="PACKED" },
  messageItem = { fieldStart="12", label="Zoned", precision="10,2",
                  convertToString="false", type="ZONED" },
}
```



```

messageItem = { fieldStart="22", label="Binary", precision="10,0",
convertToString="false", type="BINARY" },
messageItem = { fieldStart="30", label="Float", precision="7,2",
length="7", convertToString="false", type="COMP-1" },
messageItem = { fieldStart="34", label="Double", precision="10,4",
length="10", convertToString="false", type="COMP-2" }
}

```

#### Example 5 Record Mode Publisher with ECM

This example is for publishing a file using ECM. In Record Mode ECM, you have to specify `endPublishSubject` to signal the end of file publishing to FileSubscriber. The `generateFileSubjectName` parameter in FileSubscriber must be the same as `endPublishSubject`.

Configuration file name – PUB005, input file – FT1.

The FileType section of the configuration file:

```

[FileType]
FILE_OPTIONS = { filePrefix="FT1",
                  dataSetType = "SEQ",
                  isBinary = "false",
                  lineLength = "100",
                  startAtLine = "1",
                  useFieldWidth = "false",
                  delimiter = "|",
                  messagesPerTransaction = "10",
                  transactionDelay = "5000",
                  publishSubjectName = "SUB.PUB005.FT1",
                  endPublishSubject = "ENDPUB.PUB005.FT1",
                  useFilePolling = "true", pollInterval="5000",
                  removeAfterProcess = "false",
                  confirmationSubject = "ECM.PUB005.FT1",
                  totalConfirmationSubscribers = "1",
                  transferType = "RecordModeECM"
                }
MESSAGE_FIELDS = {
messageItem = { location="1", label="PartNo", type="STRING" },
messageItem = { location="2", label="Desc", type="STRING" },
messageItem = { location="3", label="Manufacturer", type="STRING" },
},
messageItem = { location="4", label="Model", type="STRING" },
messageItem = { location="5", label="Quantity", type="INTEGER" },
messageItem = { location="6", label="Price", type="FLOAT",
precision="7,2" },
messageItem = { location="7", label="Warranty", type="STRING",
constant="true", value="1 Year" }
}

```

*Example 6 Block Mode Publisher - Simple File Transfer*

You might want to publish a file without regard to its file structure. To do this, specify that the file is to be published, or subscribed to, in Block Mode SFT.

Configuration file name – PUB006, input file – FT2.

The FileType section of the configuration file:

```
[FileType]
FILE_OPTIONS = { filePrefix="FT2",
                  dataSetType = "SEQ",
                  isBinary = "false",
                  lineLength = "100",
                  startAtLine = "1",
                  messagesPerTransaction = "2",
                  transactionDelay="1000",
                  publishSubjectName = "SUB.PUB006.FT2",
                  endPublishSubject = "ENDPUB.PUB006.FT2",
                  useFilePolling = "true", pollInterval="5000",
                  removeAfterProcess = "false",
                  blockTransferSize = "128000",
                  transferType = "BlockModeSFT"
                }
```

*Example 7 Block Mode Publisher with ECM*

You might want to publish a file without regard to its file structure and at the same time use ECM. To do this, specify that the file is to be published, or subscribed to, in Block Mode ECM.

Configuration file name – PUB007, input file – FT2.

The FileType section of the configuration file:

```
[FileType]
FILE_OPTIONS = { filePrefix="FT2",
                  dataSetType = "SEQ",
                  isBinary = "false",
                  lineLength = "100",
                  startAtLine = "1",
                  messagesPerTransaction = "2",
                  transactionDelay="2000",
                  publishSubjectName = "SUB.PUB007.FT2",
                  useFilePolling = "true", pollInterval="5000",
                  removeAfterProcess = "false",
                  blockTransferSize = "128000",
                  ECMSubscriberName = "ECM.PUB007.FT2",
                  transferType = "BlockModeECM"
                }
```

### Example 8 Nesting of Mapped Messages

When nesting map messages, child messages are nested inside a stream message. This allows the nested map messages to be retrieved in the order in which they were inserted from inside the stream message without extra effort. In addition, at each transaction boundary, each map message identifies itself with an internal name-value pair generated by FilePublisher. This internal name-value pair appears as follows:

```
aXxXCONSTRAINTXxXa={String:dytsth}
```

where `dytsth` is the name defined for the container name in the `CONSTRAINT`.

Sample `MESSAGE_FIELDS` definitions:

```
MESSAGE_FIELDS = { CONSTRAINT = {POSITION = "0", length = "1",
                                value = "H", startnewmessage = "true",
                                containername = "dytsth" },
  MESSAGEITEM = { position="0", length = "1",
                  label="hdrRec", type="STRING" },
  MESSAGEITEM = { position="1", length = "11",
                  label="hPartNo", type="STRING" },
  MESSAGEITEM = { position="12",length = "15",
                  label="hDesc", type="STRING" },
  MESSAGEITEM = { position="27",length = "11",
                  label="hManufacturer", type="STRING" },
  MESSAGEITEM = { position="38",length = "12",
                  label="hModel", type="STRING" } }

MESSAGE_FIELDS = { CONSTRAINT = {POSITION = "0", length = "1",
                                value = "D", startnewmessage = "false",
                                containername = "dytstd" },
  MESSAGEITEM = { position="0", length = "1",
                  label="dtlRec", type="STRING" },
  MESSAGEITEM = { position="1", length = "11",
                  label="PartNo", type="STRING" },
  MESSAGEITEM = { position="12",length = "15",
                  label="Desc", type="STRING" },
  MESSAGEITEM = { position="27",length = "11",
                  label="Manufacturer", type="STRING" },
  MESSAGEITEM = { position="38",length = "12",
                  label="Model", type="STRING" }
}
```

Sample input data file:

```
Hpartnumber1description1111manufact1111modelnum1111
Dpartnumber1description1111manufact1111modelnum1111
Hpartnumber2description2222manufact2222modelnum2222
Dpartnumber2descriptionXXXXmanufact2222modelnum2222
Dpartnumber2description2222manufact2222modelnum2222
Hpartnumber3description3333manufact3333modelnum3333
Dpartnumber3description3333manufact3333modelnum3333
```

The following four messages would be generated and sent by FilePublisher:

```

StreamMessage={ Header={
JMSMessageID={ID:EMS-SERVER.7B84B0D3513102:25}
JMSDestination={Queue[DY.MRECFM.FILE.BACKUP]} JMSReplyTo={null}
JMSDeliveryMode={PERSISTENT} JMSRedelivered={false}
JMSCorrelationID={null} JMSType={null} JMSTimestamp={Thu Dec 03
10:03:43 EST 2009} JMSExpiration={0} JMSPriority={4} } Properties={
JMS_TIBCO_MSG_EXT={Boolean:true} } Fields={ {MapMsg:{
hDesc={String:description1111} hModel={String:modelnum1111}
aXxXCONSTRAINTXxXa={String:dytsth} hPartNo={String:partnumber1}
hManufacturer={String:manufact111} hdrRec={String:H} }} {MapMsg:{
Manufacturer={String:manufact111} Model={String:modelnum1111}
aXxXCONSTRAINTXxXa={String:dytstd} PartNo={String:partnumber1}
dtlRec={String:D} Desc={String:description1111} }} } }

StreamMessage={ Header={
JMSMessageID={ID:EMS-SERVER.7B84B0D3513102:26}
JMSDestination={Queue[DY.MRECFM.FILE.BACKUP]} JMSReplyTo={null}
JMSDeliveryMode={PERSISTENT} JMSRedelivered={false}
JMSCorrelationID={null} JMSType={null} JMSTimestamp={Thu Dec 03
10:03:44 EST 2009} JMSExpiration={0} JMSPriority={4} } Properties={
JMS_TIBCO_MSG_EXT={Boolean:true} } Fields={ {MapMsg:{
hDesc={String:description2222} hModel={String:modelnum2222}
aXxXCONSTRAINTXxXa={String:dytsth} hPartNo={String:partnumber2}
hManufacturer={String:manufact222} hdrRec={String:H} }} {MapMsg:{
Manufacturer={String:manufact222} Model={String:modelnum2222}
aXxXCONSTRAINTXxXa={String:dytstd} PartNo={String:partnumber2}
dtlRec={String:D} Desc={String:descriptionXXXX} }} {MapMsg:{
Manufacturer={String:manufact222} Model={String:modelnum2222}
aXxXCONSTRAINTXxXa={String:dytstd} PartNo={String:partnumber2}
dtlRec={String:D} Desc={String:description2222} }} } }

StreamMessage={ Header={
JMSMessageID={ID:EMS-SERVER.7B84B0D3513102:27}
JMSDestination={Queue[DY.MRECFM.FILE.BACKUP]} JMSReplyTo={null}
JMSDeliveryMode={PERSISTENT} JMSRedelivered={false}
JMSCorrelationID={null} JMSType={null} JMSTimestamp={Thu Dec 03
10:03:44 EST 2009} JMSExpiration={0} JMSPriority={4} } Properties={
JMS_TIBCO_MSG_EXT={Boolean:true} } Fields={ {MapMsg:{
hDesc={String:description3333} hModel={String:modelnum3333}
aXxXCONSTRAINTXxXa={String:dytsth} hPartNo={String:partnumber3}
hManufacturer={String:manufact333} hdrRec={String:H} }} {MapMsg:{
Manufacturer={String:manufact333} Model={String:modelnum3333}
aXxXCONSTRAINTXxXa={String:dytstd} PartNo={String:partnumber3}
dtlRec={String:D} Desc={String:description3333} }} } }

MapMessage={ Header={
JMSMessageID={ID:EMS-SERVER.7B84B0D3513102:28}
JMSDestination={Queue[DY.MRECFM.FILE.BACKUP]} JMSReplyTo={null}
JMSDeliveryMode={PERSISTENT} JMSRedelivered={false}
JMSCorrelationID={null} JMSType={null} JMSTimestamp={Thu Dec 03
10:03:44 EST 2009} JMSExpiration={0} JMSPriority={4} } Properties={
} Fields={ HOSTNAME={String:IBMI-V5R4.NA.TIBCO.COM}
RecordNumber={Integer:0} filename={String:DY2A} eof={Boolean:true}
numofretries={Integer:1} NumberOfMessages={Integer:3}
trackingid={String:Kq2@@D@@G1PVoe@56azzxkDkzzw}
version={String:5.0} USERID={String:} } }

```

## FileSubscriber Configuration

This section describes the FileSubscriber configuration file. The sections are listed in the order in which the sections appear in the file. The elements that comprise each section are listed in alphabetical order. In addition, sample elements are provided at the end of the section.

### Trace Section

You can use the Trace section to specify the trace logging behavior of FileSubscriber. This is the first section in the configuration file. The elements specified in this section are listed in the following table; all are required (except those used at the request of TIBCO Support).

Element	Description
FILE_COUNT	Specifies how many log files to keep. Each saved old log file name has a number (from 1 to the FILE_COUNT minus 1) suffixed to it. For example, if FILE_NAME is specified as FPLOG, and FILE_COUNT is set to 5, then there are four saved log files, named FPLOG1 through FPLOG4, with FPLOG4 being the earliest (or oldest) file.
FILE_LIMIT	Specifies the size of the log file in bytes. When this limit is exceeded, the current log file is saved, and a new log file is created.
FILE_NAME	Identifies the log file to be used for trace messages. This is a source physical file. If the file does not exist, then FilePublisher creates one. The default record length is 266 bytes. Default trace log files, PUBLOG and SUBLOG, are available in TIBFASMPRV or TIBFASMPER for your use. If you require a larger record length, you can create your own file; for example: <code>CRTSRCPF FILE(TIBFALIB/TIBLOGNEW) RCDLEN(1000)</code>
PRINT_STDOUT	Specifies whether to send the trace log messages to standard output (STDOUT) of the job in addition to the FILE_NAME specified when set to true. If set to false, trace messages are written only to a log file. If the adapter jobs are submitted to batch, STDOUT appears in the QPRINT printer file. If QSHELL is used to submit the adapter jobs, STDOUT appears on screen.

Element	Description
TRACE_EMS_ EPM_ERROR_ MSGs	<p>Specifies if JMS tracing have to be enabled for ERROR or EPM messages when using EMS. The valid values are <code>none</code>, <code>epm</code>, <code>error</code>, <code>both</code>. The default value is <code>none</code>.</p> <p>A value of <code>EPM</code> turns on JMS Message body tracing for all EPM messages sent by the adapter to the <code>EEM_DESTINATION</code>.</p> <p>A value of <code>error</code> turns on JMS Message body tracing for all ERROR messages sent by the adapter to the <code>ERROR_DESTINATION</code>.</p> <p>A value of <code>both</code> turns on JMS Message body tracing for all EPM and ERROR messages sent by the adapter.</p>
TRACE_EMS_ HEARTBEAT_ MSGs	<p>Enables JMS tracing for HEATBEAT messages when using EMS. The valid values are <code>true</code>, <code>false</code> (default).</p> <p>A value of <code>true</code> turns on JMS Message body tracing for all Heartbeat messages sent by the adapter.</p>
TRACE_HEAP	Used for diagnostic tracing in cases of memory exhaustion. Only used at the request of TIBCO Support. Valid value: <code>true</code>

Element	Description
TRACE_LEVEL	<p>The kind of information that FileSubscriber logs. FileSubscriber generates trace messages according to a trace level (1, 2, or 3) that you specify with this element.</p> <ul style="list-style-type: none"> <li>• <b>Trace level 1</b> generates the following session-level messages: <ul style="list-style-type: none"> <li>— Initialization of TIBCO EMS/Rendezvous, including the version number</li> <li>— TIBCO EMS/Rendezvous sessions created</li> <li>— Generate file trigger-subjects set for each file type</li> </ul> </li> <li>• <b>Trace level 2</b> generates level 1 messages plus all the following values loaded from the configuration file: <ul style="list-style-type: none"> <li>— All values loaded from the Options section of the configuration file</li> <li>— All values loaded from the FileType sections of the configuration file</li> </ul> </li> <li>• <b>Trace level 3</b> generates level 1 and 2 messages plus the following field-related messages: <ul style="list-style-type: none"> <li>— Field processing information (label type, and so on.)</li> <li>— Certified Messaging publisher information (name, sequence number, and so on.)</li> <li>— Message buffer reallocation</li> <li>— All key I/O Open/Close actions against S/390 files.</li> <li>— Traces flow through key publishing modules.</li> <li>— Traces flow of re-Publishing requests.</li> </ul> </li> <li>• <b>Trace level 4</b> for detailed tracing. Usually used only at the request of TIBCO Support. <ul style="list-style-type: none"> <li>— Traces detailed I/O calls for reading/writing data.</li> <li>— Traces detailed processing of Container fields</li> </ul> </li> <li>• <b>Trace level 5</b> for detailed tracing. Usually used only at the request of TIBCO Support. <ul style="list-style-type: none"> <li>— Traces all TIBCO RV or EMS received message callbacks</li> </ul> </li> <li>• <b>Trace Level 6</b> generates level 1 through 5 messages. Usually used only at the request of TIBCO Support. <ul style="list-style-type: none"> <li>— Traces all detailed timer callbacks used for handling ECM admin messages, re-publish messages, file lock retry, heartbeat messages</li> </ul> </li> </ul>

Element	Description
TRACE_LEVEL (Contd.)	<ul style="list-style-type: none"><li>• <b>Trace Level 10</b> generates Level 1 through 6 messages. Usually used only at the request of TIBCO Support.<ul style="list-style-type: none"><li>— Detailed trace of all RV/EMS message allocates and frees</li><li>— Detailed trace of Heap usage during message processing</li></ul></li><li>• <b>Trace Level 15</b> generates Level 1 through 10 messages. Usually used only at the request of TIBCO Support.<ul style="list-style-type: none"><li>— Detailed message content trace (traces the actual contents of the messages, up to 32K bytes)</li></ul></li></ul>
TRACE_SWITCHES	Special trace switches used by TIBCO support. Do not set this unless explicitly told to by TIBCO support. Valid values are an 8 byte text string. The default value is NNNNNNNN.
UNIT_TRACE	Allows non-SMS (System-Managed Storage) sites to specify the associated UNIT parameter to be used with the VOLSER for trace (.LOG) files. See VOLSER_TRACE, below.
VOLSER_TRACE	Allows non-SMS (System-Managed Storage) sites to specify a specific MVS volume (or set of volumes) to be used for storing trace (.LOG) files. Must be used with the UNIT_TRACE element. See UNIT_TRACE, above.

Options Section

You can use the Options section for the following options:

- Establish the type of TIBCO EMS or Rendezvous session to be used by FileSubscriber for subscribing
- Name the adapter instance
- Control heartbeat messages
- Define certain global elements that apply to all file types defined in the configuration file.

You can override these global elements in the FileType section.



Element	Description
ADAPTER_NAME	<p>Uniquely identifies the adapter instance. This element is:</p> <ul style="list-style-type: none"> <li>Used to set the terminate subject or destination (for EMS) to <code>_FILEADAPTER.&lt;adapter name&gt;.TERMINATE</code>. Sending a message to this subject either stops FileSubscriber or FilePublisher depending on the value of <code>&lt;ADAPTER_NAME&gt;</code>.</li> <li>Used to set the heartbeat subject or destination (for EMS) to <code>_FILEADAPTER.&lt;adapter name&gt;.HEARTBEAT</code>. This is used to send heartbeat messages, but does not mean that heartbeats are published. To publish (or send) heartbeats see <code>PUBLISH_HEARTBEAT</code>.</li> </ul> <p>For terminate messages, the default destination type is <code>QUEUE</code>. For heartbeat messages, the default destination type is <code>TOPIC</code>.</p>
CONTINUE_ON_CONFIG_ERROR	<p>Specifies whether to abend the adapter when it first encounters an initialization error for any FileType section. If set to <code>true</code> and, if a configuration error is found for a file, generate a message to the log and go on to the next file. If set to <code>false</code>, abend the adapter. Valid values: <code>true</code>, <code>false</code> (default).</p>
DELETE_SUB_PRG_FILES	<p>Indicates that the progress file is to be deleted at end of file. Valid values: <code>true</code>, <code>false</code> (default).</p>
EEM_DESTINATION	<p>Identifies the destination for messages that the adapter passes to TIBCO BusinessEvents. The value is the destination to which the event messages is routed. For EMS, the default value is <code>QUEUE</code>.</p>
EEM_SUBJECT	<p>Identifies the subject for messages that the adapter passes to TIBCO BusinessEvents. The value is the destination to which the event messages is routed.</p>
EMS_SESSION	<p>Specifies this element along with the following required parameters establishes reliable mode publishing:</p> <ul style="list-style-type: none"> <li><code>providerURL</code> — The URL of the TIBCO EMS server</li> <li><code>name</code> — Unique name for the connection (optional)</li> <li><code>username</code> — The user name used to connect to the EMS server</li> <li><code>password</code> — The password for the user.</li> </ul>

Element	Description
EMS_SESSION	<p>Specifies this element along with the following required parameters establishes reliable mode publishing:</p> <ul style="list-style-type: none"> <li>providerURL — The URL of the TIBCO EMS server</li> <li>name — Unique name for the connection (optional)</li> <li>username — The user name used to connect to the EMS server</li> <li>password — The password for the user.</li> </ul>
EOL_ON_SUBSCRIPTION	<p>Specifies the character that is used as a delimiter between messages and for terminating records. When using hexadecimal, literals can be specified in both lower and uppercase characters.</p> <p>If the required delimiter is the character 'G', the parameter value specified have to be 'C7', which is the EBCDIC hex value of 'G'.</p>
ERROR_EXIT_CC	<p>Specifies the return code that FileSubscriber returns when exiting with an error condition. Specifies MVS condition code at exit. Valid values: 4, 8 (default).</p>
ERROR_DESTINATION	<p>Specifies where error messages publish to. If this parameter is not blank, all traceErr and FatalErr messages are copied and sent to that destination. Valid values: blank or a string. The default value is blank "".</p> <p><b>Note:</b> Specify this field at the end of the Options section. Since the Rendezvous transport or EMS connection is not created until the end of the Options section, any error that occurs can't be published.</p>
ERROR_SUBJECT	<p>Specifies where the error message publish to. If this is not blank, all traceErr and FatalErr messages are copied and sent to that subject. Valid values: blank, string. The default value is blank.</p>
HEARTBEAT_FILE_INFO	<p>Publish additional file type information in the heartbeat message. For example, file prefix, file extension, and number of messages published are included. Valid values: true (default), false.</p>
HEARTBEAT_TIME	<p>Specifies the interval between heartbeat messages.</p> <p>The value must be greater than or equal to 100 milliseconds.</p> <p>The default value is 60000 milliseconds, or 60 seconds.</p>

Element	Description
HOST_CODEPAGE	Works with the codepage support in TIBCO Rendezvous 7.1 and above or EMS 4.2 and above. Specifies the translation table to use on the IBM i mainframe side. Defaults to the CCSID of the job in which the Publisher or Subscriber is running and which is usually tied to the profile of the user who initiated the job.
NETWORK_CODEPAGE	Works with the codepage support in TIBCO Rendezvous 7.1 and above/EMS 4.2 and above. Specifies the expected codepage sent by any remote subscribers.
OUTPUT_LIBRARYDATASET	<p>Specifies the default output library data set to be used for all files any file type. This is suffixed appropriately for the actual output data sets.</p> <p>You must identify which data set contains the files that you want to create. You can use the FILE_OPTIONS section to identify a default output data set's higher level qualifier at a global level.</p> <p>In addition, you can specify an output data set for each file type. You must identify which data set contains the files that you want to create. You can use the Options section to identify a default output data set's higher level qualifier at a global level.</p> <p>You can also specify an output data set for each file type. See <a href="#">Configuring the FILE_OPTIONS Element on page 92</a></p>
PRINT_FILE_OPTIONS	Prints all configuration information at the adapter startup. Valid values: <code>true</code> (default), <code>false</code> .
PROGRESS_DATASET	<p>Use this element to group all Progress (.PRG) files under a common high-level qualifier. Use the following parameter to specify an HLQ that can be prepended to all .PRG files generated by Adapter for Files OS/390:</p> <p><code>hlq.qualifier</code></p> <p>where <i>qualifier</i> is a user-defined string.</p>
PUBLISH_HEARTBEAT	<p>Enables publishing of heartbeat messages. Valid values: <code>true</code>, <code>false</code> (default). For EMS, heartbeat messages are sent to EMS TOPIC by default.</p> <p>For EMS, heartbeat messages are sent to EMS TOPIC by default.</p>

Element	Description
QUEUE_LIMIT	<p>Limits the number of data blocks that the publisher can asynchronously send to prevent over-consumption of memory. This element only supports implementations using SFT (Simple File Transfer) and ECM (Explicit Confirmation Mode).</p> <p>The value is one or more digits specifying the maximum depth of the RV Listen queue for incoming data block messages. The minimum value is 6. The default value is 0.</p> <p>The recommended value when Subscriber is receiving messages at a rate of 1 msg/sec is 80.</p> <p>This release of the adapter is tuned to handle message sizes 128K to 512. If you are using message sizes less than 128K, the subscriber performance might be affected.</p> <p><b>Warning:</b> QUEUE_LIMIT only can be used in ECM or RVCM transport mode. The data might be lost if you use it in RV transport mode.</p>
RV_SESSION	<p>If you include this element, do not include the RVCM_SESSION element.</p> <p>Specifying this element along with the following required parameters establishes a reliable mode of publishing.</p> <ul style="list-style-type: none"> <li>• <code>name</code> — Unique alphanumeric name identifying the TIBCO Rendezvous session</li> <li>• <code>service</code> — Service group for this session</li> <li>• <code>network</code> — Network to initialize a TIBCO Rendezvous session</li> <li>• <code>daemon</code> — Name of the TIBCO Rendezvous daemon for this session</li> </ul>

Element	Description
RVCM_SESSION	<p data-bbox="539 196 1229 256">If you include this element, do not include the RV_SESSION element.</p> <p data-bbox="539 274 1279 369">Specify this element establishes a Certified Messaging session. Include the same parameters as listed for RV_SESSION, plus the following options:</p> <ul data-bbox="539 387 1315 852" style="list-style-type: none"> <li data-bbox="539 387 1282 456">• <code>ledgerFile</code> — Name of the file-based ledger for Certified Messaging. This file is created in the Integrated File System.</li> <li data-bbox="539 473 1315 630">• <code>requireOldMessages</code> — Indicates whether a persistent correspondent requires delivery of messages sent to a previous session with the same name for which delivery was not confirmed. Setting this parameter to <code>true</code> enables delivery of old unacknowledged messages; setting it to <code>false</code> does not.</li> <li data-bbox="539 647 1290 743">• <code>defaultTimeLimit</code> — Sets the default message time limit for all outbound certified messages. The time is specified in seconds and the default value is 60.</li> <li data-bbox="539 760 1315 852">• <code>syncLedger</code> — If you want to use a synchronous ledger file, set this parameter to <code>true</code>. The default value is <code>false</code>, meaning an asynchronous ledger file is used.</li> </ul>

Element	Description
SECURITY_CHECK_FILE	<p>If set, this parameter contains the name of the mapping file to use to map incoming user ids into userids on the local system. Once the mapping is performed, a RACF CHECK is performed on any user-id for an incoming File being subscribed to, to ensure that the user has the authority to update the associated z/OS file.</p> <p>Here is the file layout for security file</p> <ul style="list-style-type: none"><li>• nodeid max 65 chars</li><li>• in_uid max 65 chars</li><li>• mapped_uid max 65 chars</li></ul> <p>All lines begin with '--' are comment line</p> <p><b>Note:</b> The FileAdapter loadlib must be “APF authorized” if you intend to activate the RACF check function. Please contact your system programmer for this procedure.</p> <p>Here is an example:</p> <pre>-- This is a security file for FileSubscriber --Purpose: To map the userid's coming from other --platforms to the ones on z/OS system. --Comments in this file start with '--'. --Node      Publisher UserId      Subscriber Mapped --                               UserId BATMAN-DT    batlab              BATMAN BATMAN-DT    batlab              BATMAN2 MARS         BATMAN              BATMAN MARS         BATMAN2             BATMAN2 BATMAN-DT    ROBIN               BATMAN BATMAN-DT    ROBIN               BATMAN2</pre>
TERMINATE_ON_RV_SEND_FAIL	<p>Specifies the adapter must abend if it cannot successfully call the API tibrvsend. This flag is only applicable to record mode processing. If the file is in block mode, the flag is ignored and the adapter is abended. Upon recovery, the adapter re-sync to the last check point. In the case of standard block mode, the entire file is retransmitted. In ECM, retransmission begin with the last block that was successfully acknowledged.</p> <p>Valid values: true, false. The default value is false.</p>

Element	Description
UNIT_CWK	<p>For implementations that do not use SMS (System-Managed Storage), you can use this element to specify a UNIT parameter to be used with the VOLSER for work (.CWK) files. (See VOLSER_CWK.)</p> <p>UNIT_CWK = "unit"</p> <p>If you need assistance determining the correct UNIT value to use, contact your MVS systems programmer.</p>
UNIT_GDG	<p>For implementations that do not use SMS (System-Managed Storage), you can use this element to specify a UNIT parameter to be used with the VOLSER for GDG output files. (See VOLSER_GDG.)</p> <p>UNIT_GDG = "unit"</p> <p>If you need assistance determining the correct UNIT value to use, contact your MVS systems programmer.</p>
UNIT_OUTPUT	<p>For implementations that do not use SMS (System-Managed Storage), you can use this element to specify a UNIT parameter to be used with the VOLSER for SEQ (sequential) output files. (See VOLSER_OUTPUT.)</p> <p>UNIT_OUTPUT = "unit"</p> <p>If you need assistance determining the correct UNIT value to use, contact your MVS systems programmer.</p>
UNIT_PRG	<p>For implementations that do not use SMS (System-Managed Storage), you can use this element to specify a UNIT parameter to be used with the VOLSER for Progress (.PRG) files. (See VOLSER_PRG.)</p> <p>UNIT_PRG = "unit"</p> <p>If you need assistance determining the correct UNIT value to use, contact your MVS systems programmer.</p>

Element	Description
VOLSER_CWK	<p>For implementations that do not use SMS (System-Managed Storage), you can use this element to specify a specific MVS volume (or set of volumes) to be used for storing File Adapter work (.CWK) files. Must be used with UNIT_PRG.</p> <p>VOLSER_CWK = "<i>volume</i>" VOLSER_CWK = "<i>volume,volume</i>"</p> <p>where <i>volume</i> is a valid setting for MVS volume.</p> <p>The maximum number of VOLSER values you can set is 59. If you need assistance determining the correct VOLSER names to use, contact your MVS systems programmer.</p>
VOLSER_GDG	<p>For implementations that do not use SMS (System-Managed Storage), you can use this element to specify a specific MVS volume (or set of volumes) to be used for storing File Adapter GDG output files. Must be used with UNIT_GDG.</p> <p>VOLSER_GDG = "<i>volume</i>" VOLSER_GDG = "<i>volume,volume</i>"</p> <p>where <i>volume</i> is a valid setting for MVS volume.</p> <p>The maximum number of VOLSER values you can set is 59. If you need assistance determining the correct VOLSER names to use, contact your MVS systems programmer.</p>
VOLSER_OUTPUT	<p>For implementations that do not use SMS (System-Managed Storage), you can use this element to specify a specific MVS volume (or set of volumes) to be used for storing File Adapter SEQ (sequential) output files. Must be used with UNIT_PRG.</p> <p>VOLSER_OUTPUT = "<i>volume</i>" VOLSER_OUTPUT = "<i>volume,volume</i>"</p> <p>where <i>volume</i> is a valid setting for MVS volume.</p> <p>The maximum number of VOLSER values you can set is 59. If you need assistance determining the correct VOLSER values to use, contact your MVS systems programmer.</p>



Element	Description
VOLSER_PRG	<p>For implementations that do not use SMS (System-Managed Storage), you can use this element to specify a specific MVS volume (or set of volumes) to be used for storing progress (.PRG) files. Must be used with UNIT_PRG.</p> <pre>VOLSER_PRG = "volume" VOLSER_PRG = "volume,volume"</pre> <p>where <i>volume</i> is a valid setting for MVS volume.</p> <p>The maximum number of VOLSER values you can set is 59. If you need assistance determining the correct VOLSER names to use, contact your MVS systems programmer.</p>
WORKFILE_DATASET	<p>You can use this element to group all File Adapter Work (.CWK) files under a common high-level qualifier (HQL). Use the following parameter to specify an HQL that can be prepended to all .CWK files generated by Adapter for Files OS/390:</p> <pre>hql .qualifier</pre> <p>where <i>qualifier</i> is a user-defined string.</p>
WRITE_TO_SYSLOG	<p>If <code>true</code>, major checkpoint log messages are sent to QSYSOPR *MSGQ z/OS errors are written to SYSLOG. Each message includes date and time information. The format is standard IBM-style message format such as an <code>SXFnnnnn</code> header followed by message text. Valid values: <code>true</code>, <code>false</code>. The default value is <code>false</code>.</p>

## FileType Section

This section describes the file that is to be written. It consists of two elements:

- `FILE_OPTIONS` (See [Configuring the FILE\\_OPTIONS Element on page 92.](#))
- `FILE_LINE` (See [Configuring the FILE\\_LINE Element on page 103.](#))

### Configuring the FILE\_OPTIONS Element

This section describes the parameters in the `FILE_OPTIONS` element. A configuration file can have multiple `[FILE_OPTIONS]` definitions in a single file.

Parameter	Description
<code>filePrefix</code> (required)	Used to construct the name of the file that is written to the output library data set. Specify a value of up to 5 characters. Also used for deriving the names of the progress file and the work file.
<code>subscribeSubjectName</code> (required)	Specifies the subject name to subscribe to for this file type.
<code>subscribeDestinationName</code>	Specifies the destination name to subscribe to for this file type.
<code>subscribeDestinationType</code>	Specifies the type of destination to subscribe to for this file type. Valid values: <code>TOPIC</code> , <code>QUEUE</code> . The default value is <code>TOPIC</code> .
<code>JMS_TIBCO_MSG_TRACE</code>	Specifies whether the entire message, or only the header of the message, is traced. Valid values: <code>body</code> , <code>null</code> .

Parameter	Description
<b>Publishing Correlation Identifier</b>	
useTrackingId	<p>Specifies that a trackingId message is to be sent by the subscriber. This message is created after the subscriber finishes writing the file and receiving the EOF acknowledgement from the publisher. Valid values: true, false (default). If true, the following additional fields are attached to the message published to trackingIdSubject:</p> <ul style="list-style-type: none"> <li>• AdapterName</li> <li>• FileName</li> <li>• FileExtension (N/A for IBM i NT only, null if z/OS)</li> <li>• FilePrefix (FileType Prefix)</li> <li>• FileSize</li> <li>• fileTransferDuration</li> <li>• DateTime</li> <li>• TransferMode (BLOCK or RECORD)</li> <li>• PubLocalResult (Publisher's Local result = 0 or 8)</li> <li>• Result (0 or 4 or 8 = OK or Fail) Overall transfer result from Subscriber</li> <li>• status (Succeeded or Failed)</li> <li>• HostName</li> <li>• Subject</li> <li>• trackingId</li> </ul>
trackingIdSubject	Name of the subject on which messages containing the trackingId information are published.
trackingIdDestination Name	Name of the destination to which messages containing the trackingId information are published.
<b>Data Set Type</b>	
datasetType	<p>Specifies that only sequentially organized data is allowed. Specify SEQ. Specifies the type of data set for this file type. This can be SEQ for sequential data sets, GDG for Generation Dataset Groups, PDS for partitioned datasets, or VSAM for Virtual Storage Access Method.</p>

Parameter	Description
<b>Output File Allocation</b>	
<code>outputLibrary</code>	<p>Name of the output library data set for this file type. The default value is what is specified in the <code>OUTPUT_LIBRARYDATASET</code> element of the Options section. This overrides the value specified in <code>OUTPUT_LIBRARY</code>.</p> <p><b>Note:</b> VSAM and GDG file types must be preallocated before subscribing to them.</p>
<code>primaryAlloc</code>	<p>Primary allocation size for work and output data sets in cylinders (CYL).</p> <p>The default value is 2. This value is used only for sequential and GDG file types and for temporary work files.</p>
<code>secondaryAlloc</code>	<p>Specifies the secondary allocation space for work and output data sets in cylinders (CYL).</p> <p>The default value is 1. This value is used only for sequential and GDG file types and temporary work files.</p>
<code>blockSizeAlloc</code>	<p>Specifies the block size of the subscriber file. Setting this value is highly recommended.</p>
<code>lineLength</code>	<p>Maximum number of characters constituting a line in the output file. This parameter is valid only if <code>skipPadding</code> is <code>true</code>. Setting this value is highly recommended.</p> <p><b>Note:</b> <code>lineLength</code> cannot be larger than 32764 bytes.</p>
<code>useFixedRecordFile</code>	<p>The subscribing file is in fixed or variable length format. Valid values: <code>true</code> (default) for fixed-length records, <code>false</code> for variable-length records.</p>

Parameter	Description
<code>truncateRecords</code>	<p>Indicates how the subscriber behaves when the receiving data record length is different from the one specified by the <code>lineLength</code> parameter.</p> <p>Valid values:</p> <ul style="list-style-type: none"> <li>• <code>wrap</code>—wrap around the rest of the record to a new line.</li> <li>• <code>discard</code>—truncate the record and do not generate an error file</li> <li>• <code>error</code>—truncate the record and rename the working file (WK-prefixed fileCWK) to the error file (ERRnn-prefixed file.ERR) at EOF (default).</li> </ul>
<code>AppendDateTime</code>	<p>Only used for sequential data sets. If <code>true</code>, FileSubscriber appends the system time to the <code>filePrefix</code> parameter when constructing the name of the file to be written. That is, the generated output data set name has the format:</p> <p><code>outputDataset.Yyyyyy.MDmmdd.Thhmmss.filePrefix</code></p> <p>The default value is <code>false</code>.</p>

Parameter	Description
<b>Output File Creation</b>	
<code>autoGenerateFile</code>	<p>If <code>true</code> (default), enables automatic output file creation based on a timer (see the next parameter, <code>saveFileInterval</code>). If <code>false</code>, output files are not generated based on a timer. This value must be explicitly turned off if not required.</p> <p>If <code>autoGenerateFile</code> is set to <code>false</code>, then the <code>saveFileInterval</code> parameter and <code>generateFileOnNumberOfMessages</code> are ignored. When set to <code>false</code>, the file cannot be generated until an End-Of-File indication is received from the Publisher.</p>
<code>saveFileInterval</code>	<p>This is the interval of time, in seconds, that <code>FileSubscriber</code> uses to generate an output file. Valid only if <code>autoGenerateFile</code> is set to <code>true</code>.</p> <p>This parameter is used to periodically save received (staged) data to the target file.</p> <p>Note that the <code>saveFileInterval</code> and <code>generateFileOnNumberOfMessages</code> parameters are mutually exclusive.</p> <p>If the value is 0 (zero), the default value of 120 is used.</p> <p>A maximum value of 1800 is enforced.</p>
<code>generateFileOnNumberOfMessages</code>	<p>Generate an output file if the number of messages received since the generation of the last output file equals this integer value. The default value is 0 which means the parameter is not used.</p> <p>This parameter is used to cause the saving of received (staged) data to the target file whenever N number of messages (records) have arrived. The <code>saveFileInterval</code> and <code>generateFileOnNumberOfMessages</code> parameters are mutually exclusive.</p>
<code>noOfRetries</code>	<p>Number of times to try creating the output file. The default value is 0.</p>

Parameter	Description
<code>retryInterval</code>	<p>The amount of time, in milliseconds, that FileSubscriber waits before retrying when a <code>File Locked</code> condition is detected.</p> <p>If value of the <code>noOfRetries</code> parameter is greater than 0, then the value of the <code>retryInterval</code> parameter must be also greater than 0.</p> <p>The default value is 0.</p>
<code>appendToExistingFile</code>	<p>If <code>true</code>, specifies that if an output file already exists, the data received is appended to the existing file. Otherwise, FileSubscriber overwrites existing data. In the case of sequential dataset output, this option is effective only when <code>appendDateTime</code> is set to <code>false</code>.</p> <p><b>Warning:</b> If the subscriber is configured with <code>appendToExistingFile="false"</code> and if there is an I/O error in writing to the output file, then the adapter removes the output file first and then renames the work file to error file.</p> <p>As a result customers lose the existing output file even though the file transfer has failed. This is like deleting the existing load module even when linking failed. This behavior is observed only with SEQ files not with GDG.</p> <p>The default value is <code>false</code>.</p>
<code>exitOnFileSaveError</code>	<p>Specifies what the adapter does when it cannot save data to the target file because of insufficient space. If <code>true</code>, exit the adapter. Otherwise continue to accept data and write it to the 'temporary' subscriber file.</p> <p>Valid values: <code>true</code>, <code>false</code>. The default value is <code>true</code>. There are two variations of out-of-space situations:</p> <ol style="list-style-type: none"> <li>1. The volume is completely out of space, and no file can be created. An error message is written to the log indicating the file could not be created because of lack of space.</li> <li>2. The file was partially written, but no more extents can be allocated. In that case, the partial file is renamed to <code>filename.ERR</code> and cataloged. An error message is written to the log indicating the file could not be created because of lack of space.</li> </ol>
<code>fileSizeInitRec</code>	<p>Specifies the initial number of records, the value can be <code>*DEFAULT</code>, <code>*NOMAX</code>, and 1-2147483646.</p>

Parameter	Description
fileSizeIncrRec	Specifies the number of records to increment, the value can be *DEFAULT and 1-32767.
fileSizeIncrMax	Specifies the maximum number of increments, the value can be *DEFAULT and 1-32767.
forcePublishedFileName	<p>If this flag is set to true, the output file is named the same as the file being published. Specifically, the output file is named the same as the filename contained in the EOF message.</p> <p>For this parameter to function properly:</p> <ol style="list-style-type: none"> <li>1. the filePrefix must be specified.</li> <li>2. generateFileSubjectName must be specified and must match the endPublishSubject as specified in the Publisher's configuration file.</li> </ol> <p>Valid values: true, false. The default value is false.</p>
genFilePublishSubject	If the subject is defined, then the subscriber publishes a message using this subject to announce that a new file is created. It is published after the executeAfterProcess is performed.
genFilePublishDestination Name	<p>If this destination is defined, then the subscriber publishes a message using this destination to announce that a new file is created. It is published after the executeAfterProcess is performed.</p> <p><b>Note:</b> The destination type defaults to subscribeDestinationType.</p>
discardUncatalogedFiles	<p>Specifies if an incoming file request have to be saved, if the file it is referring to is uncataloged. Valid values:</p> <ul style="list-style-type: none"> <li>• none – create the file regardless of whether or not an existing file is cataloged (default).</li> <li>• append – the incoming file have to be discarded, if the FileType is marked for append, but there is no current dataset cataloged.</li> <li>• new – the incoming file have to be discarded if there is no existing cataloged file entry.</li> <li>• all – combines the attributes of both append and new.</li> </ul>



Parameter	Description
<b>Output Files Based on Trigger Message</b>	
<code>generateFileSubjectName</code> (required)	The subject name to subscribe to for generating the output file for this file type. If <code>autoGenerateFile</code> is set to <code>false</code> , then the <code>saveFileInterval</code> parameter and <code>generateFileOnNumberOfMessages</code> are ignored. When set to <code>false</code> , the file cannot be generated until an EOF indication is received from the Publisher.
<code>generateFileDestinationName</code> (required)	<p>The destination name to subscribe to for generating the output file for this file type.</p> <p><b>Note:</b> The destination type defaults to <code>subscribeDestinationType</code>.</p>
<code>generateFileFieldName</code>	<p>Most usages of <code>generateFileSubjectName</code> require that it match the <code>subscribeSubjectName</code>.</p> <p>The <code>generateFileFieldName</code> is used to act as a message differentiator, so that the message being sent to a Subscriber can be identified as being a “Generate File” trigger message. It is only required if the subscriber is using the <code>generateFileSubjectName</code> for the associated <code>FileType</code>.</p> <p>The default value is “filename”.</p>
<b>Certified Subscribing</b>	
<p>If you have specified that the session identified in the configuration file is a certified session, then you can identify those files that you want to be published in certified or in non-certified (Reliable) mode.</p> <p><b>Note:</b> If you specify a certified session, then all files are published in certified mode unless you specify <code>false</code> for the <code>isCertified</code> parameter.</p>	
<code>isCertified</code>	If the FileSubscriber session is not a certified session, this parameter is ignored. If the FileSubscriber session is specified as a certified session, then the default value for this parameter is <code>true</code> , which means that the file is subscribed to in certified mode. You can specify <code>false</code> if you want to have a specific file subscribed to in non-certified (reliable) mode.

Parameter	Description
Preprocessing and Postprocessing	
executeBeforeProcess	Causes FileSubscriber to execute a command or call a program submit a job to the internal reader for execution before generating an output file. See <a href="#">Pre-Processing and Post-Processing Files on page 125</a> .
executeAfterProcess	Causes FileSubscriber to execute a command or call a program submit a job to the internal reader for execution after generating an output file. See <a href="#">Pre-Processing and Post-Processing Files on page 125</a> .
Record Handling	
fileHeader	Specifies a header record to place at the beginning of the output file.
fileTrailer	Specifies a trailer record to place at the end of the output file.
skipPadding	<p>If included and set to <code>true</code>, specifies that data fields are to be delimited and not padded.</p> <p>In that case, FileSubscriber generates variable length fields in the record. The field position in the output record is determined by the <code>position</code> parameter in the <code>FILE_LINE</code> element.</p> <p>There is no default value for the parameter delimiter, therefore it is important to specify a value.</p> <p>If <code>false</code>, data fields are padded with characters specified by the <code>padCharacter</code> parameter and are fixed width.</p> <p>In that case, FileSubscriber generates fixed-length fields in the output record. The file position in the output record is determined by the <code>position</code> parameter in the <code>FILE_LINE</code> element, and the field length is determined by the <code>length</code> parameter.</p>
padCharacter	<p>If <code>skipPadding</code> is <code>false</code>, this alphanumeric character is used as the pad character.</p> <p>The default value is a blank character.</p>
padDirection	<p>Specifies which direction to pad the data field, <code>left</code> or <code>right</code>.</p> <p>The default value is <code>right</code>, which means that the pad characters are added to the right side of the field (that is, left-justified).</p>

Parameter	Description
delimiter	An alphanumeric character used to separate the fields in a line. This parameter is valid only if skipPadding is true. This parameter does not have a default value, which means that if this parameter is not specified, there is no separator between fields.
isBinary	When set to true, specifies that the file data is in binary format. Binary data includes zoned decimal, binary, packed decimal, and floating point data types.
Transfer Type	
transferType	<p>Specifies the data transfer mode (i.e, Record Mode or Block Mode) and whether to use ECM or non-ECM with the specified data transfer mode. Valid values:</p> <p>BlockModeECM – use Block Mode with ECM.</p> <p>BlockModeSFT – use Block Mode with non-ECM mode.</p> <p>RecordModeECM – use Record Mode ECM.</p> <p>RecordMode – use Record Mode with non-ECM mode</p> <p>For additional information, see <a href="#">Message Delivery Considerations on page 10</a>.</p>
blockTransferMode	The adapter writes the data to the file in blocks. The publisher determines the block size. Valid values: true, false (default).
useExplicitConfirmation	Use ECM mode. Valid values: true, false (default).
ECMSubscriberName	<p>Name of the corresponding ECM publisher in block mode. This parameter must have a valid value if ECM mode is selected. This parameter might appear more than once in the Publisher with a different value for each Subscriber.</p> <p><b>Warning:</b> You cannot specify both an ECMSubscriberName and a confirmationSubject.</p>
SubscriberName	<p>A new SubscriberName ID is required so that the EMS server can identify which consumer clientss requires which messages resent after a failure. This SubscriberName ID is created dynamically when the Subscriber is started.</p> <p>For example, SubscriberName="sub1", where sub1 is an identifier of your choice.</p>

Parameter	Description
confirmationSubject	<p>Confirmation subject name used by FileSubscriber to exchange confirmation messages in Record Mode.</p> <p><b>Warning:</b> You cannot specify both a confirmationSubject and an ECMSubscriberName.</p>
retransmissionDelayTicks	<p>Only used for ECM Block Mode. This parameter is used to co-ordinate the startup handshake between the Publisher and the Subscriber.</p> <p>For the Publisher, this parameter represents the number of times the transactionDelay timer must pop before checking for any unacknowledged Message Blocks (retransmissionDelayTicks * transactionDelay). The default value for the Publisher is 2 sec.</p> <p>Also, this represents the number of times the transactionDelayTimer pops before the Publisher attempts to re-synchronize with the Subscriber.</p> <p>For Subscriber, this parameter represents the number of seconds that FileSubscriber waits before trying to connect to the Publisher after a startup (retransmissionDelayTicks * 1000). The default value for the Subscriber is 10 sec.</p>
VSAM	
vsamFileMode	<p>Specifies how the VSAM file can be used: INSERT only, REPLACE only, or UPSERT (insert and replace).</p>
vsamLogFile	<p>File name to use if VSAM file logging is enabled. This is a sequential file used to log any changes made to the associated VSAM file. Any inserts or updates made to the VSAM file by the adapter are recorded to the log sequential file. The sequential file is always opened in append and binary mode.</p> <p><b>Note:</b> When this flag is specified, the vsamUseLog is required or the adapter throws a fatal error.</p>

Parameter	Description
vsamUseLog	<p>Determines whether logging is to be performed. Default value: stopOnFull, No. If set to No, no logging is performed. If set to StopOnFull, logging is performed. If the log file becomes full during operation, the adapter abends to prevent making changes to the VSAM file without an associated log record of the changes.</p> <p>It is the end user's responsibility to monitor the use of the log, and to periodically clear it to prevent the adapter from abending because of the log full conditions. If vsamUseLog enables logging, then any primaryAlloc= and secondaryAlloc= keywords are used to allocate space for the associated VSAM log file.</p>

## Configuring the FILE\_LINE Element

Use the FILE\_LINE element in the FileType section to format the output record. If output records are in a different format, you can include multiple FILE\_LINE elements in a single FileType section. The FILE\_LINE element contains parameters that are defined using the following tags. These tags are optional unless specified otherwise.

The FILE\_LINE element describes the format of the message to be published. It can be formed by using one or more of the following parameters.

Parameter	Description
field	Identifies the field. You can specify different tags inside this parameter. See <a href="#">Tags in the field parameter on page 103</a> .
constraint	Must be used in the case of multi-record format. See <a href="#">Tags in the constraint Parameter on page 105</a>

### Tags in the field parameter

The FILE\_LINE element's field parameter has the following tags.

Tag	Description
description	A concise description of the data field. FileSubscriber does not process this parameter; it is treated as a comment.

Tag	Description
fromMessage	A setting of true flags this field as being generated from a data item in the incoming TIBCO Rendezvous message. A setting of false means that this field is a constant field. The default value is false.
position	For delimited files, specifies the field position in the record, starting at 1. Otherwise, specifies the byte index in the record, starting at 0. Also see the skipPadding parameter for the effect of position in output records. Use position or fieldStart but not both.
fieldStart	Used for binary files to specify the start of a field. The first byte of an input or output record is "1." Use fieldStart or position but not both.
length	Field length. The default value is 1.
type	Data type of this field. Valid values: STRING, INTEGER, UNSIGNED_INTEGER, SHORT, UNSIGNED_SHORT, FLOAT, DOUBLE, BOOLEAN, and TIME. The default value is STRING. Valid values for EMS: STRING, INTEGER, SHORT, FLOAT, DOUBLE, BOOLEAN.
value	Value of the field if it is a constant field. Otherwise, it contains the name of the message item that holds the data for this field.
padCharacter	Use this tag to override the padCharacter specified in the FILE_OPTIONS section. If required, specify an alphanumeric character.
padDirection	Use this tag to override the padDirection specified in the FILE_OPTIONS section. If required, specify either left or right.
convertTo	The TIBCO Rendezvous numeric data types INTEGER, UNSIGNED_INTEGER, SHORT, UNSIGNED_SHORT, and FLOAT can be converted to PACKED, ZONED, BINARY, or Floating Point. STRING values that are in numeric format can also be converted to PACKED, ZONED, BINARY, or Floating Point output.
precision	Specifies the size of the field and the number of decimals. This tag is not used for COMP-1 or COMP-2 fields.



When you specify TIME as a data type, FilePublisher gets the current system time, and places it in the TIBCO message. When FileSubscriber receives the message, it converts the time to a string, which represents the system time of the publishing system.

Tags in the constraint Parameter

FileSubscriber supports multiple line formats from different sections of a TIBCO message. To support this, you have to include multiple FILE\_LINE elements in the FileType section. When multiple FILE\_LINE elements are used, a constraint parameter must be supplied for each FILE\_LINE element to indicate when to use that definition.

The FILE\_LINE element’s constraint parameter contains the following tags:

Tag	Description
containerName	The name of the TIBCO message that contains the fields that map to the FILE_LINE element.
lineLength	The length of a line in the output file for this line field definition. The default value is specified by the lineLength parameter in the FILE_OPTIONS section.

## Sample Configuration Elements

This section contains several configuration element examples that you can use as models.

### *Example 9 Record Mode Subscriber - Delimited Text File*

This example shows the FileType section of a configuration file for processing TIBCO Rendezvous messages for a delimited file.

Configuration file name – SUB001.

The FileType section of the configuration file:

```
[FileType]
FILE_OPTIONS = { filePrefix="FT1",
                  delimiter = "|",
                  skipPadding = "true",
                  isBinary = "false",
                  noOfRetries = "10", retryInterval = "5",
                  subscribeSubjectName = "SUB.PUB001.FT1",
                  autoGenerateFile = "true",
                  appendDateTime = "true",
                  saveFileInterval = "100",
                  transferType = "RecordMode"
                }

FILE_LINE = {
  field = { description = "Part Number", position = "1",
    fromMessage = "true", value = "PartNo", type = "STRING" },
  field = { description = "Part Description", position = "2",
    fromMessage = "true", value = "Desc", type = "STRING" },
  field = { description = "Manufacturer", position = "3",
    fromMessage = "true", value = "Manufacturer", type = "STRING" },
  field = { description = "Model", position = "4",
    fromMessage = "true", value = "Model", type = "STRING" },
  field = { description = "Quantity", position = "5",
    fromMessage = "true", value = "Quantity", type = "INTEGER" },
  field = { description = "Unit Price", position = "6",
    fromMessage = "true", value = "Price", type = "FLOAT" },
  field = { description = "Warranty Period in years", position = "7",
    fromMessage = "true", value = "Warranty", type = "STRING" }
}
```

Assuming that the file used in the first FilePublisher example is being subscribed to, FileSubscriber would create a file that would look as follows:

```
115-01-0500|MONITOR|SONY|VIEWSONIC|1|350.500000|1 Year
115-15-6542|CPU-PIII750MHZ|COMPAQ|PRESARIO|1|900.000000|1 Year
115-67-7356|HDD20GB|SEAGATE|ST500|1|276.600006|1 Year
115-34-8767|FDD54|HP|T24333|1|86.000000|1 Year
115-77-5555|CDRW|HP|T75668|1|350.000000|1 Year
115-78-4646|KEYBOARD|COMPAQ|EASYKEY101|1|32.000000|1 Year
115-88-4454|MOUSE|MICROSOFT|M323|1|26.000000|1 Year
115-36-2727|WINDOWS2000|MICROSOFT|MSW2000|1|400.000000|1 Year
```



*Example 10 Record Mode Subscriber - Fixed Length Text File*

This example shows the FileType section of a configuration file for processing TIBCO Rendezvous messages for a fixed length text file.

Configuration file name – SUB002.

The FileType section of the configuration file:

```
[FileType]
FILE_OPTIONS = { filePrefix="FT2",
                  dataSetType = "SEQ",
                  isBinary = "false",
                  lineLength = "100",
                  useFixedRecordFile = "true",
                  appendToExistingFile = "false",
                  truncateRecords = "error",
                  exitOnFileSaveError = "false",
                  noOfRetries = "10", retryInterval = "5",
                  isCertified = "false",
                  subscribeSubjectName = "SUB.PUB002.FT2",
                  generateFileSubjectName = "ENDPUB.PUB002.FT2",
                  generateFileFieldName = "filename",
                  autoGenerateFile = "false",
                  transferType = "RecordMode"
                }

FILE_LINE = {
  field = { description = "Part Number", fromMessage = "true",
            value = "PartNo", position="0", length = "11", type = "STRING" },
  field = { description = "Part Description", fromMessage = "true",
            value = "Desc", position="11", length = "15", type = "STRING" },
  field = { description = "Vendor", fromMessage = "true",
            value = "Vendor", position="26", length = "11", type = "STRING" },
  field = { description = "Model", fromMessage = "true",
            value = "Model", position="37", length = "12", type = "STRING" },
  field = { description = "Quantity", fromMessage = "true",
            value = "Quantity", position="49", length = "2", type = "INTEGER" },
  field = { description = "Unit Price", fromMessage = "true",
            value = "Price", position="51", length = "6", type = "FLOAT" },
  field = { description = "Warranty Period in years",
            fromMessage = "true", value = "Warranty", position="57",
            length = "10", type = "STRING" }
}
```

Assuming that the file used in the second FilePublisher example is being subscribed to, FileSubscriber would create a file that looks as follows:

```
115-01-0500MONITOR SONY VIEWSONIC 2 250.501 Year
115-15-6542CPU-PIII750MHZ COMPAQ PRESARIO 4 12.3001 Year
115-67-7356HDD20GB SEAGATE ST500 2515.0001 Year
115-34-8767FDD54 HP T24333 1 45.2201 Year
115-77-5555CDRW HP T57213 5 290.001 Year
115-78-4646KEYBOARD COMPAQ EASYKEY1 10121.201 Year
```

*Example 11 Record Mode Subscriber - Different Header and Detail Formats*

In FilePublisher example, a file with multiple record types was used as an example. This example shows how to configure the FileSubscriber configuration file to handle the TIBCO Rendezvous messages to process a multiple format file. The following steps could be used to set up the configuration file:

1. Use the FileType section to specify attributes of the file the subscriber creates.
2. Use a FILE\_LINE element with a containerName parameter to define the format of each output line to be written after retrieving data from the message. FileSubscriber creates an output record for each FILE\_LINE element.

For the example shown, FileSubscriber creates a header record for the "Recfm1\_Quote" container, then it creates three detail records from the "Recfm2\_Items" record.

3. Specify the field attributes for the output record by using the messageItem parameter.
  - For fixed-length files, identify the field by using both the position and length parameters.
  - For delimited files, specify only a delimiter.

Configuration file name – SUB003.

The FileType section of the configuration file:

```
[FileType]
FILE_OPTIONS = { filePrefix="FT3",
                  dataSetType = "SEQ",
                  isBinary = "false",
                  lineLength = "100",
                  useFixedRecordFile = "true",
                  appendToExistingFile = "false",
                  truncateRecords = "error",
                  exitOnFileSaveError = "false",
                  noOfRetries = "10", retryInterval = "5",
                  isCertified = "false",
                  subscribeSubjectName = "SUB.PUB003.FT3",
                  autoGenerateFile = "true",
                  saveFileInterval = "100",
                  transferType = "RecordMode"
                }

FILE_LINE = {
  constraint = { containerName = "Recfm1_Quote" },
  field = { description = "Record Identifier", fromMessage = "false",
            value = "QUOTE", position="0", length = "5", type = "STRING" },
  field = { description = "Quote Request Number",
            fromMessage = "true", value = "QuoteRequestNo", position="5",
            length = "5", type = "INTEGER" },
  field = { description = "Supplier Code", fromMessage = "true",
            value = "SupplierCode", position="10",length = "5", type = "STRING"
          }
}
```

```

FILE_LINE = {
  constraint = { containerName = "Recfm2_Items" },
  field = { description = "Record Identifier", fromMessage = "false",
    value = "ITEMS", position="0",length = "5", type = "STRING" },
  field = { description = "Part Number", fromMessage = "true",
    value = "PartNo", position="5", length = "11", type = "STRING" },
  field = { description = "Part Description", fromMessage = "true",
    value = "Desc", position="16",length = "15", type = "STRING" },
  field = { description = "Vendor", fromMessage = "true",
    value = "Vendor", position="31",length = "11", type = "STRING" },
  field = { description = "Model", fromMessage = "true",
    value = "Model", position="42",length = "12", type = "STRING" },
  field = { description = "Quantity", fromMessage = "true",
    value = "Quantity", position="54",length = "2", type = "INTEGER" },
  field = { description = "Unit Price", fromMessage = "true",
    value = "Price", position="56",length = "6", type = "FLOAT" },
  field = { description = "Warranty Period in years",
    fromMessage = "true", value = "Warranty", position="62",
    length = "10", type = "STRING" }
}

```

Assuming that the file used in the third FilePublisher example is being subscribed to, the file that FileSubscriber creates would look as follows:

```

QUOTE15004MC001
ITEMS115-01-0500MONITOR SONY VIEWSONIC 2 250.501 Year
ITEMS115-15-6542CPU-PIII750MHZ COMPAQ PRESARIO 4 12.3001 Year
ITEMS115-67-7356HDD20GB SEAGATE ST500 2515.0001 Year
QUOTE15004MC002
ITEMS115-34-8767FDD54 HP T24333 1 45.2201 Year
ITEMS115-77-5555CDRW HP T57213 5 290.001 Year
ITEMS115-78-4646KEYBOARD COMPAQ EASYKEY1 10121.201 Year

```

### Example 12 Record Mode Subscriber - Supporting Numeric Data Types

This examples shows how a FileSubscriber configuration file can be coded to support the numeric data types.

Configuration file name – SUB004.

The FileType section of the configuration file:

```

[FileType]
FILE_OPTIONS = { filePrefix="BIN2",
  dataSetType = "SEQ",
  isBinary = "true",
  lineLength = "95",
  useFixedRecordFile = "true",
  appendToExistingFile = "false",
  truncateRecords = "error",
  exitOnFileSaveError = "false",
  noOfRetries = "10",
  retryInterval = "5",
  isCertified = "false",
  subscribeSubjectName = "SUB.PUB004.BIN2",
  autoGenerateFile = "true",

```

```

        saveFileInterval = "100",
        transferType = "RecordMode"
    }

    FILE_LINE = {
    field = { description = "Character", fromMessage = "true",
    value = "Char", position="0", length="5", type="STRING" },
    field = { description = "Packed", fromMessage = "true",
    value = "Packed", position="5", type = "DOUBLE", precision="10,2",
    convertTo="PACKED" },
    field = { description = "zoned", fromMessage = "true",
    value = "Zoned", position="11", type = "DOUBLE", precision="10,2",
    convertTo="ZONED" },
    field = { description = "binary", fromMessage = "true",
    value = "Binary", position="21", type = "INTEGER",
    precision="10,0", convertTo="BINARY" }
    field = { description = "Float", fromMessage = "true",
    value = "Float", position="29", type = "DOUBLE", precision="7,2",
    convertTo="COMP-1" }
    field = { description = "Double", fromMessage = "true",
    value = "Double", position="33", type = "DOUBLE", precision="10,4",
    convertTo="COMP-2" }
    field = { description = "Hex", fromMessage = "true", value = "Hex",
    position="41", length="10", type = "OPAQUE" }
    field = { description = "Date", fromMessage = "true",
    value = "Date", position="51", length="10", type = "OPAQUE" }
    field = { description = "Time", fromMessage = "true",
    value = "Time", position="61", length="8", type = "OPAQUE" }
    field = { description = "STime", fromMessage = "true",
    value = "STime", position="69", length="26", type = "OPAQUE" }
    }

```

*Example 13 Record Mode Subscriber with ECM*

This example shows how to configure FileSubscriber to support ECM. In Record Mode ECM you have to specify endPublishSubject to signal the end of file publishing to FileSubscriber. The generateFileSubjectName parameter in FileSubscriber have to be same as endPublishSubject.

Configuration file name – SUB005.

The FileType section of the configuration file:

```
[FileType]
FILE_OPTIONS = { filePrefix="FT5",
                 delimiter = "|",
                 skipPadding = "true",
                 isBinary = "false",
                 noOfRetries = "10", retryInterval = "5",
                 subscribeSubjectName = "SUB.PUB005.FT1",
                 generateFileSubjectName = "ENDPUB.PUB005.FT1",
                 generateFileFieldName = "filename",
                 confirmationSubject = "ECM.PUB005.FT1",
                 autoGenerateFile = "false",
                 transferType = "RecordModeECM"
               }

FILE_LINE = {
  field = { description = "Part Number", position = "1",
    fromMessage = "true", value = "PartNo", type = "STRING" },
  field = { description = "Part Description", position = "2",
    fromMessage = "true", value = "Desc", type = "STRING" },
  field = { description = "Manufacturer", position = "3",
    fromMessage = "true", value = "Manufacturer", type = "STRING" },
  field = { description = "Model", position = "4",
    fromMessage = "true", value = "Model", type = "STRING" },
  field = { description = "Quantity", position = "5",
    fromMessage = "true", value = "Quantity", type = "INTEGER" },
  field = { description = "Unit Price", position = "6",
    fromMessage = "true", value = "Price", type = "FLOAT" },
  field = { description = "Warranty Period in years", position = "7",
    fromMessage = "true", value = "Warranty", type = "STRING" }
}
```

*Example 14 Block Mode Subscriber - Simple File Transfer*

This configuration is for subscribing a file using Block Mode.

Configuration file name – SUB006.

The FileType section of the configuration file:

```
[FileType]
FILE_OPTIONS = { filePrefix="FT6",
                  dataSetType = "SEQ",
                  isBinary = "false",
                  lineLength = "100",
                  useFixedRecordFile = "true",
                  appendToExistingFile = "false",
                  truncateRecords = "error",
                  exitOnFileSaveError = "false",
                  noOfRetries = "10", retryInterval = "5",
                  isCertified = "false",
                  subscribeSubjectName = "SUB.PUB006.FT2",
                  generateFileSubjectName = "ENDPUB.PUB006.FT2",
                  generateFileFieldName = "filename",
                  autoGenerateFile = "false",
                  transferType = "BlockModeSFT"
                }
```

*Example 15 Block Mode Subscriber with ECM*

This configuration is for subscribing a file in Block Mode using ECM.

Configuration file name – SUB007.

The FileType section of the configuration file:

```
[FileType]
FILE_OPTIONS = { filePrefix="FT7",
                  dataSetType = "SEQ",
                  isBinary = "false",
                  lineLength = "100",
                  useFixedRecordFile = "true",
                  appendToExistingFile = "false",
                  truncateRecords = "error",
                  exitOnFileSaveError = "false",
                  noOfRetries = "10", retryInterval = "5",
                  isCertified = "false",
                  subscribeSubjectName = "SUB.PUB007.FT2",
                  useExplicitConfirmation = "true",
                  ECMSubscriberName = "ECM.PUB007.FT2",
                  transferType = "BlockModeECM"
                }
```

## Guaranteed Delivery for EMS Messages

---

This section describes the setup required for guaranteed delivery for EMS messages using durable subscribers.

Guaranteed delivery and receipt of EMS messages is enabled by sending persistent messages over topics to durable subscribers. For this purpose, only the transferTypes of BlockModeSFT and RecordMode with EMS are used.

Durable subscribers to a topic cause published messages to be saved while the durable subscriber is offline. Also, subscribers to a topic that are connected to a fault-tolerant server receive messages from the secondary server after a failover

Follow these steps to achieve guaranteed delivery:

1. Set the `deliveryMode="persistent"` for the Publisher.

Persistent messages published to a topic are written to disk only if that topic has at least one durable subscriber or one subscriber with a fault-tolerant connection to the EMS server.

2. Set the `publishDestinationType="TOPIC"` for the Publisher

Similarly, for the subscriber, set the `subscribeDestinationName="TOPIC"`.

3. Set the `SubscriberName="sub1"`

Here, `sub1` is an identifier of your choice.

A new `SubscriberName` ID is required so that the EMS server can identify which consumer clients requires which messages resent after a failure. This `Subscribername` ID is created dynamically when the Subscriber is started. These durable subscriber `Subscriber` IDs can also be created with the aid of the EMS Administration Tool or through configuration files on the server. This first implementation of durable subscriber support in the adapter does not have any mechanism to remove a durable subscriber through the Subscriber; the EMS Administration tool is used.

4. Optional specify which `EMS_ACK_TYPE` acknowledgement for the Subscriber to use.
5. If the `endPublishDestinationName` parameter is specified in the Publisher, it must equal the same value as `publishDestinationName`. Similarly, the value of the `subscribeDestinationName` parameter and `publishDestinationName` must equal the value of the `generateFileSubjectName` parameter. in the Subscriber.





## Chapter 4      **Using the Adapter**

This chapter contains a variety of topics for using TIBCO Adapter for Files (IBM i).

### Topics

---

- [Starting and Stopping the Adapter, page 116](#)
- [Block Transfer Mode, page 120](#)
- [File Transfer Using ECM, page 121](#)
- [Sending and Receiving Numeric Data Types, page 124](#)
- [Pre-Processing and Post-Processing Files, page 125](#)
- [Using Heartbeat Messages, page 127](#)
- [Working with Batch Messages, page 128](#)
- [Sending Trigger Messages, page 129](#)
- [FilePublisher Usage Guidelines, page 130](#)
- [FileSubscriber Usage Guidelines, page 133](#)

## Starting and Stopping the Adapter

---

This section describes how to start and stop FilePublisher and FileSubscriber.

### Starting FilePublisher

Before starting FilePublisher, ensure the TIBCO Rendezvous daemon is running. For example:

```
SBMJOB CMD(CALL PGM(TIBRV/RVD)) ALWMLTTHD(*YES)
```

If using a remote daemon, ensure that it is started with the `-permanent` option. Normally if the TIBCO Rendezvous daemon is not running, execution of FilePublisher starts the daemon automatically.

The adapter, as well as the TIBCO Rendezvous daemon, require a multi-user threaded environment to run. The Command Entry interactive display does not provide this environment.

### Using the SBMJOB Command

The FilePublisher program, SXF3RPUB, can be submitted to batch. Use the SBMJOB command to do this.

The following is an example of the SBMJOB command. Note that there are two parameters, JOBMSGQFL and ALWMLTTHD, that require a value other than the default.

```
ADDLIB TIBFALIB

SBMJOB CMD(call pgm(SXF3RPUB)
PARM('-config' 'TIBFASMPRV/INIFPUB(PUB005)'))
JOBMSGQFL(*PRTWRAP)
ALWMLTTHD(*YES)
```

Parameters in this SBMJOB command example were set as follows:

- The FilePublisher program, SXF3RPUB, is assumed to be in the library TIBFALIB.
- FilePublisher was submitted with the `-config` parameter. This causes FilePublisher to use configuration file member PUB005 in file INIFPUB. If you omit this parameter, FilePublisher uses the sample configuration file, TIBFALIB/INIFPUB(INIFPUB).
- The JOBMSGQFL parameter was set to `*PRTWRAP`, in case a large number of messages issued to the job message queue.

- The ALWMLTTHD parameter must be set to \*YES. So that the job can run with multiple user threads.

### Using a CL Program

One can fashion a CL program after the CL programs in TIBFASMPRV/QCLSRC to easily submit the FilePublisher to batch.

### Using the QShell Environment

You can also start FilePublisher interactively in the QShell environment, as follows:

```
ADDLIBLE TIBFALIB
QSH
cd /usr/tibco/tibfa/TIBFALIB
sxf3rpub -config 'tibfalib/inifpub(pub003)'
```

or

```
sxf3rpub -service 7500 -daemon hawk400.na.tibco.com:7500
-config 'tibfalib/inifpub(pub003)'
```

## Starting FileSubscriber

Before starting FileSubscriber, ensure the TIBCO Rendezvous daemon is running. For example:

```
SBMJOB CMD(CALL PGM(TIBRV/RVD)) ALWMLTTHD(*YES)
```

If using a remote daemon, ensure that it is started with the `-permanent` option. Normally if the TIBCO Rendezvous daemon is not running, execution of FileSubscriber starts the daemon automatically.

### Using the SBJJOB Command

The FileSubscriber program, SXF3RSUB, can be submitted to batch. Use the SBJJOB command to do this.

The following is an example of the SBJJOB command. Note that there are two parameters, JOBMSGQFL and ALWMLTTHD, that require a value other than the default.

```
ADDLIBLE TIBFALIB

SBMJOB CMD(call pgm(SXF3RSUB)
PARM('-config' 'TIBFASMPRV/INIFSUB(SUB005)'))
JOBMSGQFL(*PRTWRAP)
ALWMLTTHD(*YES)
```

Parameters in this SBMJOB command example were set as follows:

- The FileSubscriber program, SXF3RSUB, is assumed to be in the library TIBFALIB.
- FileSubscriber was submitted with the `-config` parameter. This causes FileSubscriber to use the configuration file member SUB005 in file INIFSUB. If you omit the `parm` parameter, FileSubscriber uses the sample configuration file, TIBFALIB/INIFSUB(INIFSUB).
- The JOBMMSGQFL parameter was set to `*PRTWRAP`, in case a large number of messages issued to the job message queue.
- The ALWMLTTHD parameter must be set to `*YES`. So that the job can run with multiple user threads.

### Using a CL Program

One can fashion a CL program after the CL programs in TIBFASMPRV/QCLSRC to easily submit the FileSubscriber to batch.

### Using the QShell Environment

You can also start FileSubscriber interactively in the QShell environment, as follows:

```
ADDLIB TIBFALIB
QSH
cd /usr/tibco/tibfa/TIBFALIB
sxf3rsub -config 'tibfalib/inifsub(sub003)'
```

or

```
sxf3rsub -service 7500 -daemon hawk400.na.tibco.com:7500
-config 'tibfalib/inifsub(sub003)'
```

## Stopping FilePublisher

An active FilePublisher listens to the following terminate subject:

```
_FILEADAPTER.<adapterName>.TERMINATE
```

where `<adapterName>` is the name of the adapter instance as configured in the FilePublisher configuration file. You can stop FilePublisher by sending a terminate message to this terminate subject.

### Using the SBMJOB Command

You can use the following SBMJOB command to stop FilePublisher:

```
SBMJOB CMD(CALL PGM(TIBFALIB/SXFTRIGR) PARM('-service' 7500
'-daemon' 'hawk400.na.tibco.com:7500'
'_FILEADAPTER.TEST_FILE_PUB.TERMINATE' 'any message'))
ALWMLTTHD(*YES)
```

### Using the QShell Environment

You can also stop FilePublisher interactively in the QShell environment, as follows:

```
ADDLIB TIBFALIB
QSH
cd /usr/tibco/tibfa/TIBFALIB
sxftrigr -service 7500 -daemon hawk400.na.tibco.com:7500
_FILEADAPTER.TEST_FILE_PUB.TERMINATE 'any message'
```

## Stopping FileSubscriber

An active FileSubscriber listens to the following terminate subject:

```
_FILEADAPTER.<adapterName>.TERMINATE
```

where *<adapterName>* is the name of the adapter instance as configured in the FileSubscriber configuration file. You can stop FileSubscriber by sending a terminate message to this subject.

### Using the SBJOB Command

You can use the following SBJOB command to stop FileSubscriber:

```
SBMJOB CMD(CALL PGM(TIBFALIB/SXFTRIGR) PARM('-service' 7500
'-daemon' 'hawk400.na.tibco.com:7500'
'_FILEADAPTER.TEST_FILE_SUB.TERMINATE' 'any message'))
ALWMLTTHD(*YES)
```

### Using the QShell Environment

You can also stop FileSubscriber interactively in the QShell environment, as follows:

```
ADDLIB TIBFALIB
QSH
cd /usr/tibco/tibfa/TIBFALIB
sxftrigr -service 7500 -daemon hawk400.na.tibco.com:7500
_FILEADAPTER.TEST_FILE_SUB.TERMINATE 'any message'
```

## Block Transfer Mode

---

In some situations, you might want to publish a file without regard to its file structure. To do this, specify that the file is to be published, or subscribed to, in block mode. When a file is processed in block mode, the data from the file is read without regard to the field or record structure, and the data is published as a block.

### Configuration

To enable block mode, specify the following parameters in the `FileType` section, `FILE_OPTIONS` element:

- `blockTransferMode` — When enabled, the file is published in blocks of data without further processing of the data (you must also specify the `endPublishSubject` parameter for `FilePublisher` and the `generateFileSubjectName` parameter for `FileSubscriber`).
- `blockTransferSize` — Specifies the block size, in bytes, that the adapter uses to publish the file. The default value is 65536. This parameter is not used for `FileSubscriber`.

A file that is published in block transfer mode must be subscribed to in block transfer mode. Data received from a file that is published in block transfer mode does not have record or field characteristics. When the data is received by `FileSubscriber`, the blocks are written to a progress file. `FileSubscriber` generates a final output file only when it receives a message with the subject specified in `genFileSubjectName`. The `FILE_LINE` element is not required, and is ignored.

`FileSubscriber` receives a message with a subject that `FileSubscriber` has defined with the `generateFileSubjectName` parameter. When block transfer mode is enabled, the following tags are ignored for that file type in `FileSubscriber`:

- `appendDateTime`
- `autoGenerateFile`
- `generateFileOnNumberOfMessages`

When a file is published in block transfer mode, `FilePublisher` reads the file in blocks, as specified with the `blockTransferSize`, without additional processing of the data. The `MESSAGE_FIELD` element is not required, and is ignored.

## File Transfer Using ECM

---

ECM is well suited for transferring files without having to define schemas. Both FilePublisher and FileSubscriber must be set up to use ECM for successful transfer.

In ECM, FilePublisher performs the following tasks:

- On initial startup, FilePublisher sends out an administration message to the ECM subscribers specified in its configuration file. This process is initiated only when FilePublisher starts for the first time.
- On restart, the active configuration information is retrieved from the progress file. Resynchronized handshakes are performed as needed to re-establish communication between FilePublisher and its subscribers.
- When an inactive subscriber requests activation, FilePublisher marks the subscriber to be activated on a new file boundary.

In ECM, FileSubscriber performs the following tasks:

- On initial startup, FileSubscriber waits for an administration message from FilePublisher. Once the handshake has been established, FileSubscriber creates a progress file.



Do not edit or delete the progress file.

- When FileSubscriber is restarted, it uses the progress file it previously created and proceeds.
  - If a file transfer was previously active, FileSubscriber continues from where it left off.
  - If no file transfer is active, FileSubscriber waits for an administration message from FilePublisher and starts receiving file transfers on a new file boundary.

### Error Handling

FilePublisher keeps track of the last block published in the progress file for restart purposes. When the adapter encounters an error, it publishes a message using the error subject established in the configuration file (ERROR\_SUBJECT).

FileSubscriber keeps track of the last block of data received and written in the progress file for restart purposes. When a subscriber detects an error (such as a locked file) in the middle of a file transfer, the subscriber converts the working file to an error file. When the adapter encounters an error, it publishes a message using the error subject established in the configuration file (ERROR\_SUBJECT).

### TIBCO Rendezvous Subjects Used by ECM

The ECM administration messages are exchanged using the following subject names:

- `_TIBCO_AE_ADAPTER_FAFT_PUBTOSUB` . <endpoint subject name>
- `_TIBCO_AE_ADAPTER_FAFT_SUBTOPUB` . <endpoint subject name>

Administrators can listen to these subjects at runtime.

## ECM Configuration

The following are used for FilePublisher configuration:

- `useExplicitConfirmation`: Flag that specifies whether Explicit Confirmation Mode is on or off during block transfer.

All the options below are only valid if this flag is true. [default: "false"]

- `ECMSubscriberName`: This field specifies the corresponding ECM subscribers name. When there are more than one participating subscribers, repeat this option and enter a unique subscriber name.
- `BlockTransferMode`: set to true to ensure block mode is used.
- `retransmissionDelayTicks`: Specifies the number of seconds that FileSubscriber has to wait between retries to re-connect to the Publisher after a startup. It is used to coordinate the startup handshake used between the Publisher and Subscriber, when ECM is being used. The default value is 10 seconds.
- `noWaitAfterConfirmations`: If true, the next transaction is processed immediately after FilePublisher receives all the confirmations. If false, it uses the normal `transactionDelay` parameter to process transactions. The default value is "true".

The following are the FileSubscriber options.

- `useExplicitConfirmation`: Flag that specifies whether Explicit Confirmation Mode is on or off during block transfer. All the options below are only valid if this flag is true. [default: "false"]



- `ECMSubscriberName`: the ECM subscriber name. This entry must match the corresponding participating ECM Publisher.



TIBCO Adapter for Files (IBM i) and TIBCO Adapter for z/OS can communicate using the block mode and record mode using ECM only with Rendezvous.

TIBCO Adapter for Files (IBM i) and TIBCO Adapter for Files can communicate using ECM with Rendezvous only in the block mode.

## Sending and Receiving Numeric Data Types

---

When sending records from FilePublisher with Packed Decimal fields (COMP-3), the message item description must set the type to either COMP-3 or PACKED. The packed data is automatically converted to 8-byte floating point DOUBLE on the wire, unless the `convertToString` option is used. A `convertToString="true"` causes the packed data to be converted in to a DISPLAY format, sent as STRING on the wire. This avoids any loss of precision and rounding that can occur if FLOAT is used to represent decimal numbers. Sample definitions are shown below:

Sent as DOUBLE:

```
messageItem = { position="4", precision="7,2",
                label="ITEM-PRICE", type="COMP-3" },
```

Sent as STRING:

```
messageItem = { position="4", precision="7,2",
                label="ITEM-PRICE", type="COMP-3",
                convertToString="true" },
```

When sending records with COMP-1 FLOAT, you might set the type field to COMP-1 or FLOAT. When sending COMP-2 DOUBLE, you might set the type field to COMP-2 or DOUBLE. When sending BINARY fields, you can either set the type field as SHORT (2 byte), INTEGER (4 byte) or LONGLONG (8 byte), or you might set the type field to BINARY or COMP, and explicitly set the length using the precision field as described above.

When receiving records at the Subscriber, Packed Decimal field (COMP-3) message descriptions have to match what was sent from the Publisher. If the Publisher sent the data on the wire as DOUBLE, the type field have to be set to DOUBLE, and the `convertTo` option must be used, and set to COMP-3 or PACKED. If the Publisher sent the data on the wire as STRING, the type field have to be set to STRING, and the `convertTo` option must be used, and set to COMP-3 or PACKED. Sample definitions are shown below:

Sent as DOUBLE:

```
field = { fieldStart="4", precision="7,2",
          value="ITEM-PRICE", type="DOUBLE",
          convertTo=COMP-3" },
```

Sent as STRING:

```
field = { fieldStart="4", precision="7,2",
          value="ITEM-PRICE", type="STRING",
          convertTo=COMP-3" },
```

## Pre-Processing and Post-Processing Files

---

You can configure TIBCO Adapter for Files (IBM i) to perform the following pre- and post-processing tasks:

- Process a file before or after it is published, or before or after it is written.
- Execute an IBM i command or program before or after a file is published, or before or after an output file is generated.

To perform these actions, you use the `executeBeforeProcess` and `executeAfterProcess` configuration parameters; these parameters are found in the `FileType` section, `FILE_OPTIONS` element in the configuration file for either `FilePublisher` or `FileSubscriber`.

If you want to have a program called, you must include the IBM i command `call` as part of the parameter. If you specify a valid command or program name that is not in the `FilePublisher`'s or `FileSubscriber`'s library list, then you must include the library name in the parameter.



When a program is called by either `FilePublisher` or `FileSubscriber`, `FilePublisher` or `FileSubscriber` stops processing the current file until the called program has finished executing.

### Pre-Processing Files

When you configure the adapter to pre-process files, the following occurs:

- **FilePublisher** When `FilePublisher` has a file that is to be published, `FilePublisher` executes the command or program. You must provide for a parameter that `FilePublisher` returns to the command or program. When `FilePublisher` executes the command, it supplies the name of the file that is to be published as a parameter.
- **FileSubscriber** When `FileSubscriber` has an output file to be generated, `FileSubscriber` executes the command or program. You must provide for a parameter that `FileSubscriber` passes to the command or call program. When `FileSubscriber` invokes the command, it passes the name of the file as a parameter.

### Sample Configuration

Suppose that you want to call a CL program named `PROCFILE`, which is in a library named `PGMLIB`, before you publish a file. In the `FileType` section, `FILE_OPTIONS` element of the `FilePublisher` configuration file, you would include the `executeBeforeProcess` parameter and include the file name, as follows:

```
executeBeforeProcess="CALL PGMLIB/PROCFILE"
```

For the CL program, you must include a CL variable that receives the name of the file that is to be published. For example:

```
PGM PARM(&FILENAME)
```

and

```
DCL VAR(&FILENAME) TYPE(*CHAR) LEN(50)
```

When the CL program is called, the CL variable &FILENAME contains the name of the library and the file that are to be published. For example, if the file to be published is FT111, and the library name of the output library is TIBRVOUT:

```
&FILENAME = 'TIBRVOUT/FT111'
```

## Post-Processing Files

When you configure the adapter to post-process files, the following occurs:

- **FilePublisher** When FilePublisher has finished publishing a file, FilePublisher executes the command or program. You must provide for two parameters that FilePublisher returns to the command or program. When FilePublisher executes the command or program, it supplies the name of the file that was published and the file status as parameters. The file name parameter passed includes the library name. The file status parameter is 0 if the file was published successfully, or 1 if there was an error publishing the file.
- **FileSubscriber** When FileSubscriber has finished generating an output file, FileSubscriber executes the command or program. You must provide for two parameters that FileSubscriber returns to the command or program. When FileSubscriber executes the command, it supplies the name of the file that was written and the file status as a parameter. The file name parameter passed includes the library name. The file status parameter is "0" if the file was processed successfully, or "1" if there was an error processing the file.

## Using Heartbeat Messages

Heartbeat messages provide an indication that FilePublisher or FileSubscriber is active. These messages can be monitored by TIBCO Hawk™. TIBCO Hawk can send notifications or alerts when FilePublisher or FileSubscriber goes down.

### Configuration

To enable heartbeat messages, specify the following elements in the Options section of the FilePublisher or FileSubscriber configuration file:

- `PUBLISH_HEARTBEAT` – To enable heartbeat messages, set this element to `true`. The default value is `false`.
- `HEARTBEAT_TIME` – Specifies the interval between heartbeat messages. The default value is 60000 milliseconds, or 60 seconds.

### Message Format

The subject on which the adapter publishes heartbeat messages is in the following format:

```
_FILEADAPTER.<adaptername>.HEARTBEAT
```

For FilePublisher, the format of the heartbeat message is FilePublisher start time and current time. For each configured file type, the format is:

- File prefix
- File extension (N/A for IBM i)
- File status — Idle or Busy (being published)
- If busy, number of messages (or blocks) published

For FileSubscriber, the format of the heartbeat message is FileSubscriber start time and current time. Then, for each file type, the format is file prefix and file extension (N/A for IBM i).



The adapter does not provide TIBCO Hawk microagents. It only publishes Heartbeat messages at specified intervals as specified above. You can monitor key events that are processed by the adapter by setting the `EEM_Subject` element and installing the TIBCO EEM monitoring software. For details, see [BusinessEvents™ Messages on page 15](#).

## Working with Batch Messages

---

If you want to subscribe to a batch of messages and generate the out file after receiving the entire batch, complete the following steps:

1. Publish the batch of records with a `publishSubjectName`, for example, A.B.
2. Using the subscriber, configure a subscription to the messages with the `subscribeSubjectName` set to "A.B" (in the FileSubscriber configuration file).
3. Set the `generateFileSubjectName`="A.B" in the FileSubscriber configuration file.
4. Set the `generateFileFieldName`="STOP" (can be any value, STOP is descriptive)
5. After publishing the entire batch, publish a message with `subjectName`="A.B", which contains the STOP field. For example.

```
A.B
{
  RVMSG_STRING 6  STOP  "12345"
}
```

or

```
tibrvlisten: Listening to subject A.B
[2002-01-22 20:30:14]: subject=A.B, message={STOP="TIBADAPTER"}
```

Note that this solution might not work in a multi-threaded environment. That is, if several FilePublishers are sending messages on the same subject, the above scenario fails

## Sending Trigger Messages

---

Trigger messages can be sent with the file syntax shown below (QShell environment):

1. Start QShell:

```
QSH
```

2. Change to the bin directory:

```
CD /usr/tibco/tibfa/TIBFALIB
```

3. Send the following messages:

```
sxftrigr -service 7500 -daemon hk-as400.na.tibco.com:7500  
TRIGGER.FB80.ONE 'TIBFALIB/FILE1'
```

```
sxftrigr -service 7500 -daemon hk-as400.na.tibco.com:7500  
TRIGGER.FB80.ONE 'TIBFALIB/FILE1(MEMBERn)'
```

```
sxftrigr -service 7500 -daemon hk-as400.na.tibco.com:7500  
TRIGGER.FB80.ONE 'TIBFALIB/FILE1(*FIRST)'
```

```
sxftrigr -service 7500 -daemon hk-as400.na.tibco.com:7500  
TRIGGER.FB80.ONE 'TIBFALIB/FILE1(*LAST)'
```

```
sxftrigr -service 7500 -daemon hk-as400.na.tibco.com:7500  
TRIGGER.FB80.ONE 'TIBFALIB/FILE1(*ALL)'
```

## FilePublisher Usage Guidelines

---

This section provides some guidelines for using FilePublisher and explains special configuration options. It includes the following information:

- [Sending Numeric Data Types on page 130](#)
- [Sending Data Untranslated \(OPAQUE\) on page 131](#)
- [Constructing a Subject Name from Data on page 131](#)
- [Publishing Double Values on page 132](#)
- [Pre-registering Subscribers on page 132](#)

### Sending Numeric Data Types

This section explains how FilePublisher can be configured to handle the numeric data types supported by the adapter.

You can configure FilePublisher to publish non-text files by using the `isBinary=true` parameter and the `messageItem` parameter tags in the `FileType` section. When transmitted to a Subscriber, numeric data types are converted to `FLOAT`, `STRING` or `INTEGER` TIBCO data types.

When you configure one of these data types, you use the `type` tag to identify the type of numeric field. To specify a binary decimal field, you can specify `type="binary"` or `type="COMP"` or `type="COMP-4"`. To specify packed decimal, you can specify `type="PACKED"` or `type="COMP-3"`.

Floating-point numeric fields are specified as either `COMP-1` or `COMP-2`. `COMP-1` indicates that the input field is a 4 byte long floating point number. `COMP-2` indicates that the input field is an 8 byte long floating point number.



Since there can be a loss of precision when converting to `FLOAT` from non-float values, it might be preferable to specify `STRING` as the published data type.

See [Sample Configuration Elements on page 70](#) for an example of a FilePublisher configuration file that handles the supported numeric data types.



## Sending Data Untranslated (OPAQUE)

You can configure the adapter to publish a field without translating the data within the field. Usually, when FilePublisher reads a record from a file, it translates the data within each field to TIBCO Rendezvous “wire format,” which is then published. When FileSubscriber receives the published message, it converts the “wire format” message to the appropriate data type for the platform that it is on.

If it is necessary to publish binary data other than the supported COBOL numeric data types, from one platform to another platform that supports the same data types (such as from one EBCDIC machine to another EBCDIC machine), you can specify OPAQUE for the data type. When you do this, the data in the field is published “as is,” with no conversion. When FileSubscriber receives the OPAQUE TIBCO Rendezvous message, it writes the data in that field to the field “as is.”

### To support zoned, binary, packed, floating-point, and opaque data types:

1. Set the `isBinary` parameter in the `FILE_OPTIONS` element to `true`.
2. Set the `type` tag in the `messageItem` parameter to the desired value.

## Constructing a Subject Name from Data

Data from a file is normally published with a predefined subject name. However, if the subject name is not constant for all the lines of data in a file, and if the subject depends on a certain data field from the input file, specify a valid node within angle brackets (< >). A valid node is a label of a defined `messageItem` parameter prefixed by its container name separated by a period.

For example:

```
publishSubjectName = "CHECK.JAN.<ChKey.RoutingCode>.CLEARED"
```

uses data of the label `RoutingCode` in the container name `ChKey` to construct the `publishSubjectName`. That is, if the value of `RoutingCode` in the message to publish is 154363, the `publishSubjectName` for publishing, that message is `CHECK.JAN.154363.CLEARED`.

The configuration file coding necessary in this example would be as follows:

```
[FileType]
FILE_OPTIONS={...
PublishSubjectName="CHECK.JAN.<ChKey.RoutingCode>.CLEARED",
}
MESSAGE_FIELDS={
messageContainer={name="ChKey",
messageItem={position="0",length="6",
Label="RoutingCode", type="STRING"},
messageItem={ ... },
messageItem={ ... },
```



Constructing a Subject Name from Data is ONLY supported by Rendezvous. It is not supported for EMS.

## Publishing Double Values

FilePublisher supports publishing of floating point numbers in three ways:

- Float (4 byte)
- Double (8 byte)
- STRING – can be used to avoid losing precision

The adapter defaults to 6-digit precision for these values.

## Pre-registering Subscribers

You can configure FilePublisher to specify a list of anticipated subscribers for the subjects it is publishing. By pre-registering the anticipated subscribers, the delivery of certified messages is guaranteed even if FileSubscribers start before or after the configured FilePublisher or even if a FileSubscriber is stopped and restarted later. To use this feature, both FilePublisher and FileSubscriber have to establish RVCN sessions with the `requireOldMessages` parameter set to `true`. To implement the pre-registering of subscribers, include this section in the FilePublisher configuration file:

```
[Pre-Register]
SUBSCRIBER = { listenerName = "<name of the subscribing session>",
               listeningSubject = "<name of the subject subscribed for>" },
SUBSCRIBER = { listenerName = "<name of the subscribing session>",
               listeningSubject = "<name of the subject subscribed for>" }
.....
```

Currently, the maximum number of subscribers that can be pre-registered is 126.

## FileSubscriber Usage Guidelines

---

This section discusses the following FileSubscriber usage guidelines:

- [Receiving Numeric Data Types on page 133](#)
- [Adding Header and Trailer Records on page 133](#)

### Receiving Numeric Data Types

This section explains how FileSubscriber can be configured to handle the numeric data types supported by the adapter.

FileSubscriber can be configured to create non-text files by using the `isBinary=true` parameter in the `FileType` section and two field parameter tags in the `FILE_LINE` element. As a result, FileSubscriber can process non-text data types. The numeric data types are converted from incoming TIBCO data types of either `INTEGER`, `UNSIGNED INTEGER`, `SHORT`, `UNSIGNED SHORT`, `FLOAT`, or `STRING` into the desired target data type that is specified in the `convertTo` parameter. Because there can be a loss of precision when converting from `FLOAT` values, it might be preferable to specify `STRING` as the published data type. The numeric data type to be used for the output record for each field is specified by the tag `convertTo` in a field parameter of the `FILE_LINE` element.

To support zoned, binary, packed, and floating-point data types:

1. Set the `isBinary` parameter in the `FILE_OPTIONS` element to `true`.
2. Set the `convertTo` and `precision` tabs in the `FILE_LINE` element to desired values.



Since the fields of data type `BINARY(COMP, COMP-4)` are being published as `INTEGER`, mention the `type="INTEGER"` and `convertTo="BINARY"`.

### Adding Header and Trailer Records

You can add a header record or a trailer record, or both, to an output file. The added records can contain any text string and variable data.

To add a header record, use the `fileHeader` parameter in the `FILE_OPTIONS` element of the FileSubscriber configuration file. To add a trailer record, use the `fileTrailer` parameter in the `FILE_OPTIONS` element of the FileSubscriber configuration file. For each of these parameters, a text string and up to three variables — the number of records, a blank field, and the date and time — can be specified in the following format:

```
fileHeader="<text>| [%<fieldWidth>,NUMBER_OF_RECORDS%]|
%<fieldWidth>,BLANK%]| [<text>| %<fieldWidth>,DATE_TIME%]"

fileTrailer="<text>| [%<fieldWidth>,NUMBER_OF_RECORDS%]|
%<fieldWidth>,BLANK%]| [<text>| %<fieldWidth>,DATE_TIME%]"
```

where

`<text>` is any alphanumeric descriptive string. If included, it can appear anywhere within the quotation marks but must not be included within the square brackets ([ ]).

`%<fieldWidth>` is the number of characters in the related field; `%` is a C type substitution

`NUMBER_OF_RECORDS` is replaced with the number of records in the output file. If there are no errors, this number have to correspond to the number of messages received to generate the file.



Because the adapter receives records into a work file then generates the final output file, the `NUMBER_OF_RECORDS` variable can be used when specifying a header record.

`DATE_TIME` is the date and time in the format `YYYYMMDDHHMMSS`.

`BLANK` is a blank field that can be used to provide a space between the other variables.

### FileHeader Example

Suppose you want to add a header record to an output file that shows the number of records received and the date and time that the output file was created, with two spaces between the fields. You would include the following parameters in the `FILE_OPTIONS` element of the FileSubscriber configuration file as follow:

```
FILE_OPTIONS = { prefix="test",
fileHeader="[%4,NUMBER_OF_RECORDS%] [%2,BLANK%] [%14,DATE_TIME%]" ,
subscribeSubjectName=...
```





## Appendix A **Trace Log Examples**

This appendix shows trace logs that record a successful session and a failed session for FilePublisher and FileSubscriber. Information in these trace logs is produced according to trace levels specified by the configuration files for FilePublisher and FileSubscriber.

### Topics

---

- [Successful FilePublisher Session, page 138](#)
- [Failed FilePublisher Session, page 142](#)
- [Successful FileSubscriber Session, page 145](#)
- [Failed FileSubscriber Session, page 151](#)

## Successful FilePublisher Session

---

```

2016 Jan 18 23:40:01 PBL INFO [CFG] D9 AE400-INIT_TRACE SXF0050I Copyright (c)
1997-2016. TIBCO Software Inc.
2016 Jan 18 23:40:01 PBL INFO [CFG] D9 AE400-INIT_TRACE SXF0050I All Rights
Reserved. Confidential & Proprietary.
2016 Jan 18 23:40:01 PBL INFO [CFG] D9 AE400-INIT_TRACE SXF0051I
*****
2016 Jan 18 23:40:01 PBL INFO [CFG] D9 AE400-INIT_TRACE SXF0052I TIBCO Adapter for
Files (IBM i) start...
2016 Jan 18 23:40:01 PBL INFO [CFG] D9 AE400-INIT_TRACE SXF0053I File Adapter
Publisher 5.0.0 GAV06 [31151] Jan 7 2016
2016 Jan 18 23:40:01 PBL INFO [CFG] D9 AE400-INIT_TRACE SXF0055I Using
configuration file: <TIBFA500R/INIFPUB(PUB001)>
2016 Jan 18 23:40:01 PBL INFO [CFG] D9 AE400-INIT_TRACE SXF0355I Using trace file:
<TIBFA500R/PUBLOG(PUB001)> Size=2000000
2016 Jan 18 23:40:01 PBL INFO [CFG] D9 AE400-INIT_TRACE SXF0341I Trace Level: 2
TZ=N/A
2016 Jan 18 23:40:01 PBL INFO [CFG] D9 AE400-INIT_TRACE SXF0368I Trace Option
Switches: NNNNNNNN
2016 Jan 18 23:40:01 PBL INFO [CFG] D9 AE400-INIT_TRACE SXF0056I
*****
2016 Jan 18 23:40:01 PBL INFO [CFG] D9 AE400-PRS_CFG_FL SXF0179I [===] OPTIONS
section:
2016 Jan 18 23:40:01 PBL INFO [CFG] D9 AE400-LD_RV_PARM SXF0061I <--- RV_SESSION
Summary Begin
2016Jan1823:40:01PBLINFO[CFG]D9AE400-LD_RV_PARMSXF0062IName:FILE_PUB_001
2016Jan1823:40:01PBLINFO[CFG]D9AE400-LD_RV_PARMSXF0063IService:9721
2016Jan1823:40:01PBLINFO[CFG]D9AE400-LD_RV_PARMSXF0064INetwork:
2016Jan1823:40:01PBLINFO[CFG]D9AE400-LD_RV_PARMSXF0065IDaemon:tcp:9720
2016 Jan 18 23:40:01 PBL INFO [CFG] D9 AE400-LD_RV_PARM SXF0066I ----> RV_SESSION
Summary End
2016 Jan 18 23:40:01 PBL INFO [CFG] D9 AE400-PRS_OPT_AT SXF0163I Adapter Name:
FILE_PUB_001
2016 Jan 18 23:40:01 PBL INFO [CFG] D9 AE400-PRS_OPT_AT SXF0160I Input Directory:
TIBFA500R
2016 Jan 18 23:40:01 PBL INFO [CFG] D9 AE400-PRS_OPT_AT SXF0161I Process Directory:
FAPUBPROC
2016 Jan 18 23:40:01 PBL INFO [CFG] D9 AE400-PRS_OPT_AT SXF0162I Output Directory:
FAPUBOUT
2016 Jan 18 23:40:01 PBL INFO [CFG] D9 AE400-PRS_OPT_AT SXF0175I Continue on
ConfigurationError: true
2016 Jan 18 23:40:01 PBL INFO [CFG] D9 AE400-PRS_OPT_AT SXF0168I Max Concurrent
Jobs: 5
2016 Jan 18 23:40:01 PBL INFO [CFG] D9 AE400-PRS_OPT_AT SXF0172I Print File
Options: true
2016 Jan 18 23:40:01 PBL INFO [CFG] D9 AE400-PRS_OPT_AT SXF0177I Log to QSYSOPR
*MSGQ: true
2016 Jan 18 23:40:01 PBL INFO [CFG] D9 AE400-PRS_OPT_AT SXF0352I EPM
Subject/DestinationName: <EPM_SUBJECT>
2016 Jan 18 23:40:01 PBL INFO [CFG] D9 AE400-PRS_OPT_AT SXF0170I Error
Subject/Destination: ERROR.SUBJECT
2016 Jan 18 23:40:01 PBL INFO [CFG] D9 AE400-PRS_OPT_AT SXF0173I Error Exit
Condition Code: 1

```



```

2016 Jan 18 23:40:01 PBL INFO [CFG] D9 AE400-INIT_RVTRAN SXF0057I Initializing,
using RV client library version 8.4.4
2016 Jan 18 23:40:01 PBL INFO [CFG] D9 AE400-INIT_RVTRAN SXF0058I Creating RV
session. Hostcodepage= Networkcodepage=
2016Jan1823:40:01PBLINFO [CFG]D9AE400-PRS_CFG_FL SXF0180I
2016 Jan 18 23:40:01 PBL INFO [CFG] D9 AE400-PRS_CFG_FL SXF0181I [===] FILE TYPE
section:
2016 Jan 18 23:40:01 PBL INFO [CFG] D9 AE400-LOAD_FLOPT SXF0080I --> File Type
Options Begin: Num 0
2016Jan1823:40:01PBLINFO [CFG]D9AE400-LOAD_FLOPTSXF0081IFilePrefix:FT1
2016Jan1823:40:01PBLINFO [CFG]D9AE400-LOAD_FLOPTSXF0083IFileExtension:
2016 Jan 18 23:40:01 PBL INFO [CFG] D9 AE400-LOAD_FLOPT SXF0375I Transfer Type :
RecordMode
2016Jan1823:40:01PBLINFO [CFG]D9AE400-LOAD_FLOPTSXF0084IDataSetType:SEQ
2016 Jan 18 23:40:01 PBL INFO [CFG] D9 AE400-LOAD_FLOPT SXF0085I Use File Polling:
true
2016 Jan 18 23:40:01 PBL INFO [CFG] D9 AE400-LOAD_FLOPT SXF0086I Poll Interval
(milli-seconds): 5000
2016 Jan 18 23:40:01 PBL INFO [CFG] D9 AE400-LOAD_FLOPT SXF0087I
Subject/DestinationName: SUB.PUB001.FT1
2016 Jan 18 23:40:01 PBL INFO [CFG] D9 AE400-LOAD_FLOPT SXF0088I Start Publish
Subject/Destination:
2016 Jan 18 23:40:01 PBL INFO [CFG] D9 AE400-LOAD_FLOPT SXF0089I End Publish
Subject/Destination:
2016 Jan 18 23:40:01 PBL INFO [CFG] D9 AE400-LOAD_FLOPT SXF0090I Trigger
Subject/Destination Name:
2016 Jan 18 23:40:01 PBL INFO [CFG] D9 AE400-LOAD_FLOPT SXF0091I Trigger Field
Name:
2016 Jan 18 23:40:01 PBL INFO [CFG] D9 AE400-LOAD_FLOPT SXF0093I Input Directory:
TIBFA500R
2016 Jan 18 23:40:01 PBL INFO [CFG] D9 AE400-LOAD_FLOPT SXF0094I Process Directory:
FAPUBPROC
2016 Jan 18 23:40:01 PBL INFO [CFG] D9 AE400-LOAD_FLOPT SXF0095I Output Directory:
FAPUBOUT
2016 Jan 18 23:40:01 PBL INFO [CFG] D9 AE400-LOAD_FLOPT SXF0096I Publish Start
Message: false
2016 Jan 18 23:40:01 PBL INFO [CFG] D9 AE400-LOAD_FLOPT SXF0097I Publish End
Message: false
2016 Jan 18 23:40:01 PBL INFO [CFG] D9 AE400-LOAD_FLOPT SXF0098I Execute Before
Process:
2016 Jan 18 23:40:01 PBL INFO [CFG] D9 AE400-LOAD_FLOPT SXF0099I Execute After
Process:
2016 Jan 18 23:40:01 PBL INFO [CFG] D9 AE400-LOAD_FLOPT SXF0100I Remove After
Process: runJCL
2016 Jan 18 23:40:01 PBL INFO [CFG] D9 AE400-LOAD_FLOPT SXF0101I Messages Per
Transaction: 10
2016 Jan 18 23:40:01 PBL INFO [CFG] D9 AE400-LOAD_FLOPT SXF0102I Transaction Delay
(milli-seconds): 2000
2016 Jan 18 23:40:01 PBL INFO [CFG] D9 AE400-LOAD_FLOPT SXF0103I Block Transfer
Mode: false
2016 Jan 18 23:40:01 PBL INFO [CFG] D9 AE400-LOAD_FLOPT SXF0104I Block Transfer
Size: 65536
2016 Jan 18 23:40:01 PBL INFO [CFG] D9 AE400-LOAD_FLOPT SXF0105I Use Explicit
Confirmation: false
2016 Jan 18 23:40:01 PBL INFO [CFG] D9 AE400-LOAD_FLOPT SXF0112I Retransmission
Delay Tick Count: 10

```

```

2016 Jan 18 23:40:01 PBL INFO [CFG] D9 AE400-LOAD_FLOPT SXF0113I No Wait After
Confirmations: true
2016 Jan 18 23:40:01 PBL INFO [CFG] D9 AE400-LOAD_FLOPT SXF0114I Is Certified:
false
2016 Jan 18 23:40:01 PBL INFO [CFG] D9 AE400-LOAD_FLOPT SXF0115I RVCN Time Limit:
60
2016 Jan 18 23:40:01 PBL INFO [CFG] D9 AE400-LOAD_FLOPT SXF0116I Binary input file:
false
2016Jan1823:40:01PBLINFO[CFG]D9AE400-LOAD_FLOPTSX0118IlineLength:80
2016Jan1823:40:01PBLINFO[CFG]D9AE400-LOAD_FLOPTSX0135IDelimiter:|
2016Jan1823:40:01PBLINFO[CFG]D9AE400-LOAD_FLOPTSX0136IStartAtLine:1
2016 Jan 18 23:40:01 PBL INFO [CFG] D9 AE400-LOAD_FLOPT SXF0137I Use Field Width:
false
2016 Jan 18 23:40:01 PBL INFO [CFG] D9 AE400-LOAD_FLOPT SXF0138I Remove Leading
Blanks: false
2016 Jan 18 23:40:01 PBL INFO [CFG] D9 AE400-LOAD_FLOPT SXF0139I Remove Trailing
Blanks: false
2016 Jan 18 23:40:01 PBL INFO [CFG] D9 AE400-LOAD_FLOPT SXF0140I Use Sequential
Temp Files: true
2016 Jan 18 23:40:01 PBL INFO [CFG] D9 AE400-LOAD_FLOPT SXF0141I Default (Polling)
User ID:
2016 Jan 18 23:40:01 PBL INFO [CFG] D9 AE400-LOAD_FLOPT SXF0146I <-- File Type
Options End: Num 0
2016 Jan 18 23:40:01 PBL INFO [CFG] D9 AE400-LOD_MSG_FP SXF0158I ---> Message
Fields Begin: 0 [0]
2016 Jan 18 23:40:01 PBL INFO [CFG] D9 AE400-LD_MSG_IPM SXF0154I Item: 0 - Label:
PartNo, Type: STRING, Value: , Multiple: true
2016 Jan 18 23:40:01 PBL INFO [CFG] D9 AE400-LD_MSG_IPM SXF0154I Item: 1 - Label:
Desc, Type: STRING, Value: , Multiple: true
2016 Jan 18 23:40:01 PBL INFO [CFG] D9 AE400-LD_MSG_IPM SXF0154I Item: 2 - Label:
Manufacturer, Type: STRING, Value: , Multiple: true
2016 Jan 18 23:40:01 PBL INFO [CFG] D9 AE400-LD_MSG_IPM SXF0154I Item: 3 - Label:
Model, Type: STRING, Value: , Multiple: true
2016 Jan 18 23:40:01 PBL INFO [CFG] D9 AE400-LD_MSG_IPM SXF0154I Item: 4 - Label:
Quantity, Type: INTEGER, Value: , Multiple: true
2016 Jan 18 23:40:01 PBL INFO [CFG] D9 AE400-LD_MSG_IPM SXF0154I Item: 5 - Label:
Price, Type: FLOAT, Value: , Multiple: true
2016 Jan 18 23:40:01 PBL INFO [CFG] D9 AE400-LD_MSG_IPM SXF0154I Item: 6 - Label:
Warranty, Type: STRING, Value: 1 Year, Multiple: true
2016 Jan 18 23:40:01 PBL INFO [CFG] D9 AE400-LOD_MSG_FP SXF0159I <--- Message
Fields End: 0 [0]
2016Jan1823:40:01PBLINFO[CFG]D9AE400-PRS_CFG_FLSXF0180I
2016 Jan 18 23:40:01 PBL INFO [APP] D9 AE400-COMP_Q_FILE SXF1000I Job Queue
File=<FAPUBPROC/fpqueue>.
2016 Jan 18 23:40:01 PBL INFO [APP] D9 AE400-FAPUB_MAIN SXF1519I Setting up
terminateadapterSubject/Destination<_FILEADAPTER.FILE_PUB_001.TERMINATE>
2016 Jan 18 23:40:01 PBL INFO [APP] D9 AE400-INT_FL_PLR SXF1060I <FT1,> File
Pollingrequested. configValid=1 admin_SubCount=0
2016 Jan 18 23:40:01 PBL INFO [APP] D9 AE400-FAPUB_MAIN SXF1525I Completed
Initialization and any Recovery of Publisher.
2016 Jan 18 23:40:06 PBL INFO [APP] D9 AE400-SCH_JOBQ SXF1009I <FT1> Publishing
file: <FT11> withUserId=
2016 Jan 18 23:40:06 PBL INFO [APP] D9 AE400-PB_EPM_ADV SXF2324I <FT11> Publishing
EPM Advisory message using TrackingId= from Subject/Destination=SUB.PUB001.FT1 with
AdvisoryType=10
2016 Jan 18 23:40:06 PBL INFO [APP] D9 AE400-PUBLS_FILE SXF2060I <FT1> file=<FT11>
recov=0 Tracking ID=D9Y@@@JdrcJU@j1zzw0q@zzw

```

```

2016 Jan 18 23:40:06 PBL INFO  [APP] D9 AE400-MOVE_FILE SXF2002I Moving file: <FT11>
from<TIBFA500R> to <FAPUBPROC>
2016 Jan 18 23:40:06 PBL INFO  [APP] D9 AE400-MOVE_FILE SXF2003I Move file <FT11> to
directory <FAPUBPROC> completed successfully, and previous output
file=<FAPUBPROC/FT11> was deleted.
2016 Jan 18 23:40:06 PBL INFO  [APP] D9 AE400-PUBLS_FILE SXF2057I <FT1> SEQ/GDG/PDS
file=<FT11>DSName=<FAPUBPROC/FT11>.Dynamicrecl=80 bufsize=160 vb=1
2016 Jan 18 23:40:06 PBL INFO  [APP] D9 AE400-PB_EPM_MSG SXF2092I <FT1> <FT11>
Publishing status message using TrackingId=D9Y@@@JdrcJU@jllzzw0q@zzw from
Subject/Destination=SUB.PUB001.FT1 and JCL CC=0. epm=1
2016 Jan 18 23:40:06 PBL INFO  [APP] D9 AE400-PUBLS_FILE SXF2059I <FT1> Start
publishingusing PROGRESS file=<FAPUBPROC/PRGFT1>.
2016 Jan 18 23:40:06 PBL INFO  [APP] D9 AE400-PUBLS_FILE SXF2064I <FT1> Start
publishingRECORDMODE file=<FT11>tranDelaytimer=2000, ECM=0
2016 Jan 18 23:40:06 PBL INFO  [APP] D9 AE400-HDL_EOF_RECORD SXF2334I <FT11>
handleEOF RecordMode invoked. msgcont=0 eofsm=0 ewait=0 lrec=8 remove=0 haserr=0
remerr=0 execc=0 execcmd=<>
2016 Jan 18 23:40:06 PBL INFO  [APP] D9 AE400-PB_EPM_MSG SXF2092I <FT1> <FT11>
Publishing status message using TrackingId=D9Y@@@JdrcJU@jllzzw0q@zzw from
Subject/Destination=SUB.PUB001.FT1 and JCL CC=0. epm=1
2016 Jan 18 23:40:06 PBL INFO  [APP] D9 AE400-HDL_EOF_RECORD SXF2079I <FT1> Finished
publishing RECORDMODE file: <FT11>
2016 Jan 18 23:40:06 PBL INFO  [APP] D9 AE400-MOVE_FILE SXF2002I Moving file: <FT11>
from <FAPUBPROC> to <FAPUBOUT>
2016 Jan 18 23:40:06 PBL INFO  [APP] D9 AE400-MOVE_FILE SXF2003I Move file <FT11> to
directory <FAPUBOUT> completed successfully, and previous output
file=<FAPUBOUT/FT11> was deleted.
2016 Jan 18 23:40:06 PBL INFO  [APP] D9 AE400-HDL_EOF_RECORD SXF2083I Completed
processing file: <FT11>, no errors
2016 Jan 18 23:40:06 PBL INFO  [APP] D9 AE400-DN_PUB_CLB SXF1014I <FT1> File Done
Callback. Completed file: <FT11> using ECM=0.
2016 Jan 18 23:40:06 PBL INFO  [APP] D9 AE400-FIN_JOB SXF1001I FinishJob status:
NumConCurrJobs=0   MaxConcur=5   NumQueued=0   JobQ=0

```

## Failed FilePublisher Session

---

```

2016 Jan 19 00:11:34 PBL INFO [CFG] 30 AE400-INIT_TRACE SXF0050I Copyright (c)
1997-2016. TIBCO Software Inc.
2016 Jan 19 00:11:34 PBL INFO [CFG] 30 AE400-INIT_TRACE SXF0050I All Rights
Reserved. Confidential & Proprietary.
2016 Jan 19 00:11:34 PBL INFO [CFG] 30 AE400-INIT_TRACE SXF0051I
*****
2016 Jan 19 00:11:34 PBL INFO [CFG] 30 AE400-INIT_TRACE SXF0052I TIBCO Adapter for
Files (IBM i) start...
2016 Jan 19 00:11:34 PBL INFO [CFG] 30 AE400-INIT_TRACE SXF0053I File Adapter
Publisher 5.0.0 GAV06 [31151] Jan 7 2016
2016 Jan 19 00:11:34 PBL INFO [CFG] 30 AE400-INIT_TRACE SXF0055I Using
configuration file: <TIBFA500R/INIFPUB(PUB001)>
2016 Jan 19 00:11:34 PBL INFO [CFG] 30 AE400-INIT_TRACE SXF0355I Using trace file:
<TIBFA500R/PUBLOG(PUB001)> Size=2000000
2016 Jan 19 00:11:34 PBL INFO [CFG] 30 AE400-INIT_TRACE SXF0341I Trace Level: 2
TZ=N/A
2016 Jan 19 00:11:34 PBL INFO [CFG] 30 AE400-INIT_TRACE SXF0368I Trace Option
Switches: NNNNNNNN
2016 Jan 19 00:11:34 PBL INFO [CFG] 30 AE400-INIT_TRACE SXF0056I
*****
2016 Jan 19 00:11:34 PBL INFO [CFG] 30 AE400-PRS_CFG_FL SXF0179I [===] OPTIONS
section:
2016 Jan 19 00:11:34 PBL INFO [CFG] 30 AE400-LD_RV_PARM SXF0061I <--- RV_SESSION
Summary Begin
2016Jan1900:11:34PBLINFO[CFG]30AE400-LD_RV_PARMSXF0062IName:FILE_PUB_001
2016Jan1900:11:34PBLINFO[CFG]30AE400-LD_RV_PARMSXF0063IService:9721
2016Jan1900:11:34PBLINFO[CFG]30AE400-LD_RV_PARMSXF0064INetwork:
2016Jan1900:11:34PBLINFO[CFG]30AE400-LD_RV_PARMSXF0065IDaemon:tcp:9720
2016 Jan 19 00:11:34 PBL INFO [CFG] 30 AE400-LD_RV_PARM SXF0066I ----> RV_SESSION
Summary End
2016 Jan 19 00:11:34 PBL INFO [CFG] 30 AE400-PRS_OPT_AT SXF0160I Input Directory:
TIBFA500R
2016 Jan 19 00:11:34 PBL INFO [CFG] 30 AE400-PRS_OPT_AT SXF0161I Process Directory:
FAPUBPROC
2016 Jan 19 00:11:34 PBL INFO [CFG] 30 AE400-PRS_OPT_AT SXF0162I Output Directory:
FAPUBOUT
2016 Jan 19 00:11:34 PBL INFO [CFG] 30 AE400-PRS_OPT_AT SXF0175I Continue on
Configuration Error: true
2016 Jan 19 00:11:34 PBL INFO [CFG] 30 AE400-PRS_OPT_AT SXF0168I Max Concurrent
Jobs: 5
2016 Jan 19 00:11:34 PBL INFO [CFG] 30 AE400-PRS_OPT_AT SXF0172I Print File
Options: true
2016 Jan 19 00:11:34 PBL INFO [CFG] 30 AE400-PRS_OPT_AT SXF0177I Log to QSYSOPR
*MSGQ: true
2016 Jan 19 00:11:34 PBL INFO [CFG] 30 AE400-PRS_OPT_AT SXF0352I EPM
Subject/DestinationName: <EPM_SUBJECT>
2016 Jan 19 00:11:34 PBL INFO [CFG] 30 AE400-PRS_OPT_AT SXF0170I Error
Subject/Destination: ERROR.SUBJECT
2016 Jan 19 00:11:34 PBL INFO [CFG] 30 AE400-PRS_OPT_AT SXF0173I Error Exit
Condition Code: 1
2016 Jan 19 00:11:34 PBL INFO [CFG] 30 AE400-INIT_RVTRAN SXF0057I Initializing,
using RV client library version 8.4.4

```

```

2016 Jan 19 00:11:34 PBL INFO [CFG] 30 AE400-INIT_RVTRAN SXF0058I Creating RV
session. Hostcodepage= Networkcodepage=
2016Jan1900:11:34PBLINFO [CFG] 30AE400-PRS_CFG_FLSXF0180I
2016 Jan 19 00:11:34 PBL INFO [CFG] 30 AE400-PRS_CFG_FL SXF0181I [===] FILE TYPE
section:
2016 Jan 19 00:11:34 PBL INFO [CFG] 30 AE400-LOAD_FLOPT SXF0080I --> File Type
Options Begin: Num 0
2016Jan1900:11:34PBLINFO [CFG] 30AE400-LOAD_FLOPTSXF0081IFilePrefix:FT1
2016Jan1900:11:34PBLINFO [CFG] 30AE400-LOAD_FLOPTSXF0083IFileExtension:
2016 Jan 19 00:11:34 PBL INFO [CFG] 30 AE400-LOAD_FLOPT SXF0375I Transfer Type :
RecordMode
2016Jan1900:11:34PBLINFO [CFG] 30AE400-LOAD_FLOPTSXF0084IDataSetType:SEQ
2016 Jan 19 00:11:34 PBL INFO [CFG] 30 AE400-LOAD_FLOPT SXF0085I Use File Polling:
true
2016 Jan 19 00:11:34 PBL INFO [CFG] 30 AE400-LOAD_FLOPT SXF0086I Poll Interval
(milli-seconds): 5000
2016 Jan 19 00:11:34 PBL INFO [CFG] 30 AE400-LOAD_FLOPT SXF0087I
Subject/DestinationName: SUB.PUB001.FT1
2016 Jan 19 00:11:34 PBL INFO [CFG] 30 AE400-LOAD_FLOPT SXF0088I Start Publish
Subject/Destination:
2016 Jan 19 00:11:34 PBL INFO [CFG] 30 AE400-LOAD_FLOPT SXF0089I End Publish
Subject/Destination:
2016 Jan 19 00:11:34 PBL INFO [CFG] 30 AE400-LOAD_FLOPT SXF0090I Trigger
Subject/DestinationName:
2016 Jan 19 00:11:34 PBL INFO [CFG] 30 AE400-LOAD_FLOPT SXF0091I Trigger Field
Name:
2016 Jan 19 00:11:34 PBL INFO [CFG] 30 AE400-LOAD_FLOPT SXF0093I Input Directory:
TIBFA500R
2016 Jan 19 00:11:34 PBL INFO [CFG] 30 AE400-LOAD_FLOPT SXF0094I Process Directory:
FAPUBPROC
2016 Jan 19 00:11:34 PBL INFO [CFG] 30 AE400-LOAD_FLOPT SXF0095I Output Directory:
FAPUBOUT
2016 Jan 19 00:11:34 PBL INFO [CFG] 30 AE400-LOAD_FLOPT SXF0096I Publish Start
Message: false
2016 Jan 19 00:11:34 PBL INFO [CFG] 30 AE400-LOAD_FLOPT SXF0097I Publish End
Message: false
2016 Jan 19 00:11:34 PBL INFO [CFG] 30 AE400-LOAD_FLOPT SXF0098I Execute Before
Process:
2016 Jan 19 00:11:34 PBL INFO [CFG] 30 AE400-LOAD_FLOPT SXF0099I Execute After
Process:
2016 Jan 19 00:11:34 PBL INFO [CFG] 30 AE400-LOAD_FLOPT SXF0100I Remove After
Process: runJCL
2016 Jan 19 00:11:34 PBL INFO [CFG] 30 AE400-LOAD_FLOPT SXF0101I Messages Per
Transaction: 10
2016 Jan 19 00:11:34 PBL INFO [CFG] 30 AE400-LOAD_FLOPT SXF0102I Transaction Delay
(milli-seconds): 2000
2016 Jan 19 00:11:34 PBL INFO [CFG] 30 AE400-LOAD_FLOPT SXF0103I Block Transfer
Mode: false
2016 Jan 19 00:11:34 PBL INFO [CFG] 30 AE400-LOAD_FLOPT SXF0104I Block Transfer
Size: 65536
2016 Jan 19 00:11:34 PBL INFO [CFG] 30 AE400-LOAD_FLOPT SXF0105I Use Explicit
Confirmation: false
2016 Jan 19 00:11:34 PBL INFO [CFG] 30 AE400-LOAD_FLOPT SXF0112I Retransmission
Delay Tick Count: 10
2016 Jan 19 00:11:34 PBL INFO [CFG] 30 AE400-LOAD_FLOPT SXF0113I No Wait After
Confirmations: true

```

```

2016 Jan 19 00:11:34 PBL INFO [CFG] 30 AE400-LOAD_FLOPT SXF0114I Is Certified:
false
2016 Jan 19 00:11:34 PBL INFO [CFG] 30 AE400-LOAD_FLOPT SXF0115I RVCN Time Limit:
60
2016 Jan 19 00:11:34 PBL INFO [CFG] 30 AE400-LOAD_FLOPT SXF0116I Binary input file:
false
2016Jan1900:11:34PBLINFO[CFG]30AE400-LOAD_FLOPTSXF0118IlineLength:80
2016Jan1900:11:34PBLINFO[CFG]30AE400-LOAD_FLOPTSXF0135IDelimiter:|
2016Jan1900:11:34PBLINFO[CFG]30AE400-LOAD_FLOPTSXF0136IStartAtLine:1
2016 Jan 19 00:11:34 PBL INFO [CFG] 30 AE400-LOAD_FLOPT SXF0137I Use Field Width:
false
2016 Jan 19 00:11:34 PBL INFO [CFG] 30 AE400-LOAD_FLOPT SXF0138I Remove Leading
Blanks: false
2016 Jan 19 00:11:34 PBL INFO [CFG] 30 AE400-LOAD_FLOPT SXF0139I Remove Trailing
Blanks: false
2016 Jan 19 00:11:34 PBL INFO [CFG] 30 AE400-LOAD_FLOPT SXF0140I Use Sequential
Temp Files: true
2016 Jan 19 00:11:34 PBL INFO [CFG] 30 AE400-LOAD_FLOPT SXF0141I Default (Polling)
User ID:
2016 Jan 19 00:11:34 PBL INFO [CFG] 30 AE400-LOAD_FLOPT SXF0146I <-- File Type
Options End: Num 0
2016 Jan 19 00:11:34 PBL INFO [CFG] 30 AE400-LOD_MSG_FP SXF0158I ---> Message
Fields Begin: 0 [0]
2016 Jan 19 00:11:34 PBL INFO [CFG] 30 AE400-LD_MSG_IPM SXF0154I Item: 0 - Label:
PartNo, Type: STRING, Value: , Multiple: true
2016 Jan 19 00:11:34 PBL INFO [CFG] 30 AE400-LD_MSG_IPM SXF0154I Item: 1 - Label:
Desc, Type: STRING, Value: , Multiple: true
2016 Jan 19 00:11:34 PBL INFO [CFG] 30 AE400-LD_MSG_IPM SXF0154I Item: 2 - Label:
Manufacturer, Type: STRING, Value: , Multiple: true
2016 Jan 19 00:11:34 PBL INFO [CFG] 30 AE400-LD_MSG_IPM SXF0154I Item: 3 - Label:
Model, Type: STRING, Value: , Multiple: true
2016 Jan 19 00:11:34 PBL INFO [CFG] 30 AE400-LD_MSG_IPM SXF0154I Item: 4 - Label:
Quantity, Type: INTEGER, Value: , Multiple: true
2016 Jan 19 00:11:34 PBL INFO [CFG] 30 AE400-LD_MSG_IPM SXF0154I Item: 5 - Label:
Price, Type: FLOAT, Value: , Multiple: true
2016 Jan 19 00:11:34 PBL INFO [CFG] 30 AE400-LD_MSG_IPM SXF0154I Item: 6 - Label:
Warranty, Type: STRING, Value: 1Year, Multiple: true
2016 Jan 19 00:11:34 PBL INFO [CFG] 30 AE400-LOD_MSG_FP SXF0159I <--- Message
Fields End: 0 [0]
2016Jan1900:11:34PBLINFO[CFG]30AE400-PRS_CFG_FLSXF0180I
2016 Jan 19 00:11:34 PBL ERROR [CFG] 30 AE400-PRS_CFG_FL SXF0325E Missing
ADAPTER_NAME definition
fatal error, exiting app...

```

## Successful FileSubscriber Session

---

```

2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-INIT_TRACE SXF5050I Copyright (c)
1997-2016. TIBCO Software Inc.
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-INIT_TRACE SXF5050I All Rights
Reserved. Confidential & Proprietary.
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-INIT_TRACE SXF5051I
*****
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-INIT_TRACE SXF5052I TIBCO Adapter for
Files (IBMi) start...
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-INIT_TRACE SXF5053I File Adapter
Subscriber 5.0.0 GAV06 [31151] Jan 7 2016
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-INIT_TRACE SXF5055I Using
configuration file: <TIBFA500R/INIFSUB(SUB001)>
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-INIT_TRACE SXF5322I Using trace file:
<TIBFA500R/SUBLOG(SUB001)> Size=2000000
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-INIT_TRACE SXF5305I Trace Level: 2
TZ=N/A
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-INIT_TRACE SXF0368I Trace Option
Switches: NNNNNNNN
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-INIT_TRACE SXF5056I
*****
2016Jan1823:40:01SUBINFO [CFG] 127AE400-PRS_CFG_FL SXF5186I
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-PRS_CFG_FL SXF5187I [===] OPTIONS
section:
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LOAD_RVSPM SXF5061I <--- RV_SESSION
Summary Begin
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LOAD_RVSPM SXF5062I Name:
FILE_SUB_001
2016Jan1823:40:01SUBINFO [CFG] 127AE400-LOAD_RVSPMSXF5063I Service: 9721
2016Jan1823:40:01SUBINFO [CFG] 127AE400-LOAD_RVSPMSXF5064I Network:
2016Jan1823:40:01SUBINFO [CFG] 127AE400-LOAD_RVSPMSXF5065I Daemon: tcp:9720
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LOAD_RVSPM SXF5066I ----> RV_SESSION
Summary End
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-PRS_OPT_AT SXF5171I Adapter Name:
FILE_SUB_001
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-PRS_OPT_AT SXF5170I Output Directory:
FASUBOUT
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-PRS_OPT_AT SXF5182I Continue on
Configuration Error: true
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-PRS_OPT_AT SXF5179I Print File
Options: true
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-PRS_OPT_AT SXF5184I Terminate on RV
Send Error: true
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-PRS_OPT_AT SXF5185I Log to QSYSOPR
*MSGQ: true
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-PRS_OPT_AT SXF5321I EPM
Subject/Destination Name: <EPM_SUBJECT>
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-PRS_OPT_AT SXF5177I Error
Subject/Destination: ERROR.SUBJECT
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-PRS_OPT_AT SXF5180I Error Exit
Condition Code: 1
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-INIT_RVTRAN SXF0057I Initializing,
using RV client library version 8.4.4

```



```

2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-INIT_RVTRAN SXF0058I Creating RV
session. Hostcodepage= Networkcodepage=
2016Jan1823:40:01SUBINFO [CFG] 127AE400-PRS_CFG_FLSXF5186I
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-PRS_CFG_FL SXF5188I [===] FILE TYPE
section:
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LD_FOPT_PR SXF5080I --> File Type
Options Begin: Num 0
2016Jan1823:40:01SUBINFO [CFG] 127AE400-LD_FOPT_PR SXF5081I File Prefix: FT1
2016Jan1823:40:01SUBINFO [CFG] 127AE400-LD_FOPT_PR SXF5083I File Extension:
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LD_FOPT_PR SXF5327I Transfer Type :
RecordMode
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LD_FOPT_PR SXF5084I Data Set Type:
SEQ
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LD_FOPT_PR SXF5085I
Subject/DestinationName: SUB.PUB001.FT1
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LD_FOPT_PR SXF5087I Generate File
PublishSubject/Destination:
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LD_FOPT_PR SXF5088I Generate File On
Number of Messages: 0
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LD_FOPT_PR SXF5089I Publish Generate
File Message: false
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LD_FOPT_PR SXF5090I Save File
Interval (seconds): 100
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LD_FOPT_PR SXF5091I Auto Generate
File: true
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LD_FOPT_PR SXF5092I Generate File
FieldName: filename
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LD_FOPT_PR SXF5093I Process
Directory:
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LD_FOPT_PR SXF5095I Output Directory:
FASUBOUT
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LD_FOPT_PR SXF5096I Append Date/Time:
true
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LD_FOPT_PR SXF5097I Append File
sequence number: false
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LD_FOPT_PR SXF5098I Force published
filename: false
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LD_FOPT_PR SXF5099I Execute Before
Process:
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LD_FOPT_PR SXF5100I Execute After
Process:
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LD_FOPT_PR SXF5101I Block Transfer
Mode: false
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LD_FOPT_PR SXF5102I Block Transfer
Size: 65536
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LD_FOPT_PR SXF5103I Use Explicit
Confirmation: false
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LD_FOPT_PR SXF5104I Confirmation
Subject/Destination:
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LD_FOPT_PR SXF5109I Is Certified:
false
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LD_FOPT_PR SXF5110I Binary output
file: false
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LD_FOPT_PR SXF5111I Use Fixed Record
File: true
2016Jan1823:40:01SUBINFO [CFG] 127AE400-LD_FOPT_PR SXF5113I LineLength: 80

```



```

2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LD_FOPT_PR SXF5336I File Size Initial
Number of Records: *DEFAULT
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LD_FOPT_PR SXF5337I File Size Number
of Records to Increment: *DEFAULT
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LD_FOPT_PR SXF5338I File Size Maximum
Number of Increments: *DEFAULT
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LD_FOPT_PR SXF5118I Append To An
Existing File: false
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LD_FOPT_PR SXF5119I Exit On File Save
Error: true
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LD_FOPT_PR SXF5120I Max number of
retries for a locked Target file: 10
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LD_FOPT_PR SXF5121I Retry Interval
(milli-seconds): 5
2016Jan1823:40:01SUBINFO [CFG] 127AE400-LD_FOPT_PR SXF5142I Delimiter: |
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LD_FOPT_PR SXF5143I Pad Character: '
'
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LD_FOPT_PR SXF5144I Pad Direction:
right
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LD_FOPT_PR SXF5145I Skip Padding:
true
2016Jan1823:40:01SUBINFO [CFG] 127AE400-LD_FOPT_PR SXF5146I File Header:
2016Jan1823:40:01SUBINFO [CFG] 127AE400-LD_FOPT_PR SXF5147I File Trailer:
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LD_FOPT_PR SXF5153I <-- File Type
Options End: Num 0
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LD_FLN_PRM SXF5154I <----> Begin Line
Fields: 0 [0]
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LD_FLN_PRM SXF5156I --> Begin Line
Field: 0
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LD_FLN_PRM SXF5157I Description: Part
Number
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LD_FLN_PRM SXF5158I From Message:
true
2016Jan1823:40:01SUBINFO [CFG] 127AE400-LD_FLN_PRM SXF5160I Position: 1
2016Jan1823:40:01SUBINFO [CFG] 127AE400-LD_FLN_PRM SXF5161I Length: 0
2016Jan1823:40:01SUBINFO [CFG] 127AE400-LD_FLN_PRM SXF5162I Precision:
2016Jan1823:40:01SUBINFO [CFG] 127AE400-LD_FLN_PRM SXF5163I Type: STRING
2016Jan1823:40:01SUBINFO [CFG] 127AE400-LD_FLN_PRM SXF5164I Value: <PartNo>
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LD_FLN_PRM SXF5165I Pad Character: '
'
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LD_FLN_PRM SXF5166I Pad Direction:
right
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LD_FLN_PRM SXF5167I <-- End Line
Field: 0
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LD_FLN_PRM SXF5156I --> Begin Line
Field: 1
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LD_FLN_PRM SXF5157I Description: Part
Description
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LD_FLN_PRM SXF5158I From Message:
true
2016Jan1823:40:01SUBINFO [CFG] 127AE400-LD_FLN_PRM SXF5160I Position: 2
2016Jan1823:40:01SUBINFO [CFG] 127AE400-LD_FLN_PRM SXF5161I Length: 0
2016Jan1823:40:01SUBINFO [CFG] 127AE400-LD_FLN_PRM SXF5162I Precision:
2016Jan1823:40:01SUBINFO [CFG] 127AE400-LD_FLN_PRM SXF5163I Type: STRING
2016Jan1823:40:01SUBINFO [CFG] 127AE400-LD_FLN_PRM SXF5164I Value: <Desc>
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LD_FLN_PRM SXF5165I Pad Character: '
'

```

```

2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LD_FLN_PRM SXF5156I Pad Direction:
right
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LD_FLN_PRM SXF5167I <-- End Line
Field: 1
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LD_FLN_PRM SXF5156I --> Begin Line
Field: 2
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LD_FLN_PRM SXF5157I Description:
Manufacturer
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LD_FLN_PRM SXF5158I From Message:
true
2016Jan1823:40:01SUBINFO [CFG] 127AE400-LD_FLN_PRMSXF5160IPosition:3
2016Jan1823:40:01SUBINFO [CFG] 127AE400-LD_FLN_PRMSXF5161ILength:0
2016Jan1823:40:01SUBINFO [CFG] 127AE400-LD_FLN_PRMSXF5162IPrecision:
2016Jan1823:40:01SUBINFO [CFG] 127AE400-LD_FLN_PRMSXF5163IType:STRING
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LD_FLN_PRM SXF5164I Value:
<Manufacturer>
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LD_FLN_PRM SXF5165I Pad Character: '
'
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LD_FLN_PRM SXF5166I Pad Direction:
right
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LD_FLN_PRM SXF5167I <-- End Line
Field: 2
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LD_FLN_PRM SXF5156I --> Begin Line
Field: 3
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LD_FLN_PRM SXF5157I Description:
Model
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LD_FLN_PRM SXF5158I From Message:
true
2016Jan1823:40:01SUBINFO [CFG] 127AE400-LD_FLN_PRMSXF5160IPosition:4
2016Jan1823:40:01SUBINFO [CFG] 127AE400-LD_FLN_PRMSXF5161ILength:0
2016Jan1823:40:01SUBINFO [CFG] 127AE400-LD_FLN_PRMSXF5162IPrecision:
2016Jan1823:40:01SUBINFO [CFG] 127AE400-LD_FLN_PRMSXF5163IType:STRING
2016Jan1823:40:01SUBINFO [CFG] 127AE400-LD_FLN_PRMSXF5164IValue:<Model>
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LD_FLN_PRM SXF5165I Pad Character: '
'
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LD_FLN_PRM SXF5166I Pad Direction:
right
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LD_FLN_PRM SXF5167I <-- End Line
Field: 3
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LD_FLN_PRM SXF5156I --> Begin Line
Field: 4
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LD_FLN_PRM SXF5157I Description:
Quantity
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LD_FLN_PRM SXF5158I From Message:
true
2016Jan1823:40:01SUBINFO [CFG] 127AE400-LD_FLN_PRMSXF5160IPosition:5
2016Jan1823:40:01SUBINFO [CFG] 127AE400-LD_FLN_PRMSXF5161ILength:1
2016Jan1823:40:01SUBINFO [CFG] 127AE400-LD_FLN_PRMSXF5162IPrecision:
2016Jan1823:40:01SUBINFO [CFG] 127AE400-LD_FLN_PRMSXF5163IType:INTEGER
2016Jan1823:40:01SUBINFO [CFG] 127AE400-LD_FLN_PRMSXF5164IValue:<Quantity>
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LD_FLN_PRM SXF5165I Pad Character: '
'
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LD_FLN_PRM SXF5166I Pad Direction:
right
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LD_FLN_PRM SXF5167I <-- End Line
Field: 4

```

```

2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LD_FLN_PRM SXF5156I --> Begin Line
Field: 5
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LD_FLN_PRM SXF5157I Description: Unit
Price
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LD_FLN_PRM SXF5158I From Message:
true
2016Jan1823:40:01SUBINFO[CFG]127AE400-LD_FLN_PRMSXF5160IPosition:6
2016Jan1823:40:01SUBINFO[CFG]127AE400-LD_FLN_PRMSXF5161ILength:1
2016Jan1823:40:01SUBINFO[CFG]127AE400-LD_FLN_PRMSXF5162IPrecision:
2016Jan1823:40:01SUBINFO[CFG]127AE400-LD_FLN_PRMSXF5163IType:FLOAT
2016Jan1823:40:01SUBINFO[CFG]127AE400-LD_FLN_PRMSXF5164IValue:<Price>
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LD_FLN_PRM SXF5165I Pad Character: '
'
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LD_FLN_PRM SXF5166I Pad Direction:
right
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LD_FLN_PRM SXF5167I <-- End Line
Field: 5
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LD_FLN_PRM SXF5156I --> Begin Line
Field: 6
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LD_FLN_PRM SXF5157I Description:
Warranty Period in years
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LD_FLN_PRM SXF5158I From Message:
true
2016Jan1823:40:01SUBINFO[CFG]127AE400-LD_FLN_PRMSXF5160IPosition:7
2016Jan1823:40:01SUBINFO[CFG]127AE400-LD_FLN_PRMSXF5161ILength:0
2016Jan1823:40:01SUBINFO[CFG]127AE400-LD_FLN_PRMSXF5162IPrecision:
2016Jan1823:40:01SUBINFO[CFG]127AE400-LD_FLN_PRMSXF5163IType:STRING
2016Jan1823:40:01SUBINFO[CFG]127AE400-LD_FLN_PRMSXF5164IValue:<Warranty>
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LD_FLN_PRM SXF5165I Pad Character: '
'
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LD_FLN_PRM SXF5166I Pad Direction:
right
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LD_FLN_PRM SXF5167I <-- End Line
Field: 6
2016 Jan 18 23:40:01 SUB INFO [CFG] 127 AE400-LD_FLN_PRM SXF5168I <----- Line
Fields: 0 [0]
2016Jan1823:40:01SUBINFO[CFG]127AE400-PRS_CFG_FLSXF5186I
2016 Jan 18 23:40:01 SUB INFO [APP] 127 AE400-FASUB_MAIN SXF6011I Setting up
terminateadapterSubject/Destination<_FILEADAPTER.FILE_SUB_001.TERMINATE>
2016 Jan 18 23:40:01 SUB INFO [APP] 127 AE400-READ_ERR_ST SXF7373I <FASUBOUT/WKFT1>
STERR num file=FASUBOUT/STERRFT1 does not exist for Prefix=FT1. Will create it if
needed.
2016 Jan 18 23:40:01 SUB INFO [APP] 127 AE400-SET_SUBSCRS SXF7113I <SUB.PUB001.FT1>
SettingupsubscribetoSubject/DestinationforFileTypePrefix=FT1
2016 Jan 18 23:40:01 SUB INFO [APP] 127 AE400-FASUB_MAIN SXF6016I Completed
InitializationandanyRecoveryofSubscriber.
2016 Jan 18 23:40:07 SUB INFO [APP] 127 AE400-OPN_FILES SXF7009I <FASUBOUT/WKFT1>
TEXTfilelineLng=80recl=80blksize=320.Fullname=NT_flldata_returned_filename
2016 Jan 18 23:41:41 SUB INFO [APP] 127 AE400-GEN_OUT_FL SXF7020I <FASUBOUT/WKFT1>
Generating file for FileType=FT1 - for criterion=auto generate timer. nl=8 Blk=0.
errflags=0:0 Caller=saveTimerCallback
2016 Jan 18 23:41:41 SUB INFO [APP] 127 AE400-GEN_OUT_FL SXF7028I <FT1>
<FASUBOUT/WKFT1> Closed working file
2016 Jan 18 23:41:41 SUB INFO [APP] 127 AE400-GEN_OUT_FL SXF7038I <FT1>
<FASUBOUT/WKFT1> Generating target file=<FASUBOUT/FT1234141> 8 lines, 8 messages,
no errors

```

```
2016 Jan 18 23:41:41 SUB INFO [APP] 127 AE400-GEN_OUT_FL SXF7044I <FT1>
appendToExisting=False,soRemovedPREVoutput file=<FASUBOUT/FT1234141>
2016 Jan 18 23:41:41 SUB INFO [APP] 127 AE400-GEN_OUT_FL SXF7399I <FASUBOUT/WKFT1>
work file is being renamed to <FASUBOUT/FT1234141>.
2016 Jan 18 23:41:41 SUB INFO [APP] 127 AE400-GEN_OUT_PP SXF7053I <FASUBOUT/PRGFT1>
Progress file has been closed. EOF reason=auto generate timer
2016 Jan 18 23:41:41 SUB INFO [APP] 127 AE400-GEN_OUT_PP SXF7317W <FT1> FileType -
Deleting PRG file
```

## Failed FileSubscriber Session

---

```

2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-INIT_TRACE SXF5050I Copyright (c)
1997-2016. TIBCO Software Inc.
2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-INIT_TRACE SXF5050I All Rights
Reserved. Confidential & Proprietary.
2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-INIT_TRACE SXF5051I
*****
2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-INIT_TRACE SXF5052I TIBCO Adapter for
Files (IBMi) start...
2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-INIT_TRACE SXF5053I File Adapter
Subscriber 5.0.0 GA V06 [31151] Jan 7 2016
2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-INIT_TRACE SXF5055I Using
configuration file: <TIBFA500R/INIFSUB(SUB001)>
2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-INIT_TRACE SXF5322I Using trace file:
<TIBFA500R/SUBLOG(SUB001)> Size=2000000
2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-INIT_TRACE SXF5305I Trace Level: 2
TZ=N/A
2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-INIT_TRACE SXF0368I Trace Option
Switches: NNNNNNNN
2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-INIT_TRACE SXF5056I
*****
2016Jan1900:11:34SUBINFO [CFG] 16AE400-PRS_CFG_FLSXF5186I
2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-PRS_CFG_FL SXF5187I [===] OPTIONS
section:
2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-LOAD_RVSPM SXF5061I <--- RV_SESSION
Summary Begin
2016Jan1900:11:34SUBINFO [CFG] 16AE400-LOAD_RVSPMSXF5062IName: FILE_SUB_001
2016Jan1900:11:34SUBINFO [CFG] 16AE400-LOAD_RVSPMSXF5063IService: 9721
2016Jan1900:11:34SUBINFO [CFG] 16AE400-LOAD_RVSPMSXF5064INetwork:
2016Jan1900:11:34SUBINFO [CFG] 16AE400-LOAD_RVSPMSXF5065IDaemon: tcp: 9720
2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-LOAD_RVSPM SXF5066I ----> RV_SESSION
Summary End
2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-PRS_OPT_AT SXF5170I Output Directory:
FASUBOUT
2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-PRS_OPT_AT SXF5182I Continue on
Configuration Error: true
2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-PRS_OPT_AT SXF5179I Print File
Options: true
2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-PRS_OPT_AT SXF5184I Terminate on RV
Send Error: true
2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-PRS_OPT_AT SXF5185I Log to QSYSOPR
*MSGQ: true
2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-PRS_OPT_AT SXF5321I EPM
Subject/DestinationName: <EPM_SUBJECT>
2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-PRS_OPT_AT SXF5177I Error
Subject/Destination: ERROR.SUBJECT
2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-PRS_OPT_AT SXF5180I Error Exit
Condition Code: 1
2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-INIT_RVTRAN SXF0057I Initializing,
using RV client library version 8.4.4
2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-INIT_RVTRAN SXF0058I Creating RV
session. Hostcodepage= Networkcodepage=
2016Jan1900:11:34SUBINFO [CFG] 16AE400-PRS_CFG_FLSXF5186I

```

```

2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-PRS_CFG_FL SXF5188I [===] FILE TYPE
section:
2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-LD_FOPT_PR SXF5080I --> File Type
Options Begin: Num 0
2016Jan1900:11:34SUBINFO [CFG] 16AE400-LD_FOPT_PR SXF5081I File Prefix: FT1
2016Jan1900:11:34SUBINFO [CFG] 16AE400-LD_FOPT_PR SXF5083I File Extension:
2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-LD_FOPT_PR SXF5327I Transfer Type :
RecordMode
2016Jan1900:11:34SUBINFO [CFG] 16AE400-LD_FOPT_PR SXF5084I DataSetType: SEQ
2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-LD_FOPT_PR SXF5085I
Subject/DestinationName: SUB.PUB001.FT1
2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-LD_FOPT_PR SXF5087I Generate File
Publish Subject/Destination:
2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-LD_FOPT_PR SXF5088I Generate File On
Number of Messages: 0
2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-LD_FOPT_PR SXF5089I Publish Generate
File Message: false
2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-LD_FOPT_PR SXF5090I Save File Interval
(seconds): 100
2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-LD_FOPT_PR SXF5091I Auto Generate
File: true
2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-LD_FOPT_PR SXF5092I Generate File
Field Name: filename
2016Jan1900:11:34SUBINFO [CFG] 16AE400-LD_FOPT_PR SXF5093I ProcessDirectory:
2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-LD_FOPT_PR SXF5095I Output Directory:
FASUBOUT
2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-LD_FOPT_PR SXF5096I Append Date/Time:
true
2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-LD_FOPT_PR SXF5097I Append File
sequence number: false
2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-LD_FOPT_PR SXF5098I Force published
filename: false
2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-LD_FOPT_PR SXF5099I Execute Before
Process:
2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-LD_FOPT_PR SXF5100I Execute After
Process:
2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-LD_FOPT_PR SXF5101I Block Transfer
Mode: false
2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-LD_FOPT_PR SXF5102I Block Transfer
Size: 65536
2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-LD_FOPT_PR SXF5103I Use Explicit
Confirmation: false
2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-LD_FOPT_PR SXF5104I Confirmation
Subject/Destination:
2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-LD_FOPT_PR SXF5109I Is Certified:
false
2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-LD_FOPT_PR SXF5110I Binary output
file: false
2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-LD_FOPT_PR SXF5111I Use Fixed Record
File: true
2016Jan1900:11:34SUBINFO [CFG] 16AE400-LD_FOPT_PR SXF5113I LineLength: 80
2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-LD_FOPT_PR SXF5336I File Size Initial
Number of Records: *DEFAULT
2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-LD_FOPT_PR SXF5337I File Size Number
of Records to Increment: *DEFAULT
2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-LD_FOPT_PR SXF5338I File Size Maximum
Number of Increments: *DEFAULT

```

```

2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-LD_FOPT_PR SXF5118I Append To An
Existing File: false
2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-LD_FOPT_PR SXF5119I Exit On File Save
Error: true
2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-LD_FOPT_PR SXF5120I Max number of
retries for a locked Target file: 10
2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-LD_FOPT_PR SXF5121I Retry Interval
(milli-seconds): 5
2016Jan1900:11:34SUBINFO[CFG]16AE400-LD_FOPT_PR SXF5142IDelimiter:|
2016Jan1900:11:34SUBINFO[CFG]16AE400-LD_FOPT_PR SXF5143IPadCharacter:''
2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-LD_FOPT_PR SXF5144I Pad Direction:
right
2016Jan1900:11:34SUBINFO[CFG]16AE400-LD_FOPT_PR SXF5145ISkipPadding:true
2016Jan1900:11:34SUBINFO[CFG]16AE400-LD_FOPT_PR SXF5146IFileHeader:
2016Jan1900:11:34SUBINFO[CFG]16AE400-LD_FOPT_PR SXF5147IFileTrailer:
2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-LD_FOPT_PR SXF5153I <-- File Type
Options End: Num 0
2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-LD_FLN_PRM SXF5154I -----> Begin Line
Fields: 0 [0]
2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-LD_FLN_PRM SXF5156I --> Begin Line
Field: 0
2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-LD_FLN_PRM SXF5157I Description: Part
Number
2016Jan1900:11:34SUBINFO[CFG]16AE400-LD_FLN_PRMSXF5158IFromMessage:true
2016Jan1900:11:34SUBINFO[CFG]16AE400-LD_FLN_PRMSXF5160IPosition:1
2016Jan1900:11:34SUBINFO[CFG]16AE400-LD_FLN_PRMSXF5161ILength:0
2016Jan1900:11:34SUBINFO[CFG]16AE400-LD_FLN_PRMSXF5162IPrecision:
2016Jan1900:11:34SUBINFO[CFG]16AE400-LD_FLN_PRMSXF5163IType:STRING
2016Jan1900:11:34SUBINFO[CFG]16AE400-LD_FLN_PRMSXF5164IValue:<PartNo>
2016Jan1900:11:34SUBINFO[CFG]16AE400-LD_FLN_PRMSXF5165IPadCharacter:''
2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-LD_FLN_PRM SXF5166I Pad Direction:
right
2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-LD_FLN_PRM SXF5167I <-- End Line
Field: 0
2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-LD_FLN_PRM SXF5156I --> Begin Line
Field: 1
2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-LD_FLN_PRM SXF5157I Description: Part
Description
2016Jan1900:11:34SUBINFO[CFG]16AE400-LD_FLN_PRMSXF5158IFromMessage:true
2016Jan1900:11:34SUBINFO[CFG]16AE400-LD_FLN_PRMSXF5160IPosition:2
2016Jan1900:11:34SUBINFO[CFG]16AE400-LD_FLN_PRMSXF5161ILength:0
2016Jan1900:11:34SUBINFO[CFG]16AE400-LD_FLN_PRMSXF5162IPrecision:
2016Jan1900:11:34SUBINFO[CFG]16AE400-LD_FLN_PRMSXF5163IType:STRING
2016Jan1900:11:34SUBINFO[CFG]16AE400-LD_FLN_PRMSXF5164IValue:<Desc>
2016Jan1900:11:34SUBINFO[CFG]16AE400-LD_FLN_PRMSXF5165IPadCharacter:''
2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-LD_FLN_PRM SXF5166I Pad Direction:
right
2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-LD_FLN_PRM SXF5167I <-- End Line
Field: 1
2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-LD_FLN_PRM SXF5156I --> Begin Line
Field: 2
2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-LD_FLN_PRM SXF5157I Description:
Manufacturer
2016Jan1900:11:34SUBINFO[CFG]16AE400-LD_FLN_PRMSXF5158IFromMessage:true
2016Jan1900:11:34SUBINFO[CFG]16AE400-LD_FLN_PRMSXF5160IPosition:3
2016Jan1900:11:34SUBINFO[CFG]16AE400-LD_FLN_PRMSXF5161ILength:0
2016Jan1900:11:34SUBINFO[CFG]16AE400-LD_FLN_PRMSXF5162IPrecision:

```



```

2016Jan1900:11:34SUBINFO [CFG] 16AE400-LD_FLN_PRMSXF5163IType:STRING
2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-LD_FLN_PRMSXF5164I Value:
<Manufacturer>
2016Jan1900:11:34SUBINFO [CFG] 16AE400-LD_FLN_PRMSXF5165IPadCharacter:''
2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-LD_FLN_PRMSXF5166I Pad Direction:
right
2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-LD_FLN_PRMSXF5167I <-- End Line
Field: 2
2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-LD_FLN_PRMSXF5168I --> Begin Line
Field: 3
2016Jan1900:11:34SUBINFO [CFG] 16AE400-LD_FLN_PRMSXF5169IDescription:Model
2016Jan1900:11:34SUBINFO [CFG] 16AE400-LD_FLN_PRMSXF5170IFromMessage:true
2016Jan1900:11:34SUBINFO [CFG] 16AE400-LD_FLN_PRMSXF5171IPosition:4
2016Jan1900:11:34SUBINFO [CFG] 16AE400-LD_FLN_PRMSXF5172IILength:0
2016Jan1900:11:34SUBINFO [CFG] 16AE400-LD_FLN_PRMSXF5173IPrecision:
2016Jan1900:11:34SUBINFO [CFG] 16AE400-LD_FLN_PRMSXF5174IType:STRING
2016Jan1900:11:34SUBINFO [CFG] 16AE400-LD_FLN_PRMSXF5175IValue:<Model>
2016Jan1900:11:34SUBINFO [CFG] 16AE400-LD_FLN_PRMSXF5176IPadCharacter:''
2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-LD_FLN_PRMSXF5177I Pad Direction:
right
2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-LD_FLN_PRMSXF5178I <-- End Line
Field: 3
2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-LD_FLN_PRMSXF5179I --> Begin Line
Field: 4
2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-LD_FLN_PRMSXF5180I Description:
Quantity
2016Jan1900:11:34SUBINFO [CFG] 16AE400-LD_FLN_PRMSXF5181IFromMessage:true
2016Jan1900:11:34SUBINFO [CFG] 16AE400-LD_FLN_PRMSXF5182IPosition:5
2016Jan1900:11:34SUBINFO [CFG] 16AE400-LD_FLN_PRMSXF5183IILength:1
2016Jan1900:11:34SUBINFO [CFG] 16AE400-LD_FLN_PRMSXF5184IPrecision:
2016Jan1900:11:34SUBINFO [CFG] 16AE400-LD_FLN_PRMSXF5185IType:INTEGER
2016Jan1900:11:34SUBINFO [CFG] 16AE400-LD_FLN_PRMSXF5186IValue:<Quantity>
2016Jan1900:11:34SUBINFO [CFG] 16AE400-LD_FLN_PRMSXF5187IPadCharacter:''
2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-LD_FLN_PRMSXF5188I Pad Direction:
right
2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-LD_FLN_PRMSXF5189I <-- End Line
Field: 4
2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-LD_FLN_PRMSXF5190I --> Begin Line
Field: 5
2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-LD_FLN_PRMSXF5191I Description: Unit
Price
2016Jan1900:11:34SUBINFO [CFG] 16AE400-LD_FLN_PRMSXF5192IFromMessage:true
2016Jan1900:11:34SUBINFO [CFG] 16AE400-LD_FLN_PRMSXF5193IPosition:6
2016Jan1900:11:34SUBINFO [CFG] 16AE400-LD_FLN_PRMSXF5194IILength:1
2016Jan1900:11:34SUBINFO [CFG] 16AE400-LD_FLN_PRMSXF5195IPrecision:
2016Jan1900:11:34SUBINFO [CFG] 16AE400-LD_FLN_PRMSXF5196IType:FLOAT
2016Jan1900:11:34SUBINFO [CFG] 16AE400-LD_FLN_PRMSXF5197IValue:<Price>
2016Jan1900:11:34SUBINFO [CFG] 16AE400-LD_FLN_PRMSXF5198IPadCharacter:''
2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-LD_FLN_PRMSXF5199I Pad Direction:
right
2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-LD_FLN_PRMSXF5200I <-- End Line
Field: 5
2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-LD_FLN_PRMSXF5201I --> Begin Line
Field: 6
2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-LD_FLN_PRMSXF5202I Description:
Warranty Period in years
2016Jan1900:11:34SUBINFO [CFG] 16AE400-LD_FLN_PRMSXF5203IFromMessage:true

```



```

2016Jan1900:11:34SUBINFO [CFG] 16AE400-LD_FLN_PRMSXF5160IPosition:7
2016Jan1900:11:34SUBINFO [CFG] 16AE400-LD_FLN_PRMSXF5161ILength:0
2016Jan1900:11:34SUBINFO [CFG] 16AE400-LD_FLN_PRMSXF5162IPrecision:
2016Jan1900:11:34SUBINFO [CFG] 16AE400-LD_FLN_PRMSXF5163IType:STRING
2016Jan1900:11:34SUBINFO [CFG] 16AE400-LD_FLN_PRMSXF5164IValue:<Warranty>
2016Jan1900:11:34SUBINFO [CFG] 16AE400-LD_FLN_PRMSXF5165IPadCharacter:''
2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-LD_FLN_PRM SXF5166I Pad Direction:
right
2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-LD_FLN_PRM SXF5167I <-- End Line
Field: 6
2016 Jan 19 00:11:34 SUB INFO [CFG] 16 AE400-LD_FLN_PRM SXF5168I <----- Line
Fields: 0 [0]
2016Jan1900:11:34SUBINFO [CFG] 16AE400-PRS_CFG_FLSXF5186I
2016 Jan 19 00:11:34 SUB ERROR [CFG] 16 AE400-PRS_CFG_FL SXF5287E Missing
ADAPTER_NAME definition
fatal error, exiting app...

```



## Appendix B   **Error Messages**

This appendix describes the error messages used by Adapter.

### Topics

---

- [Error Message Format, page 160](#)
- [Publisher Error Messages, page 161](#)
- [Subscriber Error Messages, page 216](#)

## Error Message Format

---

Messages that can appear in trace logs contain several repetitive preamble fields that you can ignore. These preamble fields are omitted from the messages in this appendix. For example, a complete message in a trace log is shown in this appendix as follows:

```
2004 March 5 19:24:03 PBL ERROR [CFG] SXF7145E HDL_TEXT_ERR <%s> reached...
```

Message in this appendix—  
starts here

Adapter produces several categories of messages. The last letter of the error number, I, W, or E, indicates the type of message.

- Informational (INFO) messages confirm that an event, task, operation, or condition is normal. No action is necessary; therefore, INFO messages are omitted from this appendix.
- Warning messages indicate that there is a problem but that processing continues.
- Error messages indicate a problem so severe that processing cannot continue for that file type.



A % symbol indicates a C type substitution. For example, %s indicates that the error message includes a string value as a substitute for the %s value.

## Publisher Error Messages

### SXF0000 – SXF0500

Number	Message	Description
SXF0133W	VSAM RLS Share Options 3 and 4 are not supported	You might specify a SHARE option that the FileAdapter does not support. Only levels 1 and 2 are supported.
SXF0192W	<Delimiter> can not be used when useFieldWidth is true	You have specified both a delimiter character and useFieldWidth=true. Use only one of the parameters for each file type.
SXF0193W	Invalid messagesPerTransaction value <%d>	The parameter was specified with an invalid value. Correct the value. Ensure that the value is greater than zero.
SXF0194W	messagesPerTransaction set to default value: <%d>	No value was specified in the configuration file for this parameter. Add this parameter if the default value is not the value you want to be used.
SXF0195W	Not a Certified Session. Cannot pre-register listener %s for subject %s	You might specify pre-registration, but the session is not a certified session. Either specify a certified session, or omit the pre-registration entries.
SXF0196W	Dataset not found: %s File not found: %s	The file indicated was not found. Verify that the file exists and is available to FilePublisher.
SXF0197W	Library not found: %s	The library indicated was not found. Verify that the dataset file exists and is available to FilePublisher.
SXF0198W	Directory not found: %s	The specified directory was not found. Make sure that the specified directory exists and is available.

Number	Message	Description
SXF0199E	Config parse error near line %d, failed to parse value	The format or value for a name/tag was not correct. A supplied value is incomplete; typically the value is missing an opening or closing quote ("), brace ({}), or an equal sign (=). Make sure all values in the configuration file have matching quotes and braces, and that there are equal signs where required. Check the name/value tags in the section defined in the configuration file. Refer to the manual for the proper format.
SXF0200E	Supplied VSAM start key is longer than 255 bytes	VSAM limits keys to 255 bytes. Go re-check the length of your key, because it exceeds VSAM's maximum length
SXF0201E	Supplied VSAM end key is longer than 255 bytes	VSAM limits keys to 255 bytes. Go re-check the length of your key, because it exceeds VSAM's maximum length
SXF0202E	Supplied VSAM start key is longer than 510 nibbles	VSAM limits keys to 255 bytes (510 bytes in HEXADECIMAL nibbles). Go re-check the length of your key, because it exceeds VSAM's maximum length
SXF0203E	Supplied VSAM end Hex key is longer than 510 nibbles	VSAM limits keys to 255 bytes (510 bytes in HEXADECIMAL nibbles). Go re-check the length of your key, because it exceeds VSAM's maximum length
SXF0204E	Max record COUNT must be > 0	A Max record count of zero was invalid. It says you want zero records from the file, which is meaningless.
SXF0205E	Invalid vsamShare= option. Must be either NRI or CR	Only the VSMA NRI and CR options are supported by the File Adapter.
SXF0206E	AltIndex name too long. Max of 44 bytes	VSAM limits AltIndex names to 44 characters. Go re-check you parameter.
SXF0207E	Duplicate retransmissionDelay entry. Ignored !	The retransmissionDelay was previously specified. A duplicate entry was encountered, and ignored.

Number	Message	Description
SXF0208E	Invalid value set for %s	The %s contained the name of the FileAdapter parameter that had been set to an invalid value in the FileAdapter INI file.
SXF0209E	Unable to allocate memory for ECMSubscriber entry	An out of memory error was reported. Raise the Job Limits (REGION size) to provide more memory for the FileAdapter to run.
SXF0210E	Config error: ECMSubscriber defined without useExplicitConfirmation=true. ECMSubscriber ignored.	A mismatch in the configuration had been configured. Either also turn on the useExplicitConfirmation=true parameter, or eliminate the ECMSubscriber definition.
SXF0211E	Unknown/Invalid configuration parameter <%s> at line %d. Ignored	The %s contained the name of the FileAdapter parameter that had been set to an invalid value in the FileAdapter INI file, and the Line number indicated which FileType or other definition contains the parameter in error.
SXF0212E	Invalid file type: prefix more than 8 char	Names used in File type PREFIX= definitions are limited to 8 bytes in length.
SXF0213E	Unsupported dataSetType <%s>	The dataSetType= parameter can only be SEQ.
SXF0214E	Starting RRN / RBA must be greater than 0	A VSAM RRN was to a negative value, which is invalid. A VSAM RRN must be zero or larger.
SXF0215E	Ending RRN / RBA must be greater than 0	A VSAM ending RRN must be greater than zero, otherwise no records in the file can be processed.
SXF0216E	AltIndex filenames only allowed for KSDS PATHs	VSAM only allows AltIndex names to be used if the file is defined as a KSDS dataset.
SXF0217E	Missing parameter <%s>	The %s contained the name of the FileAdapter parameter that is missing from the FileAdapter INI file.
SXF0218E	Config ERROR: you have defined Record ECM and Block Mode ECM in the same Filetype. Config has been overridden to use Block Mode ECM.	You cannot use both Block mode and Record mode in the same FileType definition. Make the definition use one mode or the other.

Number	Message	Description
SXF0219E	File cannot be defined as BINARY yet have useFieldWidth=%s	Setting a file type to BINARY is mutually exclusive with useFieldWidth=No. Change one or the other parameter.
SXF0220E	Input directory library [%s] cannot be the same as the process directory library.	The input directory library and the process directory library cannot be the same. Specify unique input and process directories libraries.
SXF0221E	Process directory library [%s] cannot be the same as the output directory library.	The output directory library and the process directory library cannot be the same. Specify unique output and process directories libraries.
SXF0222E	Invalid input library name: <%s>	The specified input library name was invalid. Use a valid library name. Check for invalid characters. Name length is 10 characters maximum.
SXF0223E	Invalid output library name: <%s>	The specified library name was invalid. Use a valid library name. Check for invalid characters. Name length is 10 characters maximum.
SXF0224E	Invalid process library name: <%s>	The library name specified was invalid. Use a valid library name. Check for invalid characters. Name length is 10 characters maximum.
SXF0225E	Pre-Register Sections appears before Options section near line %d	The Pre-Register section was specified in the configuration file in an invalid location. Specify the Pre-Register section after the Options section.
SXF0226E	Duplicate definition for [%s] section	The section specified appears more than once in the configuration file. Correct the configuration file.
SXF0227E	Config parse error near line %d, invalid section name: '%s'	A section name in the configuration file was incorrect. A section name must be one of Trace, Options, Pre-Register, or FileType. Check the spelling of the section names of the configuration file. Names are case sensitive.





Number	Message	Description
SXF0236E	Missing [%s] section option: %s	The configuration file was missing the section option specified. You must specify the indicated option. Add the option to the configuration file.
SXF0237E	Missing [%s] section option: %s	The configuration file was missing the section option specified. You must specify the indicated option. Add the option to the configuration file.
SXF0238E	Missing [%s] section option: %s	The configuration file was missing the section option specified. You must specify the indicated option. Add the option to the configuration file.
SXF0239E	Missing %s definition for file prefix <%s>	The specified section for this file type was missing. Add the definitions to the configuration file.
SXF0240E	*** CSI error: moduleid=0x%X%X reason=%d retcode=%d	The FileAdapter received an un-expected error from IBM's Catalog facility. Record this error information, and report it to TIBCO technical support.
SXF0241E	FileType cannot be published. Specify Polling or Trigger.	A publishing method was not specified for this file type. Update the configuration file to provide a publishing method for this FileType.
SXF0242E	Only %d message fields options are supported	The maximum number of field options had been exceeded. Correct the configuration file.
SXF0243E	Failed to open TIBCO Rendezvous -- %s	Unable to start TIBCO Rendezvous TIBCO Rendezvous might not be active. Check with your TIBCO administrator that TIBCO Rendezvous is available.
SXF0244E	Failed to create RV transport session: %s	Unable to create a network transport. TIBCO Rendezvous might not be active. Check with your TIBCO administrator that TIBCO Rendezvous is available.

Number	Message	Description
SXF0245E	Failed to open TIB/Rendezvous RVCN -- %s	Unable to start TIBCO Rendezvous. TIBCO Rendezvous might not be active. Check with your TIBCO administrator that TIBCO Rendezvous is available.
SXF0246E	Failed to initialize RV Transport session: %s	Unable to create a network transport session. TIBCO Rendezvous might not be active. Check with your TIBCO administrator that TIBCO Rendezvous is available.
SXF0247E	Failed to initialize RVCN Transport session: %s	Unable to create a transport for certified delivery. TIBCO Rendezvous might not be active. Check with your TIBCO administrator that TIBCO Rendezvous is available.
SXF0248E	Configuration File Error near line <%d> missing ',' or ';' or Contents=%s	An error occurred in the configuration file. Verify that the configuration file is coded properly.
SXF0249E	Found definitions for both Reliable and Certified mode	The configuration file contained definitions for both modes. The modes are exclusive. Use only one type of mode.
SXF0250E	Found definitions for both Reliable and Certified mode	The configuration file contained definitions for both modes. The modes are exclusive. Use only one type of mode.
SXF0251E	Missing tag <%s> for %s	The configuration file was missing a required parameter. The error message indicates which parameter is missing. Check the configuration file, and correct the discrepancy.
SXF0252E	Missing tag <%s> for %s	The configuration file was missing a required parameter. The error message indicates which parameter is missing. Check the configuration file, and correct the discrepancy.
SXF0253E	Unable to allocate memory for MaxECMSubs of length=%d	Unable to allocate memory. Report error to your TIBCO administrator.
SXF0254E	A VSAM key cannot be longer than 255 bytes. Length=%d	VSAM limits keys to 255 bytes. Go re-check the length of your key, because it exceeds VSAM's maximum length.

Number	Message	Description
SXF0255E	GETMAIN failed when adding AltIndex name to Config List. Length=%d	An out of memory error was reported by MVS. Raise the Job Limits (REGION size) to provide more memory for the FileAdapter to run.
SXF0256E	Invalid file type: no prefix or extension	The File Type definition requires a prefix= entry to be supplied.
SXF0257E	File <%s> is not a VSAM file or else is not Cataloged	The File Adapter cannot find a valid entry for this file in the MVS catalog. It is either mis-spelled or has been deleted from the MVS catalog.
SXF0258E	VSAM file type: missing Key Length	Each VSAM FileType definition requires a KeyLength parameter.
SXF0259E	VSAM file type: missing Key Offset	Each VSAM FileType definition requires a KeyLength parameter.
SXF0260E	SFT and Record Mode ECM Transfers require that you specify an endPublishSubject	SFT and Record Mode ECM operation requires that a endPublishSubject= parameter be supplied in the FileType definition.
SXF0261E	Do not use both <position> and <fieldStart> tags	You might have used both a position and a fieldStart tag in a constraint parameter. Use only one of the tags to specify a field location.
SXF0262E	Do not use both <position> and <fieldStart> tags	You might have used both a position and a fieldStart tag in a constraint parameter. Use only one of the tags to specify a field location.
SXF0263E	Expecting <%s> or <%s> tag	The configuration file was missing a required position parameter. The error message indicates which parameter is missing. Check the configuration file and correct the discrepancy.
SXF0264E	Expecting <%s> tag	The configuration file was missing a required length parameter. The error message indicates which parameter is missing. Check the configuration file and correct the discrepancy.

Number	Message	Description
SXF0265E	Expecting <%s> tag	The configuration file was missing a required location parameter. The error message indicates which parameter is missing. Check the configuration file and correct the discrepancy.
SXF0266E	Invalid data type <%s> between line <%d> and line <%d>	The "type" value was invalid. Correct the "type" value entered.
SXF0267E	Config parse error between line %d and line %d, invalid precision: <%s>	The precision value was invalid. Ensure that the precision specified in the configuration file is entered correctly. The precision values must be numeric positive values, separated by a comma.
SXF0268E	Config parse error between line %d and line %d, invalid precision: <%s>	The precision value was invalid. Ensure that the precision specified in the configuration file is entered correctly. The precision values must be numeric positive values, separated by a comma.
SXF0269E	Config parse error between line %d and line, bad message token: '%s'	In the configuration file for FileSubscriber, a token was invalid in the FILE_LINE element of the FileType section. The invalid token is displayed. Correct or delete the invalid token.
SXF0270E	Binary fields require a PRECISION definition tag	The configuration file did not have a "precision" tag for this field. Edit the configuration and supply a precision value for all Packed and Zoned fields.
SXF0271E	Expecting <%s> or <%s> tag	The FileAdapter was expecting parameter or tag identified by %s in the configuration, and did not find it. Add that tag to the definition.
SXF0272E	Expecting <%s> tag	The FileAdapter was expecting parameter or tag identified by %s in the configuration, and did not find it. Add that tag to the definition.
SXF0273E	Expecting <%s> tag	The FileAdapter was expecting parameter or tag identified by %s in the configuration, and did not find it. Add that tag to the definition.

Number	Message	Description
SXF0274E	Expecting <%s> tag	The FileAdapter was expecting parameter or tag identified by %s in the configuration, and did not find it. Add that tag to the definition.
SXF0275E	Packed/Zoned/Binary/Float fields require a PRECISION definition tag	The configuration file did not have a "precision" tag for this field. Edit the configuration and supply a precision value for all numeric fields.
SXF0276E	Packed/Binary/Zoned/Float fields require Fixed field width processing	A FileType had a field type defined that requires a "binary" file specification. Specify isBinary="true" for this file type.
SXF0277E	Packed/Binary/Zoned/Float fields can only be defined with BINARY files.	A FileType had a field type defined that requires a "binary" file specification. Specify isBinary="true" for this file type.
SXF0278E	Unable to allocate memory for Msg Container of length=%d	Unable to allocate memory. Report error to your TIBCO administrator
SXF0279E	Config parse error between line %d and line %d, failed to parse value for %s	A supplied value was incomplete, typically the supplied value is missing an opening or closing quote ("), brace ({ }), or an equal sign (=). Make sure that the values in the configuration file have matching quotes and braces, and that there are equal signs where required.
SXF0280E	Config parse error between line %d and line %d, missing container name	A message container did not have a name. Check the configuration file, and make sure that all message containers have a name=" " entry. Values are case sensitive.
SXF0281E	Config parse error between line %d and line %d, bad message token: '%s'	In the configuration file for FileSubscriber, a token was invalid in the FILE_LINE element of the FileType. The invalid token is displayed. Correct or delete the invalid token.
SXF0282E	Option Attributes: Memory re-allocation failed	Unable to allocate memory. Report error to your TIBCO administrator.
SXF0283E	RV Session Attributes: Memory allocation failed, length=%d	Unable to allocate memory. Report error to your TIBCO administrator.

Number	Message	Description
SXF0284E	RV Session Attributes: Config parse error near line %d, failed to parse value	A supplied value was incomplete, typically the supplied value is missing an opening or closing quote ("), brace ({}), or an equal sign (=). Make sure that the values in the configuration file have matching quotes and braces, and that there are equal signs where required.
SXF0285E	RVCM Session Attributes: Memory allocation failed, length=%d	Unable to allocate memory. Report error to your TIBCO administrator.
SXF0286E	RVCM Session Attributes: Config parse error near line %d, failed to parse value	A supplied value was incomplete, typically the supplied value is missing an opening or closing quote ("), brace ({}), or an equal sign (=). Make sure that the values in the configuration file have matching quotes and braces, and that there are equal signs where required.
SXF0287E	Config parse error near line %d, invalid option name: '%s'	One of the values in the Options section might be misspelled, or might not have the proper case. The invalid option is displayed. Check the spelling of the values in the Options section of the configuration file. Values must be in upper case
SXF0288E	Config parse error near line %d, failed to parse Options line	A supplied value was incomplete, typically the supplied value is missing an opening or closing quote ("), brace ({}), or an equal sign (=). Make sure that the values in the configuration file have matching quotes and braces, and that there are equal signs where required.
SXF0289E	FileType Attributes: Memory re-allocation failed	Unable to allocate memory. Report error to your TIBCO administrator
SXF0290E	FileType Attributes: Memory allocation failed, length=%d	Unable to allocate memory. Report error to your TIBCO administrator

Number	Message	Description
SXF0291E	FileType Attributes: Config parse error between line %d and line %d, failed to parse value	A supplied value was incomplete, typically the supplied value is missing an opening or closing quote ("), brace ({}), or an equal sign (=). Make sure that the values in the configuration file have matching quotes and braces, and that there are equal signs where required.
SXF0292E	MsgFields Attributes: Memory allocation failed, length=%d	Unable to allocate memory. Report error to your TIBCO administrator
SXF0293E	FileType Attributes: Config parse error between line %d and line %d, failed to parse value	A supplied value was incomplete, typically the supplied value is missing an opening or closing quote ("), brace ({}), or an equal sign (=). Make sure that the values in the configuration file have matching quotes and braces, and that there are equal signs where required.
SXF0294E	Config parse error between line %d and line %d, invalid option name: '%s'	One of the values in the Options section might be misspelled, or might not have the proper case. The invalid option is displayed. Check the spelling of the values in the Options section of the configuration file. Values must be in upper case.
SXF0295E	Config parse error between line %d and line %d, failed to parse File Type line	In the configuration file, the Options section must have matching opening and closing '}'. Check the configuration file and make sure that the Options section has matching opening and closing '}'.
SXF0296E	Config parse error near line %d, invalid Trace option name: '%s'	One of the values in the Options section might be misspelled, or might not have the proper case. The invalid option is displayed. Check the spelling of the values in the Options section of the configuration file. Values must be in upper case.



Number	Message	Description
SXF0297E	Config parse error near line %d, failed to parse Trace line	In the configuration file, the Options section must have matching opening and closing '}'. Check the configuration file and make sure that the Options section has matching opening and closing '}'.
SXF0298E	Failed to pre-register listener %s for subject %s cause=%s	An invalid subject name was given, or there is a problem with TIBCO Rendezvous. Specify a valid subject name. If the problem continues, contact your TIBCO administrator.
SXF0299E	Expecting <%s> tag	The configuration file was missing a required parameter. The error message indicates which parameter is missing. Check the configuration file and correct the discrepancy.
SXF0300E	Expecting <%s> tag	The configuration file was missing a required parameter. The error message indicates which parameter is missing. Check the configuration file and correct the discrepancy.
SXF0301E	PreRegister Attributes: Memory re-allocation failed	Unable to allocate memory. Report error to your TIBCO administrator.
SXF0302E	Only %d Subscribers can be pre-registered	You had specified too many pre-registered subscribers. Reduce the number of pre-registered listeners. The maximum is 126.
SXF0303E	Only %d Subscribers can be pre-registered	You had specified too many pre-registered subscribers. Reduce the number of pre-registered listeners. The maximum is 126.
SXF0304E	Subscriber Attributes: Memory allocation failed, length=%d	Unable to allocate memory. Report error to your TIBCO administrator.
SXF0305E	Subscriber Attributes: Config parse error near line %d, failed to parse value	A supplied value was incomplete, typically the supplied value is missing an opening or closing quote ("), brace ({ }), or an equal sign (=). Make sure that the values in the configuration file have matching quotes and braces, and that there are equal signs where required.

Number	Message	Description
SXF0306E	Config parse error near line %d, invalid Subscriber option name: '%s'	One of the values in the Options section might be misspelled, or might not have the proper case. The invalid option is displayed. Check the spelling of the values in the Options section of the configuration file. Values must be in upper case.
SXF0307E	Config parse error near line %d, failed to parse options line	A supplied value was incomplete, typically the supplied value is missing an opening or closing quote ("), brace ({}), or an equal sign (=). Make sure that the values in the configuration file have matching quotes and braces, and that there are equal signs where required.
SXF0308E	Only %d file types are supported	A maximum of 1024 file types is supported, for each File Adapter.
SXF0309E	Missing %s definition for file prefix <%s>	The specified section for this file type was missing. Add the definitions to the configuration file.
SXF0310E	Missing %s definition for file prefix <%s>	The specified section for this file type was missing. Add the definitions to the configuration file.
SXF0311E	Missing %s definition for file prefix <%s>	The specified section for this file type was missing. Add the definitions to the configuration file.
SXF0312E	Process Dataset not found: %s	The data set indicated was not found. Verify that the data set exists and is available to FilePublisher.
SXF0313E	Output Dataset not found: %s	The data set indicated was not found. Verify that the data set exists and is available to FilePublisher.
SXF0314E	Missing %s definition for file prefix <%s>	The specified section for this file type was missing. Add the definitions to the configuration file.

Number	Message	Description
SXF0315E	Missing %s definition for file prefix <%s>	The specified section for this file type was missing. Add the definitions to the configuration file.
SXF0316E	Missing %s definition for file prefix <%s>	The specified section for this file type was missing. Add the definitions to the configuration file.
SXF0317E	Missing %s definition for file prefix <%s>	The specified section for this file type was missing. Add the definitions to the configuration file.
SXF0318E	Process Library not found: %s	The library indicated was not found. Verify that the data set exists and is available to FilePublisher.
SXF0319E	Output Library not found: %s	The library indicated was not found. Verify that the data set exists and is available to FilePublisher.
SXF0320E	Missing %s definition for file prefix <%s>	The specified section for this file type was missing. Add the definitions to the configuration file.
SXF0321E	Missing %s definition for file prefix <%s>	The specified section for this file type was missing. Add the definitions to the configuration file.
SXF0322E	Missing %s definition for file prefix <%s>	The specified section for this file type was missing. Add the definitions to the configuration file.
SXF0323E	Process Directory not found: %s	The specified directory was not found. Make sure that the directory specified exists and is available.
SXF0324E	Output Directory not found: %s	The specified directory was not found. Make sure that the directory specified exists and is available.
SXF0325E	Missing %s definition	No Adapter name was specified for this configuration. Correct the configuration file.

Number	Message	Description
SXF0326E	Failed to open config file: %s	Unable to open the configuration file or the configuration file was not found. Make sure that the configuration file exists, and that Adapter can find it.
SXF0327E	Memory allocation for Max FileType failed, length=%d	Unable to allocate memory. Report error to your TIBCO administrator.
SXF0328E	MVS CSI problem. Incorrect file name or file not cataloged=<%s>	The File Adapter cannot find a valid entry for this file in the MVS catalog. It is either mis-spelled or has been deleted from the MVS catalog.
SXF0330W	endPublishSubject is not used by Block Mode ECM. Parameter ignored.	endPublishSubject was not a valid parameter to use when defining Block Mode ECM file types.
SXF0331W	Invalid value <%s> for DELETE_PUB_PRG_FILES. Parameter ignored.	The value specified for the DELETE_PUB_PRG_FILES was not correct. Go re-check the correct options in the manual.
SXF0333E	File <%s> not found. Requested by user=%s on Subject/Destination %s. File publish request Rejected.	The File Adapter cannot find a valid entry for this file. It is either mis-spelled or has been deleted.
SXF0336W	Both Code Pages must be configured. Single code page config is ignored.	Normally, both HOSTCODEPAGE and NETWORKCODEPAGE have to be specified when overriding the default code pages used by the File Adapter.
SXF0337W	Code Page Setup failed: reason=<%s>. Will use defaults instead.	Unable set up the requested code pages. Verify that the code page settings are valid for your system and that TIBCO Rendezvous is active. Check with your TIBCO administrator.
SXF0339E	Location value (%d) exceeds max allowed (%d), near line %d	The location= attribute tag exceeded the maximum field offset allowed in a record. Check the configuration file, and correct the discrepancy.

Number	Message	Description
SXF0340E	Total number of message item/container definitions (%d) exceeds max number of items allowed (%d), near line %d	The maximum number of message definitions for a given FileType had been exceeded. Correct the configuration file.
SXF0342E	Missing Block Mode ECM Subscriber defs. FileType definition ignored	Block Mode ECM file type definitions required a EcmSubscriber definition. The Block mode definition is incorrect, because it lacks any EcmSubscriber definitions.
SXF0343E	Missing Record Mode ECM confirmationSubject/endPublish Subject or TotalCount defs.	Record Mode ECM file type definitions required a confirmationSubject and a Total count. Supply those parameters.
SXF0344E	Missing useExplicitConfirmation defs when using %s. FileType definition ignored	A config mismatch had been detected. When using the parameter identified by %s, a useExplicitConfirmation must also be used.
SXF0345E	Config error near line %d ic=%d:\n contents=%s	A configuration error was detected in the INI file near the line specified. The %s contains the reason for the error.
SXF0346E	Config error near line %d. Reading till next section is encountered	A configuration error was detected in the INI file near the line specified. The File Adapter skips to the next section, because of the severe config errors.
SXF0361E	EMS Session Attributes: Memory allocation failed, length=%d	Unable to allocate memory. Report error to your TIBCO administrator.
SXF0362E	EMS Session Attributes: Config parse error near line %d, failed to parsevalue	A supplied value was incomplete, typically the supplied value is missing an opening or closing quote ("), brace ( { } ), or an equal sign (=). Make sure that the values in the configuration file have matching quotes and braces, and that there are equal signs where required
SXF0363E	Failed to create connection to the server. Reason :%s, exiting...	RV or EMS was unable to create the connection needed to communicate. The %s identifies the error it encountered.
SXF0364E	Failed to create EMS session. Reason :%s, exiting...	Unable to create EMS session. The %s identifies the error it encountered.

Number	Message	Description
SXF0371E	<%s> Maximum EMS Timer Limit has been exceeded. Max limit = %d. Terminating...	The limit of EMS timers (128) was reached.
SXF0373E	Not a valid value for EMS msg trace. Valid values are 'null' or 'body'.	Valid values for EMS message trace are the entire message (body) or only the header of the message (null).
SXF0374E	Invalid %s setting=%s. Must be RecordMode, RecordModeECM, BlockModeSFT, or BlockModeECM	Valid values for the transferType parameter are RecordMode, RecordModeECM, BlockModeSFT, or BlockModeECM.
SXF0381E	EMS protocol is not allowed with ECM.	The EMS protocol is not allowed with ECM.
SXF0383E	Keyword %s is not allowed with %s protocol.	The specified keyword is not allowed with the specified protocol.
SXF0385E	EMS acknowledgement %s not allowed; changed to %s.	The specified EMS acknowledgement is not allowed. It has been changed to the value reported.
SXF0387E	EMS acknowledgement %s is invalid.	The specified EMS acknowledgement is not valid.
SXF0389E	TOPIC required for EMS durable subscriber %s.	The EMS durable subscriber required a destination type of TOPIC.
SXF0390W	startNewMessage set to default value: <true>	For EMS, a startNewMessage value of <false> is not allowed. It has been set to the only allowed value <true>.

## SXF1000 – SXF1999

Number	Message	Description
SXF1011E	Failed to unsubscribe for trigger Subject/Destination <%s>: reason=%s	Adapter was being shut down, and it encountered an error when trying to disconnect. The %s identifies the error it encountered.

Number	Message	Description
SXF1013E	Failed to unsubscribe for trigger Subject/Destination <%s>: reason=%s	Adapter was being shut down, and it encountered an error when trying to disconnect. The %s identifies the error it encountered.
SXF1031E	Invalid Publish Request for file <%s>: startKey > endKey. Request rejected.	A Trigger request was received that contained an invalid startKey /endKey combination. Fix the mis-match and re-submit the trigger request.
SXF1045E	Failed to append %s to Job Queue file - storing in memory only	Unable to append an incoming Trigger request to the on-disk job queue file. Check the configuration file to ensure a valid job queue filename was set up, and check any associated MVS error messages to see if a B37 (out of space) or security access error occurred when trying to update the file.
SXF1046E	Subject/Destination name %s does not match file type %d	A trigger was received for a file type that does not support the requested subject (or EMS) destination.
SXF1047E	Failed to update Job Queue file - storing in memory only	Unable to update a completed request in the on-disk Job Queue file. Check the configuration file to ensure a valid Job Queue filename was setup, and check any associated MVS error messages to see if a B37 (out of space) error occurred when trying to update the file.
SXF1048E	Failed to process trigger for Subscriber %s, using fieldnames=%s and %s error message: %s	A trigger message was received that did not have the minimum required parameters (e.g. a field name of "DATA" or "fileName", or a user-defined "Trigger" file name). FileAdapter is unable to process the message. Fix the application that is sending the trigger message to include the required field.

Number	Message	Description
SXF1049E	No ALTINDEX name matches Trigger filename=<%s> on Subscriber=<%s>	FilePublisher received a trigger message, but was unable to process it because the specified parameter was invalid. Check that the input value for the parameter is correct, check that TIBCO Rendezvous is available, and re-submit the trigger message.
SXF1050E	MAXRECORDS field %s on trigger message to %s is an invalid number. Rejected	FilePublisher received a trigger message, but was unable to process it because the specified parameter was invalid. Check that the input value for the parameter is correct, check that TIBCO Rendezvous is available, and re-submit the trigger message.
SXF1051E	STARTKEY field <%s> on trigger message to %s is an invalid RBA/RRN. Rejected	FilePublisher received a trigger message, but was unable to process it because the specified parameter was invalid. Check that the input value for the parameter is correct, check that TIBCO Rendezvous is available, and re-submit the trigger message.
SXF1052E	ENDKEY field <%s> on trigger message to %s is an invalid RBA/RRN. Rejected	FilePublisher received a trigger message, but was unable to process it because the specified parameter was invalid. Check that the input value for the parameter is correct, check that TIBCO Rendezvous is available, and re-submit the trigger message.
SXF1053E	Security Check failed. UserId=<%s> is not allowed READ access to File=<%s> from Subject/Destination=<%s>. rc=%d subcode=0x%x VOLSER=<%s> %s. Trigger request rejected, File not published.	Security checking was requested, and the security facility indicated the incoming Trigger request from the specified user, was not allowed access to publish that file. Contact your security people, to determine why the user is not authorized to publish that file.



Number	Message	Description
SXF1055E	Trigger request for file %s rejected because no UserId received on subject %s.	FilePublisher received a trigger message, but was unable to process it because Security checking is turned on, and no user id was included on the trigger message. FileAdapter is unable to process the message. Fix the application that is sending the trigger message to include the required field.
SXF1056E	The RACF call Security check failed for userId=<%s> File=<%s> from Subject/Destination=<%s>. Return code=<%d>. Volser=<%s>	Security checking was requested, and the security facility indicated the incoming Trigger request from the specified user, was not allowed access to publish that file. Contact your security people, to determine why the user is not authorized to publish that file.
SXF1508W	RV DAEMON disconnected, exiting app...	The RV Daemon being used by the FileAdapter disconnected, either because of a network crash or a crash of the daemon. The FileAdapter is unable to publish any more messages until the daemon (or its connection) is restored.
SXF1059E	<%s> CONFIG out of sync. invalid job queue index: index=%d max=%d addr=%x items=%d fname=<%s> subjectName=<%s> buff=<%s>	Recovered job queue had a bad entry.
SXF1509W	RV DAEMON disconnected, exiting app...	The RV Daemon being used by the FileAdapter disconnected, either because of a network crash or a crash of the daemon. The FileAdapter is unable to publish any more messages until the daemon (or its connection) is restored.
SXF1511W	%s exited with errors	The File Adapter terminated because of the error specified by the %s reason code.
SXF1521E	Error %d on Heartbeat timer create: %s	The File Adapter was unable to create a Heartbeat timer. The %d and %s indicate what the error was that caused the problem.

Number	Message	Description
SXF1524E	Error %d on Started Task console check timer create: %s	The File Adapter was unable to create a Console timer. The %d and %s indicate what the error was that caused the problem.
SXF1526E	Failed to create EMS/RV Heartbeat message: %s	Unable to create a heartbeat message. Make sure that TIBCO EMS/Rendezvous is available.
SXF1527E	Failed to add [%s] to Heartbeat message: %s	Unable to add the specified string to a heartbeat message. Make sure that TIBCO EMS/Rendezvous is available.
SXF1528E	Failed to add [%s] to Heartbeat message: %s	Unable to add the specified string to a heartbeat message. Make sure that TIBCO EMS/Rendezvous is available.
SXF1529E	Failed to add file prefix for: %d:%s	Unable to add the file prefix field to a heartbeat message. Make sure that TIBCO EMS/Rendezvous is available.
SXF1530E	Failed to add file extension for %d:%s	Unable to add the file extension to a heartbeat message. Make sure that TIBCO EMS/Rendezvous is available.
SXF1531E	Failed to add status for: %d:%s	Unable to add the status string to the heartbeat timer message. Check with the TIBCO administrator.
SXF1532E	Failed to add numPublished	Unable to add the numPublished string to the heartbeat timer message. Check with the TIBCO administrator.
SXF1533E	Failed to add progressLineNo	Unable to add the progressLineNo string to the heartbeat timer message. Check with the TIBCO administrator.
SXF1534E	Failed to add progressLineNo	Unable to add the progressLineNo string to the heartbeat timer message. Check with the TIBCO administrator.

Number	Message	Description
SXF1535E	Failed to set send Heartbeat target: cause=%s Subject/Destination=%s	An error was returned when Adapter attempted to set the subject name in a heartbeat message to be published. Ensure that a valid subject name is specified in the configuration file.
SXF1536E	Error in sending EMS/RV Heartbeat message: cause=%s Subject=%s	Unable to publish a heartbeat message. Make sure that TIBCO EMS/Rendezvous is available.
SXF1537E	Failed to destroy message.	Unable to destroy a published heartbeat message. Make sure that TIBCO EMS/Rendezvous is available.
SXF1538E	Config file not found in PARM= and no INIFILE DD. Configuration file from -config parameter and QINLINE file not found. Will use default config filename <%s> instead.	No configuration file was passed in as a parameter and no inline data file was found. This is probably an error by the operator. Double-check that you really want to use the default configuration file.
SXF1539E	Failed to get hostname from system. Error code returned by 'gethostname' is = %d	Unable to extract the host name being used by TCPIP. Probably indicates that TCP/IP is not up or not functioning.
SXF1540E	Could not find a usable TCP WinSock.DLL service.	A missing TCP Steplib entry is causing problems.
SXF1541E	Failed to get hostname from system. Error code returned by 'gethostname' is = %d	The TCP Comm Server was unable to return the hostname of the current system. Contact the system programmer to try to determine what the problem.
SXF1542E	Error in sending EMS/RV Heartbeat message: cause=%s subject=%s	Unable to publish a heartbeat message. Make sure that TIBCO EMS/Rendezvous is available.
SXF1543E	FilePoller List Memory allocation failed	Unable to allocate memory. Report error to your TIBCO administrator.
SXF1544E	Failed to subscribe to Subject/Destination <%s>: %s	Unable to create a listener. TIBCO EMS/Rendezvous might be unavailable or might not be active. Check with your TIBCO administrator.

Number	Message	Description
SXF1545E	Failed to subscribe to Subject/Destination <%s>: %s	Unable to create a listener. TIBCO EMS/Rendezvous might be unavailable or might not be active. Check with your TIBCO administrator.
SXF1546E	FilePoller Memory allocation failed for length=%d	Unable to allocate memory. Report error to your TIBCO administrator.
SXF1547E	No file type objects were created from configuration	Unable to create a file type object. At least one file type entry have to be completely defined in the configuration file.
SXF1548E	JobQueue element Memory allocation failed for length=%d	An out of memory error was reported.
SXF1549E	JobQueue element Memory allocation failed for length=%d	An out of memory error was reported.
SXF1550E	TriggerQue Memory allocation failed for length=%d	Unable to allocate memory. Report error to your TIBCO administrator.
SXF1551E	TriggerQue Memory allocation failed for length=%d	Unable to allocate memory. Report error to your TIBCO administrator.
SXF1552E	File Poller Timer removal failed: %s	The File Adapter encountered an error while terminating. The %s details the reason for the problem.
SXF1553E	Max_Dir FileList Memory allocation error for length=%d	Unable to allocate memory. Report error to your TIBCO administrator.
SXF1554E	Max_Dir FileList Memory allocation error for length=%d	Unable to allocate memory. Report error to your TIBCO administrator.
SXF1555E	File polling for %s data set is not supported!	The data set type was invalid for polling. The only data set that is supported for polling is SEQ. Use a different method to publish the file.
SXF1556E	Timer creation failed: %s	Unable to create a timer. This is a system-level error. Check that TIBCO EMS/Rendezvous is available and functioning properly. Contact the TIBCO administrator.

Number	Message	Description
SXF1557E	FilePoller's Memory allocation of FileSender struct failed for length=%d	Unable to allocate memory. Report error to your TIBCO administrator.
SXF1558E	NULL filePoller in timer callback	Adapter attempted to create a callback function, but the timer callback function is NULL. This is a system-level message. Contact your TIBCO administrator.
SXF1559E	Failed to subscribe to Subject/Destination <%s>: cause=%s	The File Adapter was unable to successfully subscribe to a given Subject. The %s details the reason for the problem.
SXF1560E	Failed to subscribe to Subject/Destination <%s>: cause=%s	The File Adapter was unable to successfully subscribe to a given Subject. The %s details the reason for the problem.
SXF1564E	Transaction Delay Timer removal failed: %s	Adapter was unable to remove the transaction delay timer.
SXF1565E	Admin Timer removal failed: %s"	Adapter was unable to remove the ECM Admin message timer
SXF1561W	setenv for EDC_ZERO_RECLLEN Failed: %d %s. VB TEXT files exposed	A OMVS setenv() was issued to allow proper operation of VB files, and it failed because of the reason specified in %s. Review this issue with the MVS systems programmer. It probably indicates an OMVS configuration or authorization problem.
SXF1562E	Error creating EMS Destination/Producer for %s dest=<%s>: cause=%s	An error occurred when creating an EMS Destination/Producer for EPM, or ERROR msgs, or EMS HEARTBEAT in Publisher.

## SXF2000 – SXF2999

Number	Message	Description
SXF2004E	<%s> Unable to locate Subscriber=%s	A confirmation reply was received, but no such subscriber was found in the current configuration. It indicates some sort of configuration mis-match between the Publisher and the Subscriber.
SXF2005E	<%s> No subscriber name found on Confirm msg.	A confirmation reply was received, but no such subscriber was found in the current configuration. It indicates some sort of configuration mis-match between the Publisher and the Subscriber.
SXF2010E	Close publisher file=<%s> failed.	A close on a file being published failed. Save the %s reason code and discuss the issue with your systems programmer.
SXF2023W	ExecuteAfterProcess ignored because the file <%s> has been removed	An ExecuteAfterProcess was requested in the configuration, but the target file was removed/deleted, and not available for use.
SXF2050W	<%s> Re-publishing EOF End Message to Subject/Destination=<%s> Block=%d idx=%d retry=%d #confBlks=%d isEOFblk=%d GUID=%s	The File Publisher did not receive a Block Mode or Record Mode ECM acknowledgement for the last block it sent. If this problem persists it might indicate a network problem. The Publisher retries to send the block again.
SXF2080E	Close publisher file: <%s> failed.	A close on a file being published failed. Save the %s reason code and discuss the issue with your systems programmer.
SXF2096W	<%s> Re-publishing to Subject/Destination=<%s> %s=%d idx=%d retry=%d #confBlks=%d GUID=%s Eofsm=%d empty=%d	The File Publisher did not receive a Block Mode or Record Mode ECM acknowledgement for the last block it sent. If this problem persists it might indicate a network problem. The Publisher retries to send the block again.

Number	Message	Description
SXF2103W	Move operation could not remove file <%s>	Source file was retained while moving a file. Source file was in use by another process when Adapter attempted to move the file. Remove the source file if it is not required.
SXF2104W	Directories Libraries are identical, file was not moved.	One of the input, process, or output libraries was not unique. Specify input, process and output libraries that are unique.
SXF2105W	Opening File <%s> in Binary Mode %s lineLength=%d	A file was being opened in Binary Mode with a linelength specified that might not match the file's actual length.
SXF2106W	<%s> Move operation could not remove source file. cause=%d: %s	A remove operation on a file that was published failed. Save the %s reason code and discuss the issue with your systems programmer.
SXF2107W	Directories Libraries are identical, file is not moved.	The indicates a configuration mis-match between the input or process files, and the output target files.
SXF2108W	Failed to remove file: <%s> because there were errors during processing.	Unable to remove the file indicated. The file still exists on the system. Verify that FilePublisher has the proper authority to remove files.
SXF2109W	<%s> Failed to remove 'AfterProcess' file=<%s>. Cause=%d %s	Unable to remove the file indicated. The file still exists on the system. Verify that FilePublisher has the proper authority to remove files.
SXF2110W	<%s> Failed to remove file=<%s>. Cause=%d %s	Unable to remove the file indicated. The file still exists on the system. Verify that FilePublisher has the proper authority to remove files.
SXF2111W	Failed to extract nested msg. field <%s> to form Subject/Destination name.	Unable to construct the subject from the field data. Check the configuration file for this message.
SXF2112W	Failed to extract field <%s> to form Subject/Destination name.	Unable to construct the subject from the field data. Check the configuration file for this message.

Number	Message	Description
SXF2113W	<%s> File pointer is NULL in publishBlock invoked from timer callback	The file pointer for a timer function was NULL. Report this to the TIBCO administrator. This portends a system error.
SXF2114W	File pointer is NULL in pubRecord callback. No file was open. pubRecord callback invoked. No file is open currently.	The file pointer for a timer function was NULL. Report this to the TIBCO administrator.
SXF2115W	Failed to remove file: <%s> because there were errors during processing.	Unable to remove the file indicated. The file still exists on the system. Verify that FilePublisher has the proper authority to remove files.
SXF2116W	Failed to remove file: <%s> cause=%s.	Unable to remove the file indicated. The file still exists on the system. Verify that FilePublisher has the proper authority to remove files.
SXF2117W	Failed to remove file: <%s> cause=%s.	The file still existed on the system. Verify that FilePublisher has the proper authority to remove files.
SXF2118W	<%s>: line %d has no fields	Configuration error. No fields were found in the file specified. Correct the configuration file.
SXF2119W	The specified constraint could not be found in the input file <%s>	Configuration error. The specified constraint could not be found in the input file. Correct the configuration file.
SXF2120E	Move failed: could not open file <%s>	The move operation failed. Check that the target directory is available and that there are no authority problems.
SXF2121E	Move failed: could not open output file <%s>	The move operation failed. Check that the target directory is available and that there are no authority problems.
SXF2122E	<%s> Move failed: could not open source file using mode=%s. cause=%d: %s	The move operation failed. Check that the target directory is available and that there are no authority problems.



Number	Message	Description
SXF2123E	<%s> Move failed: could not open target file using mode=%s. cause=%d: %s	The move operation failed. Check that the target directory is available and that there are no authority problems.
SXF2124E	<%s> Move failed: Unable to close source file cause=%d: %s	The file specified could not be moved. Be sure that FilePublisher has the proper authority.
SXF2125E	<%s> Move failed: Write error on output file. cause=%d: %s	Unable to move a file. Ensure that FilePublisher has the authority to move the specified file.
SXF2126E	<%s> Move failed: Data loss while flushing the output file	Unable to complete moving a file. Ensure that FilePublisher has the authority to move the specified file.
SXF2127E	<%s> Move failed: Error renaming input file	The File Adapter was unable to successfully rename a CWK file to the target filename used by the FileType. The second %s details the reason for the problem.
SXF2128E	<%s> Move failed: Error moving Binary input file.	The File Adapter was unable to successfully rename a CWK file to the target filename used by the FileType. A second related error message details why the move failed.
SXF2129E	<%s> Move failed: Error moving Text input file.	The File Adapter was unable to successfully rename a CWK file to the target filename used by the FileType. A second related error message details why the move failed.
SXF2130E	<%s> Move failed: Error renaming input file cause=%d: %s	The File Adapter was unable to successfully rename a CWK file to the target filename used by the FileType. The second %s details the reason for the problem.
SXF2131E	Failed to extract message <%s> for adding container: cause=%s	You might mismatch between the received message and the configuration file.
SXF2132E	Failed to Add message <%s> to container: cause=%s	You might mismatch between the received message and the configuration file.

Number	Message	Description
SXF2133E	Failed to Update message <%s> to container: cause=%s	Unable to update one or more fields in a container message. Check the configuration file. The message might have exceeded the maximum buffer length, or TIBCO EMS/Rendezvous might be unavailable.
SXF2134E	Failed to Add message <%s> to container: cause=%s	An attempt to add a field to a RV message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF2135E	Failed to Add String to Data message: cause=%s	An attempt to add a field to a RV message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF2136E	Failed to get message description tag: %s	You might mismatch the configuration file. Check the configuration file and ensure that the configuration matches the received message.
SXF2137E	Failed to Add Opaque/Binary to Data message: cause=%s	An attempt to add a field to an RV message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF2138E	Packed decimal data is invalid for field <%s>. Zeros sent	The data in a PACKED field was not valid. Ensure that the file data being published is of the right format.
SXF2139E	Packed decimal data is invalid for field <%s>. Zeros sent	The data in a PACKED field was not valid. Ensure that the file data being published is of the right format.
SXF2140E	Failed to Add Packed to Data message: cause=%s	Unable to add the specified field to a data message. Check with the TIBCO administrator.

Number	Message	Description
SXF2141E	Zoned field <%s> contains invalid character <%c>. Zeros sent	A ZONED field contained invalid data. Check the file data to ensure that it is of the proper type.
SXF2142E	Failed to Add Zoned to Data message: cause=%s	Unable to add the specified field to a data message. Check with the TIBCO administrator.
SXF2143E	Failed to destroy timer: cause=%s	Unable to destroy a timer after publishing a message. Check with the TIBCO administrator.
SXF2144E	Deallocate/Free GDG failed: %s	A deallocate for a GDG file failed. Review the %s error message with your MVS Systems programmer.
SXF2145E	Failed to unsubscribe to confirmation Subject/Destination: cause=%s	During shutdown, FileSubscriber was unable to terminate the subscription to a Subject. The cause %s specifies what the problem was
SXF2146E	<%s> JCL [%s] execution failed - %s	The JCL specified for executeAfterProcess or executeBeforeProcess did not execute properly. Check the JCL specified, and ensure that it can execute properly.
SXF2147E	<%s> Command [%s] returned error rc=%d	The command identified did not complete successfully. Check the command specified for executeBeforeProcess or executeAfterProcess and ensure that the commands are valid.
SXF2148E	JCL [%s] for File <%s> on Subject/Destination <%s> was not executed and the file was NOT removed because the file transfer failed.	The JCL specified for executeAfterProcess or executeBeforeProcess did not execute because (1) the file transfer failed, >and (2) the file's executeAfterProcess parameter was set to suppress running the JCL when file transfer failed.
SXF2149E	JCL [%s] for File <%s> on Subject/Destination <%s> execution failed - %s.	The JCL specified for executeAfterProcess or executeBeforeProcess did not execute properly. Check the specified JCL, and ensure that it can execute properly.

Number	Message	Description
SXF2150E	<%s> Command [%s] returned error rc=%d	The command identified did not complete successfully. Check the command specified for executeBeforeProcess or executeAfterProcess and make sure that the commands are valid.
SXF2151E	Error from the svc99 De-Allocate/FREE GDG function <%s>	A deallocate for a GDG file failed. Review the %s error message with your MVS Systems programmer.
SXF2152E	<%s> Remote Confirmation message indicates Subscriber I/O error. Terminating Block transfer to Sub=%s. Remaining Active Subs=%d.	A Block Mode ECM subscriber indicated that it received an out-space error, or some other kind of I/O error when writing to the work (CWK) file. The File Adapter drops that subscriber and continue to publish to any other active subscriber.
SXF2153E	<%s> Remote Confirmation message indicates Subscriber I/O error. Terminating Block transfer to Sub=%s.	A Block Mode ECM subscriber indicated that it received an out-space error, or some other kind of I/O error when writing to the work (CWK) file. There are not other active subscribers for that File, so the transfer is being terminated
SXF2154E	<%s> Failed to retrieve GUID from Confirmation message	An attempt to retrieve a field to an RV message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF2155E	<%s> Failed to retrieve Block Number from Confirmation message	An attempt to add a field to an RV message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF2156E	<%s> processing file <%s> set Subscriber %s offline. Its Max FlexECM retry limit was reached.	A Block Mode ECM subscriber exceeded its maximum error count thresh-hold, and was dropped from the list of active subscribers.

Number	Message	Description
SXF2157E	<%s> set Offline while sending file <%s>. All active ECM Subscribers failed, and Max FlexECM retry limit reached.	All the subscribers to a Block Mode ECM transfer had timed out. This normally indicates some sort of network problem.
SXF2158E	Error from the svc99 function <%s>, file=<%s>	A dynamic Open for a file failed. Verify that the file name is correct. If it is, review the %s error message with your MVS systems programmer.
SXF2159E	Could not open output GDG File=<%s>	A dynamic Open for a file failed. Verify that the file name is correct. If it is, review the %s error message with your MVS systems programmer.
SXF2160E	<%s> Failed to close .PRG file=%s, aborting recovery	Unable to close the progress file. Check the process directory library. The file might be corrupted.
SXF2161E	<%s> Failed to remove .PRG file=%s, aborting recovery	Unable to remove the progress file. Check the process directory library. The file might be corrupted or not properly cataloged.
SXF2162E	<%s> Failed to read data in progress file=%s. Wrong number of fields (%d), aborting recovery	Unable to read the progress file. Check the process directory library. The file might be corrupted.
SXF2163E	<%s> Failed to close .PRG file=%s, aborting recovery	Unable to close the progress file. Check the process directory library. The file might be corrupted.
SXF2164E	<%s> Failed to remove .PRG file=%s, aborting recovery	Unable to remove the progress file. Check the process directory library. The file might be corrupted or not properly cataloged.
SXF2165E	<%s> Failed to read data in progress file=%s. Read of file failed, aborting recovery	Unable to read the progress file. Check the process directory library. The file might be corrupted.
SXF2166E	<%s> Failed to close .PRG file=%s, aborting recovery	Unable to close the progress file. Check the process directory library. The file might be corrupted.

Number	Message	Description
SXF2167E	<%s> Failed to remove .PRG file=%s, aborting recovery	Unable to remove the progress file. Check the process directory library. The file might be corrupted or not properly cataloged.
SXF2168E	<%s> Failed to extract nested RV message for NT_ECM TEXT for ECM publish block. Reason: %s	FilePublisher was unable to extract the nested TEXT data field from a ECM block data message. Check with the TIBCO administrator.
SXF2169E	<%s> Failed to create RV message of length=%d for ECM publish block. Reason: %s	Unable to create a TIBCO Rendezvous message. Contact the TIBCO administrator to ensure that TIBCO Rendezvous is functioning.
SXF2170E	<%s> Failed to create RV message of length=%d: Reason: %s	Unable to create a message. Contact the TIBCO administrator to ensure that TIBCO Rendezvous is functioning.
SXF2171E	<%s> Failed to add %s to message: %s	Unable to create a message. Contact the TIBCO administrator to be certain that TIBCO Rendezvous is functioning.
SXF2172E	<%s> Failed to create nested RV message of length=%d: Reason: %s	Unable to create a nested message. Contact the TIBCO administrator to ensure that TIBCO Rendezvous is functioning.
SXF2173E	<%s> Failed to update nested RV message of length=%d: Reason: %s	Unable to create a message. Contact the TIBCO administrator to ensure that TIBCO Rendezvous is functioning.
SXF2174E	<%s> Failed to add %s to EMS/RV message: %s	Unable to create a message. Contact the TIBCO administrator to ensure that TIBCO EMS/Rendezvous is functioning.
SXF2175E	<%s> Failed to add %s to EMS/RV message: %s	Unable to create a message. Contact the TIBCO administrator to ensure that TIBCO EMS/Rendezvous is functioning.
SXF2176E	<%s> Failed to add %s to EMS/RV message: %s	Unable to create a message. Contact the TIBCO administrator to ensure that TIBCO EMS/Rendezvous is functioning.

Number	Message	Description
SXF2177E	<%s> Failed to add %s to EMS/RV message: %s	Unable to create a message. Contact the TIBCO administrator to ensure that TIBCO EMS/Rendezvous is functioning.
SXF2178E	<%s> Failed to add %s to EMS/RV message: %s	Unable to create a message. Contact the TIBCO administrator to ensure that TIBCO EMS/Rendezvous is functioning.
SXF2179E	<%s> Failed to add %s to EMS/RV message: %s	Unable to create a message. Contact the TIBCO administrator to ensure that TIBCO EMS/Rendezvous is functioning.
SXF2180E	<%s> Failed to add %s to EMS/RV message: %s	Unable to create a message. Contact the TIBCO administrator to ensure that TIBCO EMS/Rendezvous is functioning.
SXF2181E	<%s> Failed to create EndMsg RV message of length=%d: %s	Unable to create a message. Contact the TIBCO administrator to ensure that TIBCO Rendezvous is up and running.
SXF2182E	<%s> Failed to add %s to EMS/RV End message: %s	An attempt to add a field to a RV or EMS message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF2183E	<%s> Failed to add Filename to EMS/RV End message: %s	An attempt to add a field to a RV or EMS message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF2184E	<%s> Failed to add NumberOfMessage to EMS/RV End message: %s	An attempt to add a field to a RV or EMS message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.

Number	Message	Description
SXF2185E	<%s> Failed to add %s to RV End message: %s	An attempt to add a field to a RV or EMS message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF2186E	<%s> Failed to add %s to EMS/RV End message: %s	An attempt to add a field to a RV or EMS message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF2187E	<%s> Failed to add %s to End message: %s	An attempt to add a field to a RV or EMS message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF2188E	<%s> Failed to add %s to End message: %s	An attempt to add a field to a RV or EMS message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF2189E	<%s> Failed to add %s to End message: %s	An attempt to add a field to a RV or EMS message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF2190E	<%s> Failed to add %s to End message: %s	An attempt to add a field to a RV or EMS message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.



Number	Message	Description
SXF2191E	<%s> Failed to add %s to End message: %s	An attempt to add a field to a RV or EMS message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF2192E	<%s> Failed to add %s to End message: %s	An attempt to add a field to a RV or EMS message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF2193E	Failed to create Error EMS/RV message of length=%d: %s	An attempt to add a field to a RV or EMS message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF2194E	Failed to add [%s] to Error message: %s	An attempt to add a field to a RV or EMS message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF2195E	Failed to set Error send subject/Destination: %s cause=%s	An attempt to add a field to a RV or EMS message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF2196E	Failed to send error message: cause=<%s> Subject/Destination=%s	File Publisher was unable to send a message to the network. The %s cause details why. It might indicate a network or RV daemon problem.
SXF2197E	Failed to destroy Error EMS/RV message: reason=%s	File Publisher was unable to send a message to the network. The %s cause details why.

Number	Message	Description
SXF2198E	Cannot open null file name. rmode=%d	File name in configuration file was NULL (""). Check the configuration file, and supply a file name where it's missing.
SXF2199E	Could not open output GDG file=<%s>. GDG error=%s	A open for a GDG file failed. It might be because of an invalid GD name. If the problem persists, review the %s error message with your MVS Systems programmer.
SXF2200E	Failed to open output GDG file=<%s>	A open for a GDG file failed. It might be because of an invalid GD name. If the problem persists, review the %s error message with your MVS Systems programmer.
SXF2201E	<%s> File open failed for file <%s> using mode '%s'. errno=%d R15=%d Fdbk=%d op=%d ab_sys=0x%X ab_rc=0x%X\n errno text=%s	Unable to open the file identified in the message. Correct the configuration file entry for this file, and be certain that the file is available for use.
SXF2202E	Cannot open file <%s> because RLS function is inactive. SMSVSAM not up	Unable to open the file identified in the message. Correct the configuration file entry for this file, and ensure that the file is available for use.
SXF2203E	Cannot open file <%s> because trying to output to a Recoverable Sphere	Unable to open the file identified in the message. Correct the configuration file entry for this file, and ensure that the file is available for use.
SXF2204E	Cannot open file <%s> because RLS requires SMS files.File is not SMS managed	Unable to open the file identified in the message. Correct the configuration file entry for this file, and ensure that the file is available for use.
SXF2205E	Cannot open file <%s> because RLS detected DISP=SHR vs DISP=OLD conflict.	Unable to open the file identified in the message. Correct the configuration file entry for this file, and ensure that the file is available for use.
SXF2206E	<%s> Can not open file <%s> because OPEN flag still set on from previous run or another App is trying to share the file.	Unable to open the file identified in the message. Correct the configuration file entry for this file, and ensure that the file is available for use.

Number	Message	Description
SXF2207E	Deallocate/Free GDG failed: error %s	A deallocate for a GDG file failed. Review the %s error message with your MVS Systems programmer.
SXF2208E	<%s> VSAM key flocate failed for file <%s>. Terminating request. errno=%d R15=%d Fdbk=%d op=%d ab_sys=0x%X ab_rc=0x%X\n errno text=%s	Unable to locate the requested VSAM record by key, in the file identified in the message. Correct the trigger parameters sent for this file, and be certain that the file is available for use.
SXF2209E	<%s> Failed to open file '%s', errno=%d.	Unable to open the file identified in the message. Correct the configuration file entry for this file, and ensure that the file is available for use.
SXF2210E	<%s> Transfer aborted for file: <%s>. LRECL length %d > blockTransferSize %d	Unable to process the file identified in the message because the LRECL of the file was longer than the ECM block size specified for the transfer. Correct the configuration file entry for this file to make blockTransferSize equal to or larger than the file's LRECL.
SXF2211E	Failed to open .PRG file '%s' errno=%d errtext=%s	Unable to open the progress .PRG file identified in the message. Check to see that the .PRG file is available for use.
SXF2212E	<%s> TransactionDelay Timer creation failed: %s	Unable to create a timer. This is a system level error. Check that TIBCO EMS/Rendezvous is available and functioning properly. Contact the TIBCO administrator.
SXF2213E	<%s> ConfBlock - Failed to create message of length=%d: %s	Unable to create a message. Contact the TIBCO administrator to ensure that TIBCO EMS/Rendezvous is up and running.
SXF2214E	Failed to set EMS/RV send destination/ subject: %s	An error was returned when Adapter attempted to publish a message. Ensure that a valid destination/subject name is specified in the configuration file.

Number	Message	Description
SXF2215E	Failed to set EMS/RV reply destination/ subject: %s	An attempt to add a field to a RV or EMS message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF2216E	Error in sending RVCM message: cause=%s Subject/Destination=%s	An error occurred when sending an RVCM message. Adapter might not be active. Contact your TIBCO administrator and report this error.
SXF2217E	<%s> Error in sending RV message: cause=%s Subject=%s. OR <%s> Error in sending EMS message: cause=%s Destination=%s	An error occurred when sending a RV or EMS message. Adapter might not be active. Contact your TIBCO administrator and report this error.
SXF2218E	<%s> Failed to create EMS/RV message of length=%d: %s	Unable to create a message. Contact the TIBCO administrator to ensure that TIBCO EMS/Rendezvous is functioning.
SXF2219E	<%s> Failed to initialize EMS/RV message of length=%d: %s	An attempt to add a field to a RV or EMS message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF2220E	<%s> Failed to add %s to message: %s	An attempt to add a field to a RV or EMS message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF2221E	<%s> Failed to add %s to message: %s	An attempt to add a field to a RV or EMS message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.

Number	Message	Description
SXF2222E	<%s> Failed to destroy rvMsg: %s OR <%s> Failed to destroy emsMsg: %s	An attempt to add a field to a RV or EMS message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF2223E	<%s> Failed to remove timer: %s	Unable to terminate an RV timer. Contact the TIBCO administrator to be certain that TIBCO EMS/Rendezvous is functioning.
SXF2224E	<%s> Failed to create message of length=%d: %s	Unable to create a message. Contact the TIBCO administrator to be certain that TIBCO EMS/Rendezvous is functioning.
SXF2225E	<%s> Failed to create message of length=%d: %s	Unable to create a message. Contact the TIBCO administrator to be certain that TIBCO EMS/Rendezvous is functioning.
SXF2226E	<%s> Cannot append to message, failed to find start	Unable to append a field onto an existing message. Contact the TIBCO administrator to be certain that TIBCO EMS/Rendezvous is functioning.
SXF2227E	<%s> Failed to create message of length=%d: %s	Unable to create a message. Contact the TIBCO administrator to ensure that TIBCO EMS/Rendezvous is functioning.
SXF2228E	<%s> Failed to append container: %s cause=%s	Unable to append a field to an existing message. Contact the TIBCO administrator to be certain that TIBCO EMS/Rendezvous is functioning.
SXF2229E	<%s> Failed to create message of length=%d: %s	Unable to create a message. Contact the TIBCO administrator to ensure that TIBCO EMS/Rendezvous is functioning.
SXF2230E	<%s> Cannot append RBA/RRN I32 to message: %s	Unable to append a field to an existing message. Contact the TIBCO administrator to ensure that TIBCO EMS/Rendezvous is functioning.

Number	Message	Description
SXF2231E	Failed to destroy message: %s	Unable to free memory. Contact the TIBCO administrator to ensure that TIBCO EMS/Rendezvous is functioning.
SXF2232E	<%s> Failed to create Start message of length=%d: %s	Unable to create a message. Contact the TIBCO administrator to be certain that TIBCO EMS/Rendezvous is functioning.
SXF2233E	<%s> Failed to append FileName parm to Start message: %	Unable to add the filename field to the Start File Publish message. Check the configuration file to ensure that a valid file name was specified. Make sure that TIBCO EMS/Rendezvous is available.
SXF2234E	<%s> Failed to create Tracking message of length=%d: %s	An attempt to create a RV or EMS message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF2235E	<%s> Failed to create Trigger Reply message of length=%d: %s	An attempt to create a RV or EMS message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF2236E	<%s> Failed to append fields to Trigger Reply message: %s	Unable to add one or more fields to a Trigger Reply message. Make sure that TIBCO EMS/Rendezvous is available.
SXF2237E	<%s> Failed to create NT_ECM Listener: %s	Unable to create a RV Listener. Contact the TIBCO administrator to be certain that TIBCO EMS/Rendezvous is functioning.
SXF2238E	<%s> Failed to add String Data Field to message at col %d: %s	A configuration mismatch or an out of memory error occurred. Check the configuration file, and ensure that the configuration matches the message to be sent.

Number	Message	Description
SXF2239E	<%s> Failed to add String Data Field to message at col %d: %s	A configuration mismatch or an out of memory error occurred. Check the configuration file, and ensure that the configuration matches the message to be sent.
SXF2240E	<%s> Failed to add Bool Data Field to message at col %d: %s	A configuration mismatch or an out of memory error occurred. Check the configuration file, and ensure that the configuration matches the message to be sent.
SXF2241E	<%s> Failed to add Time Data Field to message at col %d: %s	A configuration mismatch or an out of memory error occurred. Check the configuration file, and ensure that the configuration matches the message to be sent.
SXF2242E	<%s> Failed to add I16 Data Field to message at col %d: %s	A configuration mismatch or an out of memory error occurred. Check the configuration file, and ensure that the configuration matches the message to be sent.
SXF2243E	<%s> Failed to add U16 Data Field to message at col %d: %s	A configuration mismatch or an out of memory error occurred. Check the configuration file, and ensure that the configuration matches the message to be sent.
SXF2244E	<%s> Failed to add I32 Data Field to message at col %d: %s	A configuration mismatch or an out of memory error occurred. Check the configuration file, and ensure that the configuration matches the message to be sent.
SXF2245E	<%s> Failed to add U32 Data Field to message at col %d: %s	A configuration mismatch or an out of memory error occurred. Check the configuration file, and ensure that the configuration matches the message to be sent.
SXF2246E	<%s> Failed to add F32 Data Field to message at col %d: %s	A configuration mismatch or an out of memory error occurred. Check the configuration file, and ensure that the configuration matches the message to be sent.
SXF2247E	<%s> Failed to add F64 Data Field to message at col %d: %s	A configuration mismatch or an out of memory error occurred. Check the configuration file, and ensure that the configuration matches the message to be sent.

Number	Message	Description
SXF2248E	<%s> Field %s: Invalid data type %s (%d)	An invalid message description was encountered. Check that the configuration file matches the message received, and that TIBCO EMS/Rendezvous is available.
SXF2249E	<%s> Invalid length defined for string field: %s	An invalid message description was encountered. Check that the configuration file matches the message received, and that TIBCO EMS/Rendezvous is available.
SXF2250E	<%s> Failed to add String Data Field to message at col %d: %s	A configuration mismatch or an out of memory error occurred. Check the configuration file, and be certain that the configuration matches the message to be sent.
SXF2251E	<%s> Failed to add String Data Field to message at col %d: %s	A configuration mismatch or an out of memory error occurred. Check the configuration file, and be certain that the configuration matches the message to be sent.
SXF2252E	<%s> Failed to add String Data Field to message at col %d: %s	A configuration mismatch or an out of memory error occurred. Check the configuration file, and be certain that the configuration matches the message to be sent.
SXF2253E	<%s> Failed to add Bool Data Field to message at col %d: %s	A configuration mismatch or an out of memory error occurred. Check the configuration file, and be certain that the configuration matches the message to be sent.
SXF2254E	<%s> Failed to add Time Data Field to message at col %d: %s	A configuration mismatch or an out of memory error occurred. Check the configuration file, and be certain that the configuration matches the message to be sent.
SXF2255E	<%s> Failed to add I16 Data Field to message at col %d: %s	A configuration mismatch or an out of memory error occurred. Check the configuration file, and be certain that the configuration matches the message to be sent.
SXF2256E	<%s> Failed to add U16 Data Field to message at col %d: %s	A configuration mismatch or an out of memory error occurred. Check the configuration file, and be certain that the configuration matches the message to be sent.



Number	Message	Description
SXF2257E	<%s> Failed to add I32 Data Field to message at col %d: %s	A configuration mismatch or an out of memory error occurred. Check the configuration file, and be certain that the configuration matches the message to be sent.
SXF2258E	<%s> Failed to add U32 Data Field to message at col %d: %s	A configuration mismatch or an out of memory error occurred. Check the configuration file, and be certain that the configuration matches the message to be sent.
SXF2259E	<%s> Failed to add F32 Data Field to message at col %d: %s	A configuration mismatch or an out of memory error occurred. Check the configuration file, and be certain that the configuration matches the message to be sent.
SXF2260E	<%s> Failed to add F64 Data Field to message at col %d: %s	A configuration mismatch or an out of memory error occurred. Check the configuration file, and be certain that the configuration matches the message to be sent.
SXF2261E	<%s> Field %s: Invalid data type %s (%d)	An invalid message description was encountered. Check that the configuration file matches the message received, and that TIBCO EMS/Rendezvous is available.
SXF2262E	<%s> Failed to create message of length=%d: %s	Could not create a message. Check that TIBCO EMS/Rendezvous is available.
SXF2263E	<%s> Failed to get container <%s>: %s	You might mismatch the configuration file. Check the configuration file, and be certain that the configuration matches the received message.
SXF2264E	<%s> Failed to update child message %s	Unable to update one or more fields in a message to be published. Check the configuration file, the message might have exceeded the maximum buffer length, or TIBCO EMS/Rendezvous might be unavailable.
SXF2265E	<%s> Failed to destroy child message %s	Unable to free memory. Contact the TIBCO administrator to be certain that TIBCO EMS/Rendezvous is functioning.

Number	Message	Description
SXF2266E	<%s> Move failed: Unable to close target file. cause=%d: %s	The file specified could not be moved. Be sure that FilePublisher has the proper authority.
SXF2267E	<%s> Move failed: Unable to close source file. cause=%d: %s	The file specified could not be moved. Be sure that FilePublisher has the proper authority.
SXF2268E	Move failed: copying source <%s> to target <%s>	Could not move the file specified to the directory indicated. Ensure that the file and directory exists, and that FilePublisher has the proper authority.
SXF2269E	<%s> Block Buffer: Memory allocation failed for length=%d	Unable to allocate memory. Report this error to your TIBCO administrator.
SXF2270E	<%s> ConfEndMsgAck Memory allocation failed for %d entries	Unable to allocate memory. Report this error to your TIBCO administrator.
SXF2271E	<%s> Confirm Listen EventId Memory allocation failed for %d entries	Unable to allocate memory. Report this error to your TIBCO administrator.
SXF2272E	<%s> ConfBlocks Memory allocation failed for length=%d	Unable to allocate memory. Report this error to your TIBCO administrator.
SXF2273E	<%s> ConfBlock.MsgAck Memory allocation failed for length=%d	Unable to allocate memory. Report this error to your TIBCO administrator.
SXF2274E	Failed to send error message: cause=<%s> Subject/Destination=%s	Unable to send a message. Contact the TIBCO administrator to be certain that TIBCO EMS/Rendezvous is functioning.
SXF2275E	<%s> Block Buffer: Memory allocation failed for length=%d	Unable to allocate memory. Report this error to your TIBCO administrator.
SXF2276E	Error in sending RVCN message: cause=%s Subject/Destination=%s	Unable to send a message. Contact the TIBCO administrator to be certain that TIBCO EMS/Rendezvous is functioning.
SXF2277E	<%s> Error in sending message: cause=%s Subject/Destination=%s	Unable to send a message. Contact the TIBCO administrator to be certain that TIBCO EMS/Rendezvous is functioning.

Number	Message	Description
SXF2278E	NULL fileSender in timer callback	Attempted to create a callback function, but the timer callback function is NULL. This is a system level error message, contact your TIBCO administrator.
SXF2279E	Failed to subscribe to Subject/Destination <%s> cause=%s	Unable to create a Subscriber for that destination. Contact the TIBCO administrator to be certain that TIBCO EMS/Rendezvous is functioning.
SXF2280E	<%s> Failed to subscribe to Subject/Destination <%s> cause=%s	Unable to create a Subscriber for that destination. Contact the TIBCO administrator to be certain that TIBCO EMS/Rendezvous is functioning.
SXF2281E	<%s> no confirmation Subject/Destination specified for Record Mode ECM	Unable to create a Subscriber for that destination. Contact the TIBCO administrator to be certain that TIBCO EMS/Rendezvous is functioning.
SXF2282E	Failed to Add Double to Data message: cause=%s	An attempt to add a field to a RV or EMS message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF2285E	<%s> Failed to add fileTransferDuration to Status message: %s	An attempt to add a field to a RV or EMS message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF2286E	There is an error in loading the configuration for this file type. Cannot publish file <%s>.	An error was discovered for the FileType associated with this subject, during configuration processing, and the file type was marked invalid and offline. Fix the error in the configuration file, and re-submit the publish request.

Number	Message	Description
SXF2294W	<%s> Re-publishing Queued Messages to Subject/Destination=<%s>: CurrBlk=%d CurrRec=%d RecovBlk=%d retry=%d #confBlks=%d Eofsm=%d ifactive=%d NtEcm=%d	The File Publisher did not receive a Block Mode or Record Mode ECM acknowledgement for the last block it sent. If this problem persists it might indicate a network problem. The Publisher retries to send the block again.
SXF2296E	Zoned field <%s> contains invalid character <%c>. Zeros sent	The data in the file to be published was invalid. Verify that the message definition is correct, and that the file is correctly formatted.
SXF2303E	Error from the Svc99 function <%s>. File=<%s> Volser=<%s ...> Unit=%s Type_alloc=%s Mode=%s	An error occurred when dynamically opening a file. The first %s denotes what the error was. Save this message and consult with your MVS programmer.
SXF2304E	Svc99 error: Duplicate Dataset Name=%s already exists on the system.	This normally indicates that two different file types are trying to use the same file. This is not allowed.
SXF2305E	Svc99 error: Insufficient space on requested Volumes.	A PROGRESS file is being created on a VOLUME that has run out of space.
SXF2306E	Svc99 error: RACF/ACF2 Security problem.	A PROGRESS file is being created and RACF has flagged an error (e.g. the Volume is password protected).
SXF2307E	Svc99 error: Cannot put a non-SMS dataset on an SMS volume.	A VOLSER had been specified for a Progress file that exists on a SMS managed volume
SXF2308E	Svc99 error: Catalog error - program not authorized to perform operation	A PROGRESS file is being created and RACF has flagged an error (e.g. the Volume is password protected).
SXF2309E	Svc99 error: RACF/ACF2 Security problem (Not Auth) or SMS error	A PROGRESS file is being created and RACF has flagged an error (e.g. the Volume is password protected).
SXF2310E	Svc99 error: File=<%s> already allocated and the tasks' usage attributes of 'a' and 'w' conflict	This normally indicates that two different file types are trying to use the same file. This is not allowed.

Number	Message	Description
SXF2311E	Output File <%s> could not be opened using DD=<%s> mode=<%s> Type_alloc=%s errno=%d %s	The specified file could not be opened. The errno %s specifies the specific cause of the problem.
SXF2312E	Append Output File <%s> might not be currently cataloged on VOLSER=<%s ...	The VOLSER parameter specified in the INI file might be incorrect for that file. Contact your MVS systems programmer.
SXF2313W	<%s> Message acknowledged. Re-Publishing successful on Subject/Destination=%s after %d retries	Indicates that blocks that were previously being re-published have now been acknowledged. Whatever network condition was causing the problem has been corrected.
SXF2318E	Svc99 error: Invalid file name %s was rejected by Allocate	The FILENAME parameter specified in the INI file were incorrect. Contact your MVS systems programmer.
SXF2319E	Svc99 error: Invalid VolSer list was rejected by Allocate	The VOLSER parameter specified in the INI file were incorrect. Contact your MVS systems programmer.
SXF2320E	Svc99 error: Invalid UNIT name %s was rejected by Allocate	The UNIT parameter specified in the INI file were incorrect. Contact your MVS systems programmer.
SXF2321E	Svc99 error: Invalid length was rejected by Allocate. Probably due to incorrect length on File name, Volser, or Unit	The FILENAME, UNIT, or VOLSER parameters specified in the INI file were incorrect. Contact your MVS systems programmer.
SXF2322E	Svc99 error: UNIT %s and Volser %s are incompatible or missing and were rejected by Allocate	The UNIT and VOLSER parameters specified in the Ibecause ofNI file were incompatible for the file being processed. Contact your MVS systems programmer.
SXF2323E	<%s> Failed to create nested EPM Payload message of length=%d: %s	An attempt to create a RV or EMS message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.

Number	Message	Description
SXF2326E	<%s> Failed to read data from input process file. Next Block # %d	An attempt to retrieve a field to a RV or EMS message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF2327E	Error adding field to EPM message: cause=%s Subject/Destination=%s	An attempt to add a field to a RV or EMS message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF2342E	Invalid null pointer for pFileSender. Terminating Publisher	An unexpected condition was detected. The Publisher is terminated.
SXF2343E	Invalid null pointer for pFileOptions. Terminating Publisher	An unexpected condition was detected. The Publisher is terminated.
SXF2344E	Invalid null pointer for %s. pFileSender=%08X timer_kick=%d	An unexpected condition was detected.
SXF2346E	<%s> Opening file=<%s> with mode=%s failed.	The file adapter was unable to open the specified file.
SXF5329E	Only one durable subscriber can be defined per FileOption section	Only one durable subscriber can be defined per FileOption section.
SXF5332E	Subscribe <%s> and Generate <%s> Destination names need to be equal for Guaranteed Delivery.	If guaranteed delivery of EMS messages was used and if generateFileDestinationName was specified, then subscribeDestinationName must equal generateFileDestinationName

**SXF3000 – SXF3999**

Number	Message	Description
SXF3000E	<%s> Unable to get reply Subject/Destination for Subscriber=%s	A Block Mode ECM internal or recovery type error occurred. Contact TIBCO technical support.
SXF3003E	<%s> Unable to locate Subscriber=%s. Not registered.	A Block Mode ECM internal or recovery type error occurred. Contact TIBCO technical support.
SXF3004E	<%s> No subscriber name found in msgType=%s msg. Unable to reply	A Block Mode ECM internal or recovery type error occurred. Contact TIBCO technical support.
SXF3005W	<%s> Ignoring Activate from Subscriber=%s until current file completed.	A recovery race occurred. When the current file is complete, the recovery sequence is completed.
SXF3008E	<%s> Received unknown ECM Admin Subscriber request %d	A Block Mode ECM internal or recovery type error occurred. Contact TIBCO technical support.
SXF3009W	<%s> No reply subject/Destination for Subscriber=%s on msgtype=%d.	A Block Mode ECM internal or recovery type error occurred. Contact TIBCO technical support.
SXF3012E	<%s> Unable to locate Subscriber=%s. Not registered.	A Block Mode ECM internal or recovery type error occurred. Contact TIBCO technical support.
SXF3013E	<%s> No subscriber name found on msgType=%d. Not registered.	A Block Mode ECM internal or recovery type error occurred. Contact TIBCO technical support.
SXF3015E	<%s> Received unknown ECM Admin Subscriber reply %d.	A Block Mode ECM internal or recovery type error occurred. Contact TIBCO technical support.

Number	Message	Description
SXF3021E	<%s> Admin Retry Timer create failed: %s	An attempt to create a RV or EMS timer failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF3022E	<%s> set Subscriber %s offline. It's Max FlexECM retry limit was reached during Activation.	A Block Mode ECM subscriber exceeded its maximum error count thresh-hold, and was dropped from the list of active subscribers.
SXF3023E	<%s> set Offline. Not all ECM Subscribers Active, Max FlexECM retry limit reached, and .PRG file <%s> failed.	A Block Mode ECM subscriber exceeded its maximum error count thresh-hold, and was dropped from the list of active subscribers.
SXF3024E	<%s> set Offline. No ECM Subscribers Active, and Max FlexECM retry limit reached.	All the subscribers to a Block Mode ECM transfer had timed out. This normally indicates some sort of network problem.
SXF3025E	<%s> %d subscribers set Offline, because Max FlexECM retry limit reached.	A group of subscribers to a Block Mode ECM transfer had timed out. This normally indicates some sort of network problem.
SXF3026E	<%s> Failed to create RV Admin message: %s	An attempt to create a RV or EMS message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF3027E	<%s> Failed to append MsgType to Admin message: %s	Unable to add the MsgType field to an ECM Admin message. Check that TIBCO EMS/Rendezvous is available.
SXF3028E	<%s> Failed to append MsgId/Guid to Admin message: %s	Unable to add the MsgId/Guid field to an ECM Admin message. Check that TIBCO EMS/Rendezvous is available.
SXF3029E	<%s> Failed to append Version to Admin message: %s	Unable to add the Version field to an ECM Admin message. Check that TIBCO EMS/Rendezvous is available.



Number	Message	Description
SXF3030E	<%s> Failed to append bool flag to Admin message: %s	Unable to add the bool flag field to an ECM Admin message. Check that TIBCO EMS/Rendezvous is available.
SXF3031E	<%s> Failed to append NumRetries to Admin message: %s	Unable to add the NumRetries field to an ECM Admin message. Check that TIBCO EMS/Rendezvous is available.
SXF3032E	<%s> Failed to set RV Admin send subject: %s	An error was returned when Adapter attempted to set the ECM Admin Subject name in a message to be published. Ensure that a valid subject name is specified in the configuration file.
SXF3033E	<%s> Failed to set RV Admin reply subject: %s	An error was returned when Adapter attempted to set the ECM Admin Reply Subject name in a message to be published. Ensure that a valid subject name is specified in the configuration file.
SXF3034E	<%s> Failed to send RV Admin msg: cause=%s subject=%s	Unable to publish an ECM Admin message. Check that TIBCO EMS/Rendezvous is available.
SXF3035E	<%s> Failed to subscribe to Admin Request subject <%s> cause=%s	An error was returned when Adapter attempted to subscribe to an ECM Admin Subject name. Ensure that a valid subject name is specified in the configuration file. Check that TIBCO EMS/Rendezvous is available.
SXF3036E	<%s> Failed to subscribe to Admin Reply subject <%s> cause=%s	An error was returned when Adapter attempted to subscribe to an ECM Admin Subject Reply name. Ensure that a valid subject name is specified in the configuration file. Check that TIBCO EMS/Rendezvous is available.
SXF3037E	NULL fileSender in Admin timer callback	A error was detected in the ECM Admin Timer callback support. Check that TIBCO EMS/Rendezvous is available. If so, and the problem persists, contact TIBCO technical support.

Number	Message	Description
SXF3038E	<%s> Failed to send Admin msg: cause=%s subject=%s	FilePublisher was unable to publish an ECM Admin message. Check that TIBCO EMS/Rendezvous is available.
SXF3040E	File <%s> can't be deleted. UserId=<%s> does not have ALTER access privileges on the file.	A "removeFile=true" was specified in a FileType definition, but when a Trigger request came in for that File, the requesting user is not authorized to delete the field.
SXF3042W	<%s> will not be deleted when Publishing complete. UserId=<%s> does not have ALTER access privileges on the file	A "removeFile=true" was specified in a FileType definition, but when a Trigger request came in for that File, the requesting user is not authorized to delete the field.

## SXF4000 – SXF4999

Number	Message	Description
SXF4002E	API %s failed with exception %s	A TIBCO API failed. Note the return code and contact your TIBCO administrator.
SXF4003E	Input file <%s> does not exist	A Trigger request was received, and the file it was requesting to publish was not present in the Process Input directory.
SXF4004E	File <%s> is present in process library	A copy of the requested file to be published already existed in the Process file. This means that the file is already being published or recovered, and the duplicate publish request is being rejected.
SXF4005E	API %s failed with exception %s	A TIBCO API failed. Note the return code and contact your TIBCO administrator.
SXF4006E	Library not found: %s	Could not found the library specified. Supply a name of a library that exists on the system.
SXF4007E	Input file <%s> does not exist	The input file specified did not exist. Correct the file name, or the library name.

Number	Message	Description
SXF4008E	File <%s> is present in process library.	The specified file already existed in the process directory library. Move the specified file out of the process directory library to prevent Adapter from overwriting the file.
SXF4502E	Generic error: %s %d %d %d %d	File Adapter is reporting an error it encountered. The %s details the cause of the error.
SXF4503E	Generic error: %s %s %d %d %d	File Adapter is reporting an error it encountered. The %s details the cause of the error.
SXF4504W	Library not found: %s	A library needed by the FileAdapter is missing. Re-check the STEPLIB definitions used in the File Adapter JCL.
SXF4505E	Failed to open Pub log file <%s>: cause %d %s. File logging turned off...	The File Adapter was unable to open the trace log file. The second %s details the cause of the problem.
SXF4506E	Error renaming Pub log file <%s> to <%s>: cause %d %s. File logging turned off...	The File Adapter was unable to open the trace log file. The second %s details the cause of the problem.
SXF4507E	Failed to open Pub log file <%s>: cause %d %s. File logging turned off...	The File Adapter was unable to open the trace log file. The second %s details the cause of the problem.

## Subscriber Error Messages

### SXF5000 – SXF5999

Number	Message	Description
SXF5135W	WARNING: VSAM RLS Share Options 3 and 4 are not supported	You might specify a SHARE option that the FileAdapter did not support. Only levels 1 and 2 are supported.
SXF5197E	Config parse error near line %d, failed to parse value	The format or value for a name/tag was not correct. A supplied value is incomplete; typically the value is missing an opening or closing quote ("), brace ({}), or an equal sign (=). Make sure all values in the configuration file have matching quotes and braces, and that there are equal signs where required. Check the name/value tags in the section defined in the configuration file. Refer to the manual for the proper format.
SXF5198E	Invalid value set for %s	The format or value for a name/tag was not correct. A supplied value is incomplete; typically the value is missing an opening or closing quote ("), brace ({}), or an equal sign (=). Make sure all values in the configuration file have matching quotes and braces, and that there are equal signs where required. Check the name/value tags in the section defined in the configuration file. Refer to the manual for the proper format.
SXF5199E	A VSAM key cannot be longer than 255 bytes. Length=%d	For VSAM files, a key cannot be more than 255 characters. Change the configuration value supplied to the proper length.
SXF5200E	Invalid VSAM file mode value set for %s	The file mode specified for a VSAM was incorrect. It must be either INSERT, UPSERT, or REPLACE. Change the configuration value supplied to a valid value.

Number	Message	Description
SXF5201E	Invalid vsamShare= option. Must be either NRI or CR	The share mode specified for a VSAM was incorrect. Change the configuration value supplied to a valid value.
SXF5202E	Unknown/Invalid configuration parameter <%s> at line %d. Ignored	In the FileSubscriber configuration file, a token was invalid in the FileType section. The invalid token is displayed. It could either be a misspelled option name, or an obsolete keyword. Change the configuration file for the associated FileType, to either correct the spelling of the option, or to remove an obsolete or unsupported option.
SXF5203E	Invalid file type: prefix more than 8 char	The file type value supplied must not be more than 8 characters. Change the value supplied to the proper length.
SXF5204E	File <%s> is not a VSAM file or else is not Cataloged	The file type specified a VSAM file name that could not be located in the MVS catalog. Verify that the file is actually a VSAM file, and ensure that the spelling of the file name is correct in the configuration file.
SXF5205E	VSAM log name has been defined, but no vsamUseLog= has been specified	The file type indicated a VSAM log name, but did not have a corresponding vsamUseLog option to denote how the log can be used. Add a vsamUseLog parameter, or else remove the VSAM log name parameter from the configuration file.
SXF5206E	Logging for VSAM file is enabled, but no VSAM log name has been defined	The file type had a vsamUseLog parameter specified, but no corresponding VSAM log name. Add a VSAM log name, or remove the vsamUseLog parameter from the configuration file.
SXF5207E	Block Mode Transfer requires that you specify an genFileSubjectName.	You specified blockTransferMode without specifying a genFileSubjectName. For standard block mode or record-mode ECM, specify a genFileSubjectName.
SXF5208E	Unsupported data set type <%s>	A file type other than SEQ was entered in the configuration file. Correct the file type entry, use only SEQ. Value must be in upper case.

Number	Message	Description
SXF5209E	VSAM file: missing KeyLength	For VSAM files, a key must be specified. Change the configuration file to supply the proper length.
SXF5210E	VSAM REUS files are not supported by Tibco Subscriber	FileSubscriber did not process VSAM files that have the REUS attribute. Redefine the file to NOREUS mode.
SXF5211E	Process directory [%s] cannot be the same as the output directory.	The process directory library name was the same as the output directory library. Use a unique name for the process and the output directories libraries.
SXF5212E	Invalid output library name: <%s>	The output library name was not a valid file name. Correct the name.
SXF5213E	Invalid output library name: <%s>	The output library name specified was invalid. Use a valid library name. Check for invalid characters. Maximum length is 10 characters.
SXF5214E	Invalid process library name: <%s>	The process library name specified was invalid. Use a valid library name. Check for invalid characters. Maximum length is 10 characters.
SXF5215E	non-Block Mode ECM FileSubscriber must have a file output method	A "standard" block-mode transfer that did not use ECM must specify a AutoGenerate file count or subject name. Supply a valid AutoGenerate parameter in the FileType section of the configuration file.
SXF5216E	Missing <%s> tag	The configuration file was missing a required constraint parameter. The error message indicates which parameter is missing. Check the configuration file.
SXF5217E	Do not use both <position> and <fieldStart> tags)	The configuration file uses both a position and a fieldStart tag. Use only one of the tags to specify a field location.
SXF5218E	Do not use both <position> and <fieldStart> tags)	The configuration file uses both a position and a fieldStart tag. Use only one of the tags to specify a field location.

Number	Message	Description
SXF5219E	Config parse error near line %d, invalid precision: <%s>	The value for the precision tag was not correct. Check the precision tag in the configuration file. Refer to the manual for the proper format.
SXF5220E	Config parse error near line %d, invalid precision: <%s>	The value for the precision tag was not correct. Check the precision tag in the configuration file. Refer to the manual for the proper format.
SXF5221E	Invalid data type <%s> near line %d>	A invalid field type was specified. Specify a valid field type for conversion to numeric.
SXF5222E	Config parse error near line %d, bad line token: '%s'	In the configuration file for FileSubscriber, a token is invalid in the FILE_LINE element of the FileType section. The invalid token is displayed. Correct or delete the invalid token.
SXF5223E	Field <%s> in <%s %s> must have a position definition...	A position tag was not supplied. Check the configuration file for the field identified, and add a position tag.
SXF5224E	Field <%s> in <%s %s> must have a precision definition for %s conversion...	A precision tag was not supplied. Check the configuration file for the field identified, and add a precision tag.
SXF5225E	Field <%s> in <%s %s> has invalid convertTo value: <%s>...	An invalid conversion value was specified. Only "true" or "false" might be specified for the convertTo tag.
SXF5226E	Field <%s> in <%s %s> has invalid input type: <%s>...	A invalid field type was specified. Specify a valid field type for conversion to numeric.
SXF5227E	Field <%s> in <%s %s> is defined as %s, but FileType must be binary...	The FileType must be binary. Add an isBinary="true" parameter to the file type.
SXF5228E	Config parse error near line %d, failed to parse value	A supplied value was incomplete. Typically the supplied value is missing an opening or closing quote "), brace ({ }), or an equal sign (=). Make sure that the values in the configuration file have matching quotes and braces, and that there are equal signs where required.

Number	Message	Description
SXF5229E	Config parse error near line %d, bad line token: '%s'	In the configuration file for FileSubscriber, a token was invalid in the FILE_LINE element of the FileType section. The invalid token is displayed. Correct or delete the invalid token.
SXF5230E	Config parse error near line %d, failed to parse value	A supplied value was incomplete, typically the supplied value is missing an opening or closing quote ("), brace ({ }), or an equal sign (=). Make sure that the values in the configuration file have matching quotes and braces, and that there are equal signs where required.
SXF5231E	Config parse error near line %d, failed to parse value	A supplied value was incomplete, typically the supplied value is missing an opening or closing quote ("), brace ({ }), or an equal sign (=). Make sure that the values in the configuration file have matching quotes and braces, and that there are equal signs where required.
SXF5232E	Config parse error near line %d, invalid option name: '%s'	A value in the Options section might be misspelled, or might not have the proper case. The invalid option is displayed. Check the spelling of the values in the Options section of the configuration file. Values must be in upper case.
SXFF5233E	Config parse error near line %d, failed to parse options line	In the configuration file, the Options section must had matching opening and closing '}'. Check the configuration file and make sure that the Options section has matching opening and closing '}'.
SXF5234E	Config parse error near line %d, failed to parse value	A supplied value was incomplete, typically the supplied value is missing an opening or closing quote ("), brace ({ }), or an equal sign (=). Make sure that the values in the configuration file have matching quotes and braces, and that there are equal signs where required.



Number	Message	Description
SXF5235E	Config parse error near line %d, failed to parse value	A supplied value was incomplete. Typically the supplied value is missing an opening or closing quote ("), brace ({}), or an equal sign (=). Make sure that the values in the configuration file have matching quotes and braces, and that there are equal signs where required.
SXF5236E	Config parse error near line %d, invalid option name: '%s'	A value in the Options section might be misspelled, or might not have the proper case. The invalid option is displayed. Check the spelling of the values in the Options section of the configuration file. Values must be in upper case.
SXF5237E	Config parse error near line %d, failed to parse file type line	In the configuration file, the FileType did not have the proper framing, or was missing required parameters. Check the FileType section of the configuration file and make sure that the section is complete, and that there are matching opening and closing braces ({}).
SXF5238E	Config parse error near line %d, invalid option name: '%s'	A value in the Options section might be misspelled, or might not have the proper case. The invalid option is displayed. Check the spelling of the values in the Options section of the configuration file. Values must be in upper case.
SXFF5239E	Config parse error near line %d, failed to parse options line	A supplied value was incomplete, typically the supplied value was missing an opening or closing quote ("), brace ({}), or an equal sign (=). Make sure that the values in the configuration file have matching quotes and braces, and that there are equal signs where required.
SXF5240E	Only %d file types are supported	You might specify too many File Types in the configuration file. Reduce the number of File Type definitions

Number	Message	Description
SXF5241E	Config parse error near line %d, invalid section name: '%s'	A section name in the configuration file was incorrect. The section name must be one of Trace, Options, or FileType. Check the spelling of the section names of the configuration file. Names are case sensitive.
SXF5242E	Config parse error near line %d, failed to find ']'	In the configuration file, a section description (Trace, Options, or FileType) did not have a closing ']'. Check the configuration file and make sure that all section descriptions are formed correctly.
SXF5243E	Config parse error near line %d, failed to find section'	A section name in the configuration file was missing. The section name must be one of Trace, Options, or FileType. Check the spelling of the section names in the configuration file. Names are case sensitive.
SXF5244E	Config parse error near line %d, failed to terminate expression	A section was not properly configured in the configuration file. Check the configuration file and correct the error.
SXF5245E	Missing [%s] section	A section name in the configuration file was missing. The section name must be one of Trace, Options, or FileType. Check the spelling of the section names in the configuration file. Names are case sensitive.
SXF5246E	Missing [%s] section	A section name in the configuration file was missing. The section name must be one of Trace, Options, or FileType. Check the spelling of the section names in the configuration file. Names are case sensitive.
SXF5247E	Missing [%s] section	A section name in the configuration file was missing. The section name must be one of Trace, Options, or FileType. Check the spelling of the section names in the configuration file. Names are case sensitive.
SXF5248E	Missing [%s] section option: %s	The configuration file was missing the section option specified. You must specify the indicated option. Add the option to the configuration file.

Number	Message	Description
SXF5249E	Missing [%s] section option: %s	The configuration file was missing the section option specified. You must specify the indicated option. Add the option to the configuration file.
SXF5250E	Missing [%s] section option: %s	The configuration file was missing the section option specified. You must specify the indicated option. Add the option to the configuration file.
SXF5251E	Missing [%s] section option: %s	The configuration file was missing the section option specified. You must specify the indicated option. Add the option to the configuration file.
SXF5252E	Missing [%s] section option: %s	The configuration file was missing the section option specified. You must specify the indicated option. Add the option to the configuration file.
SXF5253E	Missing %s definitions for file prefix <%s>	An input directory library had not been specified for this file type. Correct the configuration file.
SXF5254E	Missing %s definition for file prefix <%s>	The specified section for this file type was missing. Add the definitions to the configuration file.
SXF5255E	GDG Model not found: %s	The GDG model for the specified file was not found. Make sure that the GDG file was properly specified.
SXF5256E	Missing %s definition for file prefix <%s>	The specified section for this file type was missing. Add the definitions to the configuration file.
SXF5257E	Directory Library not found: %s	The specified directory library was not found. Make sure that the specified directory library exists and is available.
SXF5258E	Directory Library not found: %s	The specified directory library was not found. Make sure that the specified directory library exists and is available.

Number	Message	Description
SXF5259E	Missing %s definition for file prefix <%s>	The specified section for this file type was missing. Add the definitions to the configuration file.
SXF5260E	Directory Library not found: %s	The specified directory library was not found. Make sure that the specified directory library exists and is available.
SXF5261E	Directory Library not found: %s	The specified directory library was not found. Make sure that the specified directory library exists and is available.
SXF5262E	Failed to open config file: %s	Unable to open the configuration file or the configuration file was not found. Make sure that the configuration file exists, and that it can be found by Adapter.
SXF5263E	MVS CSI problem. Incorrect file name or not cataloged=<%.44s>	The File Adapter cannot find a valid entry for this file in the MVS catalog. It is either mis-spelled or has been deleted from the MVS catalog.
SXF5264E	Failed to open TIB/Rendezvous RV. cause=%s	Unable to start TIBCO Rendezvous, it might be inactive. Check with your TIBCO administrator that TIBCO Rendezvous is available.
SXF5265E	Failed to create EMS/RV Transport session: service=%s network=%s daemon=%s. EMS/RV Error=%s	Unable to start TIBCO Rendezvous, it might be inactive. Check with your TIBCO administrator that TIBCO Rendezvous is available.
SXF5266E	Failed to open TIBCO Rendezvous CM -- %s	Unable to start TIBCO Rendezvous, it might be inactive. Check with your TIBCO administrator.
SXF5267E	Failed to create RV transport: %s	Unable to create a network transport. TIBCO Rendezvous might be inactive. Check with your TIBCO administrator.
SXF5268E	Failed to create RVCM Transport: %s	Unable to create a network transport. TIBCO Rendezvous might be inactive. Check with your TIBCO administrator.

Number	Message	Description
SXF5269E	Configuration File Error near line <%d> missing ' ' .	An error occurred in the configuration file. Verify that the configuration file is coded properly.
SXF5270E	Found definitions for both Reliable and Certified mode	The configuration file contained definitions for both modes. The modes are mutually exclusive. Specify only one mode in the configuration file.
SXF5271E	Found definitions for both Reliable and Certified mode	The configuration file contained definitions for both modes. The modes are mutually exclusive. Specify only one mode in the configuration file.
SXF5272E	Missing tag <%s> for %s	The configuration file was missing a required parameter. The error message indicates which parameter is missing. Check the configuration file and correct the discrepancy.
SXF5273E	Missing tag <%s> for %s	The configuration file was missing a required parameter. The error message indicates which parameter is missing. Check the configuration file and correct the discrepancy.
SXF5274E	Cannot specify both vsamReplaceRecords and vsamUpsertRecords =true	The FileType definition had both parameters specified. This is not allowed. Choose one or the other.
SXF5275E	VSAM file type: missing KeyOffset	A FileType was using a VSAM file. VSAM files require a KeyOffset definition in the FileType.
SXF5276E	Memory allocation error of LineField struct. Length=%d	Unable to allocate memory. Report error to your TIBCO administrator.
SXF5277E	lineBuff Option Memory allocation failed. Length=%d	Unable to allocate memory. Report error to your TIBCO administrator.
SXF5278E	optionvalue Memory allocation failed. Length=%d	An attempt to allocate memory failed. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.

Number	Message	Description
SXF5279E	optionvalue Memory allocation failed. Length=%d	Unable to allocate memory. Report error to your TIBCO administrator.
SXF5280E	lineBuff Option Memory allocation failed. Length=%d	Unable to allocate memory. Report error to your TIBCO administrator.
SXF5281E	lineBuff Option Memory allocation failed. Length=%d	Unable to allocate memory. Report error to your TIBCO administrator.
SXF5282E	lineBuff Option Memory allocation failed. Length=%d	Unable to allocate memory. Report error to your TIBCO administrator.
SXF5283E	Missing %s definition for file prefix <%s>	A FileType required a Prefix definition, and none was found.
SXF5284E	Missing %s definition for file prefix <%s>	A FileType required a Prefix definition, and none was found.
SXF5285E	GDG base not found: %s	The GDG base for the specified file was not found. Make sure that the GDG file was properly specified.
SXF5286E	Missing %s definition for file prefix <%s>	A FileType required a Prefix definition, and none was found.
SXF5287E	Missing %s definition	No Adapter name had been specified. Add the definitions to the configuration file.
SXF5288E	FileOptions Memory allocation failed	An attempt to allocate memory failed. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF5289E	Config parse error near line %d, failed to find ']'	An error occurred in the configuration file. Verify that the configuration file is coded properly.
SXF5290E	*** CSI failure: mod=0x%X%X reason=%d rc=%d	A request to access the MVS Catalog service failed. Save the error message and contact your MVS Systems programmer.
SXF5291E	Invalid file type: no prefix	A FileType required a Prefix definition, and none was found.

Number	Message	Description
SXF5292W	Caution: Binary files must always be Fixed Block. usedFixedRecordFile=false parameter is ignored.	Adapter required that binary files must use fixed block (FB) format. Correct the configuration file.
SXF5293E	There is a config error. reading till next section is encountered :[%d]	A supplied value was incomplete. Typically the supplied value is missing an opening or closing quote ("), brace ({ }), or an equal sign (=). Make sure that the values in the configuration file have matching quotes and braces, and that there are equal signs where required.
SXF5294E	There is an error while parsing the name value pairs. exit parsing	A supplied value was incomplete. Typically the supplied value is missing an opening or closing quote ("), brace ({ }), or an equal sign (=). Make sure that the values in the configuration file have matching quotes and braces, and that there are equal signs where required.
SXF5295E	There is an configuration error at line <%d>:[%s]	A supplied value was incomplete. Typically the supplied value is missing an opening or closing quote ("), brace ({ }), or an equal sign (=). Make sure that the values in the configuration file have matching quotes and braces, and that there are equal signs where required.
SXF5296E	blkSizeAlloc <%d>, is not a multiple of lineLength [%d] for fixed block	The value specified for blkSizeAlloc= was not a multiple of the record length (lineLength). Recheck the RECL and BLKZSIZE attributes of the file on your system, and set blkSizeAlloc to a valid multiple of the RECL.
SXF5297E	blkSizeAlloc <%d>, is not a multiple of lineLength [%d] for variable block	The value specified for blkSizeAlloc= was not a correct multiple of the record length (lineLength). It must be (RECL * n) + 4. Re-check the RECL and BLKZSIZE attributes of the file on your system, and set blkSizeAlloc to a valid multiple+4 of the RECL.

Number	Message	Description
SXF5299W	Target file <%.48s> not found. Requested by user=%s on Subject/Destination=%s.	The File Adapter cannot find a valid entry for this file. It is either mis-spelled or has been deleted.
SXF5302W	Both Code Pages must be configured. Single code page config is ignored.	Normally, both HOSTCODEPAGE and NETWORKCODEPAGE have to be specified when overriding the default code pages used by the File Adapter.
SXF5303W	Code Page Setup failed: reason=<%s>. Will use defaults instead.	Unable set up the requested code pages. Verify that the code page settings are valid for your system and that TIBCO Rendezvous is active. Check with your TIBCO administrator.
SXF5306E	Config ERROR: you have defined both record-mode ECM and Block-Mode ECM parms in the same Filetype. Configuration has been overridden to use Block Mode ECM.	You might specify parameters for both Block Mode ECM and record-mode ECM versions of ECM block mode. Determine which version of the ECM protocol you, then remove the parameters that apply to the other version of the protocol.
SXF5307E	Config ERROR: you have defined both forcePublishedFileName and generateFileOnNumberOfMessages=%d parms. This combination is not allowed.	The file type had been defined with both forcePublishedFileName and generateFileOnNumberOfMessages parameters. Remove one of the parameters from the configuration file.
SXF5323E	Config ERROR: you must define retryInterval > 0 if Locked File retry is enabled with noOfRetries > 0	An error occurred in the configuration file. Verify that the configuration file is coded properly.
SXF5325E	<%s> Invalid %s=%d. Must be less than 32768.	Cannot set "blockSizeAlloc" to a value more than 32767.
SXF5326E	Invalid %s setting=%s. Must be RecordMode, RecordModeECM, BlockModeSFT, or BlockModeECM	Valid values for the transferType parameter are RecordMode, RecordModeECM, BlockModeSFT, or BlockModeECM.



**SXF6000 – SXF6999**

Number	Message	Description
SXF6013E	Error %d on timer create: %s	Unable to create the RV heartbeat callback timer.
SXF6019W	Received Signal, exiting app...	Cancel or error signal received; exiting application.
SXF6020W	Could not remove SIGINT signal handler: %s	The File Adapter encountered an error while terminating. If the problem persists, contact TIBCO technical support.
SXF6021W	Could not remove SIGTERM signal handler: %s	The File Adapter encountered an error while terminating. If the problem persists, contact TIBCO technical support.
SXF6022W	Could not setup SIGINT signal handler: cause=%s	The File Adapter encountered an error while terminating. If the problem persists, contact TIBCO technical support.
SXF6023W	Could not setup SIGTERM signal handler: cause=%s	The File Adapter encountered an error while terminating. If the problem persists, contact TIBCO technical support.
SXF6024E	Error in sending Heartbeat message: cause=%s Subject/Destination=%s	An error occurred when sending a TIBCO heartbeat message. Adapter might not be active. Contact your TIBCO administrator.
SXF6025E	Error in sending Heartbeat message: cause=%s Subject/Destination=%s	An error occurred when sending a TIBCO heartbeat message. Adapter might not be active. Contact your TIBCO administrator.
SXF6026E	Failed to unsubscribe for exit Subject/Destination <%s>: cause=%s	The File Adapter encountered an error while terminating. If the problem persists, contact TIBCO technical support.
SXF6027E	Failed to unsubscribe for refresh options Subject/Destination <%s>: %s	The File Adapter encountered an error while terminating. If the problem persists, contact TIBCO technical support.
SXF6030E	Heartbeat timer removal failed: cause=%s	The File Adapter encountered an error while terminating. If the problem persists, contact TIBCO technical support.

Number	Message	Description
SXF6031E	Failed to send error message: cause=<%s> Subject/Destination=%s	File Adapter was unable to send a message to the network. The %s cause details why. It might indicate a network or RV daemon problem.
SXF6032E	FileHandler Global-List Memory allocation failed	An attempt to allocate memory failed. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF6033E	Failed to subscribe to Subject/Destination <%s> cause=%s	Unable to create a listener. TIBCO EMS/Rendezvous might be unavailable. Check with your TIBCO administrator.
SXF6034E	FileHandler Memory allocation failed for new [FileType]. Length=%d	An attempt to allocate memory failed. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF6035E	Memory allocation failed for SaveTimer entry. Length=%d	An attempt to allocate memory failed. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF6036E	HeartBeat Msg Memory allocation failed. Length=%d	An attempt to allocate memory failed. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF6037E	HeartBeat Nested Msg Memory allocation failed. Length=%d	An attempt to allocate memory failed. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF6038E	Failed to create Heartbeat message: cause=%s	FileSubscriber was unable to create a heartbeat message. Check that TIBCO EMS/Rendezvous is available.

Number	Message	Description
SXF6039E	Failed to add [%s] to Heartbeat message: %s	Unable to add the specified string to a heartbeat message. Check that TIBCO EMS/Rendezvous is available.
SXF6040E	Failed to add [%s] to Heartbeat message: %s	Unable to add the specified string to a heartbeat message. Check that TIBCO EMS/Rendezvous is available.
SXF6041E	Failed to add file prefix for: %d:%s cause=%s	Unable to add the file prefix field to a heartbeat message. Check that TIBCO EMS/Rendezvous is available.
SXF6042E	Failed to add file extension for %d:%s cause=%s	Unable to add the file extension to a heartbeat message. Check that TIBCO EMS/Rendezvous is available.
SXF6043E	Failed to add status for: %d:%s cause=%s	Unable to add the status string to the heartbeat timer message. Check with the TIBCO administrator.
SXF6044E	Failed to add Begin Time for: %d:%s cause=%s	Unable to add the Begin Time string to the heartbeat timer message. Check with the TIBCO administrator.
SXF6045E	Failed to add NumMsgs for: %d:%s cause=%s	Unable to add the NumMsgs string to the heartbeat timer message. Check with the TIBCO administrator.
SXF6046E	Failed to add BlockNum for: %d:%s cause=%s	Unable to add the BlockNum string to the heartbeat timer message. Check with the TIBCO administrator.
SXF6047E	Failed to add LineNum for: %d:%s cause=%s	Unable to add the LineNum string to the heartbeat timer message. Check with the TIBCO administrator.
SXF6048E	Aborting heartbeat message publish	Too many previous AddMessage errors forced FileSubscriber to abandon sending a heartbeat timer message. Check with the TIBCO administrator.

Number	Message	Description
SXF6049E	Failed to set Heartbeat send Subject/Destination: %s cause=%s	An error was returned when Adapter attempted to set the subject/destination name in a Heartbeat message to be published. Ensure that a valid subject/destination name is specified in the configuration file.
SXF6050E	Failed to destroy message.	Unable to destroy a published heartbeat message. Make sure that TIBCO EMS/Rendezvous is available.
SXF6051E	Failed to create error message: cause=%s	An attempt to create a RV or EMS message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF6052E	Failed to add [%s] to error message: %s	An attempt to add a field to a RV or EMS message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF6053E	Failed to set error send Subject/Destination: %s cause=%s	An attempt to add a field to a RV or EMS message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF6054E	Failed to send error message: cause=<%s> Subject/Destination=%s	File Adapter was unable to send a message to the network. The %s cause details why. It might indicate a network or RV daemon problem.
SXF6055E	Failed to destroy error message: reason=%s	File Adapter encountered an error when destroying. If it persists, contact TIBCO technical support.

Number	Message	Description
SXF6056E	Config file not found in PARM= and no INIFILE DD. Configuration file from -config parameter and QINLINE file not found. Will use default config filename <%s> instead.	No configuration file was specified. This is probably an error by the operator. Double-check that you really want to use the default configuration file.
SXF6057E	Failed to get hostname from system. Error code returned by 'gethostname' is = %d	Unable to extract the host name being used by TCPIP. Probably indicates that TCP/IP is not up or not functioning.
SXF6058E	Could not find a usable Sockets DLL connection.	The TCP Steplib entry was missed.
SXF6059E	Failed to get hostname from system. Error code returned by 'gethostname' is = %d	Unable to extract the host name being used by TCPIP. The TCP/IP was not up or not functioning.
SXF6060E	setenv for EDC_ZERO_RECLEN Failed: %d %s. VB TEXT files exposed	A OMVS setenv() was issued to allow proper operation of VB files, and it failed because of the reason specified in %s. Review this issue with the MVS systems programmer. It probably indicates an OMVS configuration or authorization problem.
SXF6061E	Unable to open security mapping file: %s cause=%d %s.	The security mapping file might be mis-named or under RACF protection.
SXF6062E	Security mapping file %s has invalid data: Line=%d <%s>	An error occurred in the security mapping file. Verify that the configuration file is coded properly.
SXF6063E	Memory allocation failed while processing Security mapping file %s at Line=%d	An attempt to allocate memory failed. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF6065E	Failed to create the Default Data RV queue-- reason: %s	TIBCO EMS/Rendezvous might be unavailable or might not be active. Check with your TIBCO administrator.

Number	Message	Description
SXF6066E	Failed to set the RV limit policy-- reason: %s	TIBCO EMS/Rendezvous might be unavailable or might not be active. Check with your TIBCO administrator.
SXF6067E	Failed to subscribe to Subject/Destination <%s> reason: %s	TIBCO EMS/Rendezvous might be unavailable or might not be active. Check with your TIBCO administrator.
SXF6068E	Failed to create the Priority Admin RV queue-- reason: %s	TIBCO EMS/Rendezvous might be unavailable or might not be active. Check with your TIBCO administrator.
SXF6069E	Failed to set the RV Queue Priority to %d for %s. Reason: %s	Setting RV Queue Priority via tibrvQueue_SetPriority API had failed.
SXF6070E	Error creating EMS Destination/Producer for %s dest=<%s>: cause=%s	An error occurred when creating an EMS Destination/Producer for EPM, or ERROR msgs, or EMS HEARTBEAT in Subscriber.

## SXF7000 – SXF7999

Number	Message	Description
SXF7001E	<%s> TEXT fopen failed. cause=%d: %s	The open of a TEXT file failed. The cause contains the reason for the failure.
SXF7002E	<%s> fclose failed. cause=%d: %s	The close of a file failed. The cause contains the reason for the failure.
SXF7003E	<%s> BINARY fopen failed. cause=%d: %s	The open of a BINARY file failed. The cause contains the reason for the failure.
SXF7004E	<%s> fclose failed. cause=%d: %s	The close of a file failed. The cause contains the reason for the failure.
SXF7005E	<%s> Generating file <%s> due to loss of data	A target file is being generated, but data had been lost or truncated. This indicates either an out of space error, or a configuration (incorrect file size) error.

Number	Message	Description
SXF7006E	<%s> Exiting because of file save error. Deleted PRP and PRG	A target file could not be properly saved and closed. The File Adapter was configured to shutdown in such a case.
SXF7026E	<%s> close failed during trailer append. errno=%d errtext=%s	The close of a file failed after file trailer data was added. The cause contains the reason for the failure.
SXF7032E	<%s> close failed during header append. errno=%d errtext=%s	The close of a file failed after file header data was added. The cause contains the reason for the failure.
SXF7035W	<%s> Recovery mode: Generating error file=<%s> due to %s	The File Adapter is performing recovery after a shutdown or crash, and is generating an ERR file because of an error encountered.
SXF7036W	<%s> File transfer: Generating error file=<%s> due to %s	The File Adapter was generating a file but received an error, so it was converting it to an ERR file.
SXF7041W	<%s> remove of PREV file failed because PREV file does not exist. errno=%d errtext=%s	A file type was configured to REPLACE an existing file, but there was a problem with the old file it was replacing, as described in the errtext portion.
SXF7042W	<%s> remove of PREV file failed. errno=%d errtext=%s	A file type was configured to REPLACE an existing file, but there was a problem with the old file it was replacing, as described in the errtext portion.
SXF7045W	<%s> GDG dataset=<%s> close failed. errno=%d errtext=%s	The close of a GDG file failed. The errtext contains the reason for the failure.
SXF7048W	GDG data set <%s> close failed. errno=%d errtext=%s	The close of a GDG file failed. The errtext contains the reason for the failure.
SXF7067W	<%s> GDG fclose failed during recovery. cause=%d: %s	The close of a file failed. The cause contains the reason for the failure.
SXF7068E	<%s> Recovery GDG fopen failed using file=<%s> mode=<%s>.\n cause=%d: %s	The close of a file failed. The cause contains the reason for the failure.

Number	Message	Description
SXF7069E	<%s> Recovery SEQ fopen failed using file=<%s> mode='r'. cause=%d: %s	The close of a file failed. The cause contains the reason for the failure.
SXF7070W	<%s> SEQ fclose failed during recovery. cause=%d: %s	The close of a file failed during Recovery/Restart processing. The cause contains the reason for the failure.
SXF7088W	<%s> Truncating TEXT data - discarded %d bytes. expectnl=%d	The data received for a FileType exceeds the max length configured for the File Type. The data has been truncated.
SXF7089W	<%s> Truncating TEXT data - discarded %d bytes. expectnl=%d	The data received for a FileType exceeds the max length configured for the File Type. The data has been truncated.
SXF7102W	Retry timer pop: retry # %d trying to move .CWK to target file=<%s>. (timer interval=%d secs)	A target file was temporarily locked by another job. This indicates when the File Adapter attempts to again try to see if the file is now free.
SXF7103E	Move operation could not remove file <%s>. Deleting PRP and PRG	Could not remove the specified file. The transfer is being aborted. See the previous error message in the trace log for the cause of the problem.
SXF7104W	<%s> ### Continuing to accept data in working file	New data had been received into the CWK file after a previous error (e.g. failed Generate). The File Adapter continues to try to process the incoming data.
SXF7106W	<%s> Retry limit=<%d> for Target File Locked <%s> has been reached. Deleting PRP and PRG files too.	A target file that was temporarily locked by another job has exceeded its max retry limit. The received new file data is converted to an ERR file.
SXF7107E	<%s> Exiting because of file save error	A target file could not be properly saved and closed. The File Adapter was configured to shutdown in such a case.
SXF7108W	<%s> ### Continuing to accept data in working file	New data had been received into the CWK file after a previous error (e.g. failed Generate). The File Adapter continues to try to process the incoming data.



Number	Message	Description
SXF7117W	Parse error: bad variable name <%s>	Encountered an invalid variable name. Correct identified invalid variable name in the configuration file.
SXF7118W	Parse error: missing end of variable	The configuration parser program was unable to properly parse a variable. Check the configuration file for adding header or trailer records. These must be framed with '% '.
SXF7119W	<%s> Move operation could not remove workfile: cause=%d: %s. Caller=%s	Could not remove the specified file. The transfer is being aborted. The cause contains the reason for the error.
SXF7120W	Failed to convert time value, using current time	Unable to convert a TIME value from a received message, and it uses the current system time for the TIME field in an output record. Check the configuration file to ensure that the field identified contains a TIME value.
SXF7121W	Failed to extract nested msg. field <%s>	Unable to construct the subject/destination from the message node. Check the configuration file for this message.
SXF7122W	Failed to extract field <%s> to form Subject/Destination name	Unable to construct the subject/destination from the field data. Check the configuration file for this message.
SXF7123W	Failed to write data from file <%s> - wrote %d out of %d	A file write failed. It is usually because of an out of space error. See the previous error message in the trace log for the exact cause.
SXF7124W	Move operation could not remove file <%s>	Could not remove the specified file. The transfer is being aborted. See the previous error message in the trace log for the cause of the problem.
SXF7125W	Move operation could not remove file <%s>	Could not remove the specified file. The transfer is being aborted. See the previous error message in the trace log for the cause of the problem.

Number	Message	Description
SXF7126W	<%s> Block number %d arrived before IFACTIVE_REPLY Ack, Discarding the block.	A timing race occurred between Block Mode ECM Publisher and Subscriber at startup. The message block is discarded and re-transmitted once the startup handshake sequence is complete.
SXF7127W	<%s> Dup Block number %d matches previous block (%d), skipping write for workfile	FileSubscriber had received a block of data that had already been received. The block is ignored.
SXF7128W	<%s> Block number %d is not next in sequence. Expected blk=%d	FileSubscriber received a block of data out of order.
SXF7129W	<%s> TEXT file write fixed width Field failed for workfile	A file write failed. It is usually because of an out of space error. See the previous error message in the trace log for the exact cause.
SXF7130W	<%s> TEXT file write padded Field Value failed	A file write failed. It is usually because of an out of space error. See the previous error message in the trace log for the exact cause.
SXF7131W	Failed to convert time value, using current time	Unable to convert a TIME value from a received message, and it uses the current system time for the TIME field in an output record. Check the configuration file to ensure that the field identified contains a TIME value.
SXF7132W	<%s> Workfile not removed. Will be renamed to .ERR	Source file was retained while moving a file. It is possible that the source file was in use by another process when Adapter attempted to move the file. Remove the source file if it is not needed.
SXF7133W	<%s> Move operation could not remove workfile: cause=%d: %s	Source file was retained while moving a file. It is possible that the source file was in use by another process when Adapter attempted to move the file. Remove the source file if it is not needed.
SXF7134W	<%s> Move operation could not remove workfile: cause=%d: %s	Source file was in use by another process when Adapter attempted to remove the file

Number	Message	Description
SXF7135W	<%s> Timer: Move operation could not remove workfile. Deleting PRP and PRG	Source file was in use by another process when Adapter attempted to remove the file
SXF7136W	Received badly formatted message	A message was received that does not match the configuration file. The configuration file/field specification for a received message does not correspond to the message received. Correct the file/field specification for the subject/destination being subscribed to, or check the source that published the message.
SXF7137W	Received badly formatted message	A message was received that does not match the configuration file. The configuration file/field specification for a received message does not correspond to the message received. Correct the file/field specification for the subject/destination being subscribed to, or check the source that published the message.
SXF7138E	Failed to unsubscribe to Subject/Destination <%s>: cause=%s	During shutdown, FileSubscriber was unable to terminate the subscription to a Subject. The cause %s specifies what the problem was
SXF7139E	Failed to unsubscribe to Subject/Destination <%s>: cause=%s	During shutdown, FileSubscriber was unable to terminate the subscription to a Subject. The cause %s specifies what the problem was
SXF7140E	Failed to unsubscribe to Subject/Destination <%s>: cause=%s	During shutdown, FileSubscriber was unable to terminate the subscription to a Subject. The cause %s specifies what the problem was
SXF7141E	Failed to unsubscribe to Subject/Destination <%s>: cause=%s	During shutdown, FileSubscriber was unable to terminate the subscription to a Subject. The cause %s specifies what the problem was
SXF7142E	Retry Timer removal failed: cause=%s	During shutdown, FileSubscriber was unable to terminate RV or EMS. The cause %s specifies the problem.
SXF7143E	Vsam LOG write failed. Abending File Adapter	The write operation on the log file had failed and Adapter is abending. This happens when the vsamUseLog parameter is set to "StopOnFull".

Number	Message	Description
SXF7144E	Vsam LOG write failed on <%s>. Abending File Adapter	Writing to the log failed; Adapter abends.
SXF7145E	Out of space condition reached while writing out to file <%s>. Terminating processing for that file.	The I/O to the file had reached an out of space condition. Any subsequent data destined for that file is rejected and discarded.
SXF7146E	Out of space condition reached while writing out to file <%s>. Terminating processing for that file.	The I/O to the file had reached an out of space condition. Any subsequent data destined for that file is rejected and discarded.
SXF7147E	%s Data set type (PDS) is not supported in this application	The data set supported for this application must not be PDS. PDS data sets cannot be used for FileSubscriber, you must use either SEQ or GDG data sets. Change the data set type to SEQ or GDG.
SXF7148E	Confirm Message create RVMsg failed	Unable to allocate memory to create a new message. Report this error to your TIBCO administrator.
SXF7149E	Compression block Memory reallocation failed. Length=%d	An attempt to re-allocate memory failed. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF7150E	<%s> Error in sending RVCM message: cause=%s Subject/Destination=%s	An error occurred when sending an RVCM message. Adapter might not be active. Contact your TIBCO administrator.
SXF7151E	<%s> Error in sending message: cause=%s	An error occurred when sending a message. Adapter might not be active. Contact your TIBCO administrator.
SXF7152E	NULL FileHandler in retry timer callback	An internal error occurred. Contact TIBCO technical support.
SXF7153E	Unable to extract CM message seqnum: %s	An internal error occurred. Contact TIBCO technical support.

Number	Message	Description
SXF7154E	Unable to extract publisher name: %s	An internal error occurred. Contact TIBCO technical support.
SXF7155E	Failed to subscribe to Subject/Destination <%s>: cause=%s	Unable to create a listener. TIBCO EMS/Rendezvous might be unavailable. Check with your TIBCO administrator.
SXF7156E	Failed to subscribe to Subject/Destination <%s>: cause=%s	Unable to create a listener. TIBCO EMS/Rendezvous might be unavailable. Check with your TIBCO administrator.
SXF7157E	Failed to subscribe to Subject/Destination <%s>: cause=%s	Unable to create a listener. TIBCO EMS/Rendezvous might be unavailable. Check with your TIBCO administrator.
SXF7158E	Failed to subscribe to Subject/Destination <%s>: cause=%s	Unable to create a listener. TIBCO EMS/Rendezvous might be unavailable. Check with your TIBCO administrator.
SXF7159E	<%s> Error moving workfile to target file=%s. Reason=%s. errno=%d file_rc=%d secur_rc=%d append=%d eof=%d	A target file could not be properly saved and closed. The reason code describes what the problem was.
SXF7160E	<%s> File open failed for workfile using mode '%s'. errno=%d R15=%d Reason Fdbk=%d op=%d ab_sys=0x%X ab_rc=0x%X	A CWK work file could not be opened. The reason code describes what the problem was.
SXF7161E	<%s> File was empty, now has data. Will re-open in non-LOAD mode	A VSAM data transfer encountered a target file that was completely empty. The File Adapter re-opens the file and treat the transfer as an initial LOAD of the VSAM file.
SXF7162E	<%s> Can not open file because RLS function is inactive. SMSVSAM not up	A shared VSAM file could not be processed, because the MVS RLS function is not active. Contact your MVS systems programmer.
SXF7163E	<%s> Can not open file because trying to output to a Recoverable Sphere	A shared VSAM file could not be processed because it was defined as being recoverable. Contact your MVS systems programmer.

Number	Message	Description
SXF7164E	<%s> Can not open file because RLS requires SMS files.File is not SMS managed	A shared VSAM file could not be processed, because it was not defined as SMS managed. Contact your MVS systems programmer.
SXF7165E	<%s> Can not open file because RLS detected DISP=SHR vs DISP=OLD conflict	A shared VSAM file could not be processed, because of conflicting DISP attributes. Contact your MVS systems programmer.
SXF7166E	<%s> Can not open file because OPEN flag still set on from previous run or another App is trying to share the file.	A VSAM file could not be opened because it's file flag is set indicating that it was not properly closed from a previous job. Contact your MVS systems programmer.
SXF7167E	<%s> Failed to open workfile: errno=%d err text=%s	The open of a file failed. The err text contains the reason for the failure.
SXF7168E	<%s> Failed to malloc TEXT LRECL buffer of length=%d	An out of memory error was reached.
SXF7169E	File open failed for LOG file <%s> using mode '%s'. errno=%d R15=%d Fdbk=%d op=%d ab_sys=0x%X ab_rc=0x%X\n errno text=%s	The open of a FileAdapter VSAM LOG file failed. The errno text contains the reason for the failure.
SXF7170E	<%s> Failed to open Progress file: <%s> errno=%d errtext=%s	The open of a FileAdapter PRG progress file failed. The errtext contains the reason for the failure.
SXF7171E	<%s> the JCL <%s> execution failed. cause=%s	The JCL specified for executeAfterProcess or executeBeforeProcess did not execute properly. Check the JCL specified, and ensure that it can execute properly.
SXF7172E	<%s> Command <%s> returned error rc=%d	The command or program specified for executeAfterProcess or executeBeforeProcess did not execute properly.
SXF7173E	<%s> the JCL <%s> execution failed. cause=%s	The JCL specified for executeAfterProcess or executeBeforeProcess did not execute properly. Check the JCL specified, and ensure that it can execute properly.

Number	Message	Description
SXF7174E	<%s> Command <%s> returned error rc=%d	The command identified did not complete successfully. Check the command specified for executeBeforeProcess or executeAfterProcess and ensure that the commands are valid.
SXF7175E	<%s> Error saving next GDG generation number for GDG=<%s>	Creation of a new target GDG file failed. See the previous error message in the trace log for the cause of the error.
SXF7176E	<%s> Error freeing GDG file, error=%s	The free/deallocate of a GDG file failed. See the previous error message in the trace log for the cause of the error.
SXF7177E	<%s> Error deallocating GDG=<%s>\n reason: %s	Unable to deallocate a dynamically allocated GDG file. Either the target data set is not available or is invalid. Provide a valid data set name in the configuration file. You must have the proper authority to create or modify data sets.
SXF7178E	<%s> Error moving workfile to GDG target file=%s	Unable to move a file. Either the target data set is not available or is invalid. Provide a valid data set name in the configuration file.
SXF7179E	<%s> Error creating GDG file=<%s>, error=%s	File subscriber attempted to create a GDG file for output, but the file create failed. You must have the proper authority to create or modify data sets.
SXF7180E	<%s> Error moving workfile to target file=%s	Unable to move a file. Either the target data set is not available or is invalid. Provide a valid data set name in the configuration file.
SXF7181E	<%s> Error moving workfile to target file=%s	Unable to move a file. Either the target data set is not available or is invalid. Provide a valid data set name in the configuration file.
SXF7182E	<%s> Error moving workfile to target file=%s	Unable to move a file. Either the target data set is not available or is invalid. Provide a valid data set name in the configuration file.

Number	Message	Description
SXF7183E	<%s> Failed to close .PRG file=%s, aborting genOutFile	Unable to close the progress file. Check the process directory library. The file might be corrupted.
SXF7184E	<%s> Failed to remove .PRG file=%s, aborting genOutFile	Unable to remove the progress file. Check the process directory library. The file might be corrupted or not properly cataloged.
SXF7185E	<%s> Failed to remove .PRG file=%s, aborting genOutFile	Unable to remove the progress file. Check the process directory library. The file might be corrupted or not properly cataloged.
SXF7186E	<%s> Type mismatch while retrieving value: %s, type: %d	The field type specified did not match the received data. Correct the configuration file and ensure that the configuration file matches the received message.
SXF7187E	<%s> Type mismatch while retrieving value: %s, type: %d	The field type specified did not match the received data. Correct the configuration file and ensure that the configuration file matches the received message.
SXF7188E	<%s> Type mismatch while retrieving value: %s, type: %d	The field type specified did not match the received data. Correct the configuration file and be certain that the configuration file matches the received message.
SXF7189E	<%s> Type mismatch while retrieving value: %s, type: %d	The field type specified did not match the received data. Correct the configuration file and ensure that the configuration file matches the received message.
SXF7190E	<%s> Type mismatch while retrieving value: %s, type: %d	The field type specified did not match the received data. Correct the configuration file and ensure that the configuration file matches the received message.
SXF7191E	<%s> Type mismatch while retrieving value: %s, type: %d	The field type specified did not match the received data. Correct the configuration file and ensure that the configuration file matches the received message.



Number	Message	Description
SXF7192E	<%s> Type mismatch while retrieving value: %s, type: %d	The field type specified did not match the received data. Correct the configuration file and ensure that the configuration file matches the received message.
SXF7193E	<%s> Type mismatch while retrieving value: %s, type: %d	The field type specified did not match the received data. Correct the configuration file and be certain that the configuration file matches the received message.
SXF7194E	<%s> Type mismatch while retrieving value: %s, type: %d	The field type specified did not match the received data. Correct the configuration file and ensure that the configuration file matches the received message.
SXF7195E	<%s> default Type mismatch while retrieving value: %s, pf_type: %d	The field type received did not match the expected data type that was configured. Correct the configuration file and ensure that the configuration file matches the received message.
SXF7196E	<%s> Failed to get RV field: cause=%s	Unable to extract a requested field from the message node. Check the configuration file for this message to ensure the fields in the message are correctly specified.
SXF7197E	Log Write <%s> failed: errno=%d last_op=%d ab_syscode=0x%X ab_rc=%d err text=%s	The write operation to the log file failed. This message gives the error codes and the text message of the error codes.
SXF7198E	<%s> LOG file <%s> reached B37/E37 out of space. Wrapping to Begin of file.	The log file is complete and the vsamUseLog parameter was not set. After this message, Adapter starts writing in the log from the beginning.
SXF7199E	<%s> Output file failed: errno=%d last_op=%d R15=%d Fdbk=%d ab_syscode=0x%X ab_rc=%d\n svc99 inf=%d err=%d err text=%s	The write operation to the target file failed. This message gives the error codes and the text message of the error codes.
SXF7200E	<%s> Text File Write failed: errno=%d err text=%s	The write operation to the TEXT file failed. This message gives the error codes and the text message of the error codes.

Number	Message	Description
SXF7201E	<%s> Binary File Write failed: errno=%d last_op=%d ab_syscode=0x%X ab_rc=%d\n err text=%s	The write operation to the BINARY file failed. This message gives the error codes and the text message of the error codes.
SXF7202E	VSAM key flocate failed for file %s. Record rejected R15=%d Fdbk=%d err=%s	Unable to find the record in the output file to update the record.
SXF7203E	VSAM key flocate failed because no such record (not found)	The subscriber received a record for a VSAM file operating in replace mode, and no corresponding record on the subscriber is available. The incoming record from the publisher is discarded. Verify that two VSAM systems are in sync with each other when using record level replacement.
SXF7204E	<%s> Failed to open Work file. cause=%d: %s	Unable to open a CWK work file. It usually indicates an out of space problem, or a RACF authorization problem. The cause contains the exact cause of the error.
SXF7205E	<%s> Failed to open Work file. cause=%d: %s	Unable to open a CWK work file. It usually indicates an out of space problem, or a RACF authorization problem. The cause contains the exact cause of the error.
SXF7206E	<%s> Failed to close .PRG file=%s, aborting recovery	A close on a Progress PRG file being used failed. Save the %s reason code and discuss the issue with your systems programmer.
SXF7207E	<%s> Failed to malloc TEXT LRECL buffer of length=%d	Unable to allocate memory. Report error to your TIBCO administrator.
SXF7208E	<%s> Unable to open .PRP file=%s, aborting recovery	Unable to open a CWK work file. It usually indicates an out of space problem, or a RACF authorization problem. The cause contains the exact cause of the error.
SXF7209E	<%s> Failed to read data in PRP file, aborting recovery	Unable to read data cached in the partial record file. Check the process directory. The file might be corrupted.

Number	Message	Description
SXF7210E	<%s> Failed to read data in progress file, aborting recovery	Unable to read the progress file. Check the process directory. The file might be corrupted.
SXF7211E	<%s> Failed to remove .PRG file=%s, aborting recovery	Unable to remove a file. Either the target data set is not available or is invalid. Report this error to your TIBCO administrator.
SXF7212E	<%s> Failed to recover Work file=<%s>. fopen with Append failed. Cause=%d: %s	Unable to recover a work file. The file might be corrupted. Contact the TIBCO administrator.
SXF7213E	<%s> Failed to allocate memory for TEXT LRECL buffer of length=%d	Unable to allocate memory. Report error to your TIBCO administrator.
SXF7214E	<%s> No hostname supplied on message with Security check	Security is turned on for Adapter, but an incoming message did not have a Publisher host name associated with it. The request is discarded. Verify that the Publisher is set up to automatically send the Publisher's host name to the Subscriber.
SXF7215E	<%s> No User Id supplied on message with Security check	Security is turned on for Adapter, but an incoming message did not have a User Id associated with it. The request is discarded. Verify that the Publisher is setup to automatically send the UserId to the Subscriber.
SXF7216E	<%s> Failed to extract checksum from end-of-file message	Invalid configuration was specified.
SXF7217E	Failed to write data from file <%s> - wrote %d out of %d	Could not add the data to the file, probably because of an out of space issue. Check the system log for related space problems.
SXF7218E	<%s> Failed to retrieve sequence/block_id number from ECM msg. cause=%s on Subject/Destination=<%s>. field=%s	Unable to retrieve a block number from a block transfer file. Ensure that the file being subscriber to is being published in block mode.

Number	Message	Description
SXF7219E	<%s> Failed to GUID from msg. cause=%s	Unable to retrieve unique GUID from a block transfer file. Ensure that the file being subscriber to is being published in block transfer mode.
SXF7220E	<%s> Failed to retrieve block DATA for block # %d for workfile. cause=%s	Unable to retrieve data element from a block transfer file. Ensure that the file being subscriber to is being published in block transfer mode.
SXF7221E	<%s> Write Failed for block num=%d on workfile. cause=%d: %s	Could not add the new block of data to the file, probably because of an out of space issue. Check the system log for related space problems.
SXF7222E	<%s> TEXT file write delimiter %s failed for workfile	Could not add the delimiter data to the file, probably because of an out of space issue. Check the system log for related space problems.
SXF7223E	<%s> Failed to retrieve field <%s> from message for workfile	Unable to retrieve field, possible configuration mismatch. Check the configuration file, and ensure that it matches the received message
SXF7224E	<%s> TEXT file write of delimited Value <%s> failed	Could not add the delimited value data to the file, probably because of an out of space issue. Check the system log for related space problems.
SXF7225E	<%s> TEXT file write of trail padding <%s> failed	Could not add the padding data to the file, probably because of an out of space issue. Check the system log for related space problems.
SXF7226E	<%s> Failed to retrieve field <%s> from message for workfile	Unable to retrieve field, possible configuration mismatch. Check the configuration file, and ensure that it matches the received message.
SXF7227E	<%s> TEXT file write fixed width Field <%s> failed	Could not add the value data to the file, probably because of an out of space issue. Check the system log for related space problems.

Number	Message	Description
SXF7228E	<%s> TEXT file write padded Field Value failed: <%s>	Could not add the value data to the file, probably because of an out of space issue. Check the system log for related space problems.
SXF7229E	<%s> TEXT file fflush failed	I/O error received for fflush operation. Check previous message in trace log from more details on the cause.
SXF7230E	<%s> Failed to retrieve embedded VSAM RRN field from message for file	Failed to retrieve field, possible configuration mismatch. Check the configuration file, and ensure that it matches a corresponding VSAM file at the publisher side.
SXF7231E	<%s> Failed to retrieve field <%s> from message for workfile	Failed to retrieve field, possible configuration mismatch on the message definition. Check the configuration file, and ensure that it matches the received message.
SXF7232E	<%s> Data type or Data content error on field <%s>	Data value retrieved from file is invalid. Possible configuration mismatch on the message definition.
SXF7233E	<%s> Failed to retrieve field <%s> from message for workfile	Failed to retrieve field, possible configuration mismatch. Check the configuration file, and ensure that it matches the received message.
SXF7234E	<%s> BINARY file write failed <%s>	The write operation to the BINARY file failed. This message gives the error codes and the text message of the error codes.
SXF7235E	<%s> BINARY file write failed details: errno=%d R15=%d Fdbk=%d op=%d ab_sys=0x%X ab_rc=0x%X\n errno text=%s	The write operation to the BINARY file failed. This message gives the error codes and the text message of the error codes.
SXF7236E	<%s> File write failed because duplicate record RRN or key already exists	The write operation to a VSAM RRDS file failed. This message gives the error codes and the text message of the error codes.
SXF7237E	<%s> Probably due to ALTINDEX Key mis-match (rcvd key != existing key)	The write operation to the VSAM KSDS file failed. This message gives the error codes and the text message of the error codes.

Number	Message	Description
SXF7238E	<%s> Write Failed for block num=%d on workfile for %d TEXT records of max %d bytes each. Error=%d: %s	The write operation to the TEXT file failed. This message gives the error codes and the text message of the error codes.
SXF7239E	<%s> Unable to open .PRP file=%s, data exposure if crash	The open for a TEXT file's partial data buffer file failed. This message gives the error codes and the text message of the error codes.
SXF7240E	<%s> Unable to write to .PRP file=%s, data exposure if crash	The write operation to the TEXT file partial data buffer file failed. This message gives the error codes and the text message of the error codes.
SXF7241E	<%s> Failed to initialize Confirm message	An attempt to create a RV or EMS message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF7242E	<%s> Failed to append Block Num to Confirm message	Unable to add the BlockNumber field to an ECM Confirm message. Make sure that TIBCO EMS/Rendezvous is available.
SXF7243E	<%s> Failed to append Status to Confirm message	Unable to add the Status field to an ECM Confirm message. Make sure that TIBCO EMS/Rendezvous is available.
SXF7244E	<%s> Failed to append Subscriber Name to Confirm message	Unable to add the Subscriber Name field to an ECM Confirm message. Make sure that TIBCO EMS/Rendezvous is available.
SXF7245E	<%s> Failed to append GUID to Confirm message	Unable to add the GUID field to an ECM Confirm message. Make sure that TIBCO EMS/Rendezvous is available.
SXF7246E	<%s> Failed to append Version to Confirm message	Unable to add the BlockNumber field to an ECM Confirm message. Make sure that TIBCO EMS/Rendezvous is available.

Number	Message	Description
SXF7247E	<%s> Failed to set RVCM send target: cause=%s Subject/Destination=%s	An error was returned when Adapter attempted to publish a message. Ensure that a valid subject/destination name is specified in the configuration file.
SXF7248E	<%s> Failed to set time limit for certified message: %s	Unable to set a time limit for a certified message. Check that TIBCO EMS/Rendezvous is available and operating properly.
SXF7249E	<%s> Error in sending RVCM message: cause=%s Subject/Destination=%s	An error occurred when sending an RVCM message. Adapter might not be active. Contact your TIBCO administrator.
SXF7250E	<%s> Failed to set send target: cause=%s Subject/Destination=%s	An error occurred when sending an RVCM message. Adapter might not be active. Contact your TIBCO administrator.
SXF7251E	<%s> Error in sending message: cause=%s Subject/Destination=%s	An error occurred when sending an RVCM message. Adapter might not be active. Contact your TIBCO administrator.
SXF7252E	<%s> Failed to initialize GenFile message: cause=%s	Unable to create a genFilePublish message. Ensure that TIBCO EMS/Rendezvous is available.
SXF7253E	<%s> Failed to add filename to genFilePublish message: cause=%s	Unable to add the filename field to a genFilePublish message. Check the configuration file to ensure that a valid file name was specified. Make sure that TIBCO EMS/Rendezvous is available.
SXF7254E	<%s> Failed to append num of lines to genFilePublish message: cause=%s	Unable to add the number of lines field to a genFilePublish message. Make sure that TIBCO EMS/Rendezvous is available.
SXF7255E	<%s> Failed to append num of messages to genFilePublish message: cause=%s	Unable to add the number of messages field to a genFilePublish message. Make sure that TIBCO EMS/Rendezvous is available.
SXF7256E	<%s> Failed to create Tracking message of length=%d: %s	Unable to create a message. Contact the TIBCO administrator to ensure that TIBCO EMS/Rendezvous is functioning.

Number	Message	Description
SXF7257E	<%s> Failed to add filename to genFilePublish message: cause=%s	An attempt to add a field to a RV or EMS message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF7258E	<%s> Type mismatch while retrieving value: %s, type: %d	The field type specified did not match the received data. Correct the configuration file, and ensure that the configuration file matches the received message.
SXF7259E	<%s> NULL string contents while retrieving value: %s, type: %d	The field type expected a STRING value but the received data contained a NULL instead. Correct the configuration file, and ensure that the configuration file matches the received message.
SXF7260E	<%s> Type mismatch while retrieving value: %s, type: %d	The specified field type did not match the received data. Correct the configuration file, and ensure that the configuration file matches the received message.
SXF7261E	<%s> Type mismatch while retrieving value: %s, type: %d	The field type specified did not match the received data. Correct the configuration file, and ensure that the configuration file matches the received message.
SXF7262E	<%s> Type mismatch while retrieving value: %s, type: %d	The field type specified did not match the received data. Correct the configuration file, and ensure that the configuration file matches the received message.
SXF7263E	<%s> Default Type mismatch while retrieving value: %s, from type=%d to type=%d	An attempt was made to retrieve a message, but the message type did not match the configuration file definition. Verify the type of field that was published.
SXF7264E	<%s> Type mismatch while retrieving value: %s, type: %d	The field type specified did not match the received data. Correct the configuration file, and ensure that the configuration file matches the received message.



Number	Message	Description
SXF7265E	<%s> Type mismatch while retrieving value: %s, type: %d	The field type specified did not match the received data. Correct the configuration file, and ensure that the configuration file matches the received message.
SXF7266E	<%s> Type mismatch while retrieving value: %s, type: %d	The field type specified did not match the received data. Correct the configuration file, and ensure that the configuration file matches the received message.
SXF7267E	<%s> Unsupported FLOAT conversion of type=%d for %s	The field type specified did not match the received data. Correct the configuration file, and ensure that the configuration file matches the received message.
SXF7268E	<%s> Invalid data type flag <%04x> for <%s>	The DataType flag was invalid. Ensure that the field type is valid.
SXF7269E	<%s> Unsupported PACKED conversion of type=%d for %s	The field type specified did not match the received data. Correct the configuration file, and ensure that the configuration file matches the received message.
SXF7270E	<%s> String <%s> length is %d. Must be less than 19.	A string field was longer than 19 bytes. If a string field is to be used for numeric conversions, it must be less than 19 bytes.
SXF7271E	<%s> Unsupported ZONED conversion of type=%d for %s	The field type specified did not match the received data. Correct the configuration file, and ensure that the configuration file matches the received message.
SXF7272E	<%s> Unsupported COMP/SHORT conversion of type=%d for %s	The field type specified did not match the received data. Correct the configuration file, and ensure that the configuration file matches the received message.
SXF7273E	<%s> Unsupported COMP/INTEGER conversion of type=%d for %s	The specified field type did not match the received data. Correct the configuration file, and ensure that the configuration file matches the received message.

Number	Message	Description
SXF7274E	<%s> Unsupported COMP/LOGLONG conversion of type=%d for %s	The field type specified did not match the received data. Correct the configuration file, and be certain that the configuration file matches the received message.
SXF7275E	<%s> Unsupported FLOAT conversion of type=%d for %s	The field type specified did not match the received data. Correct the configuration file, and be certain that the configuration file matches the received message.
SXF7276E	<%s> Unsupported DISPLAY/INTEGER conversion of type=%d for %s	The specified field type did not match the received data. Correct the configuration file, and ensure that the configuration file matches the received message.
SXF7277E	<%s> Unsupported DISPLAY/UNSIGNED INTEGER conversion of type=%d for %s	The specified field type did not match the received data. Correct the configuration file, and ensure that the configuration file matches the received message.
SXF7278E	<%s> Unsupported DISPLAY/SHORT conversion of type=%d for %s	The specified field type did not match the received data. Correct the configuration file, and ensure that the configuration file matches the received message.
SXF7279E	<%s> Unsupported DISPLAY/UNSIGNED SHORT conversion of type=%d for %s	The specified field type did not match the received data. Correct the configuration file, and ensure that the configuration file matches the received message.
SXF7280E	<%s> Unsupported DISPLAY/INTEGER conversion of type=%d for %s	The specified field type did not match the received data. Correct the configuration file, and ensure that the configuration file matches the received message.
SXF7281E	<%s> Error saving current status for workfile, into statfile=%s	A file I/O error occurred. If the problem persists, contact your systems programmer.
SXF7282E	<%s> Could not malloc I/O buffer for move. cause=%d: %s	Unable to allocate memory. Report error to your TIBCO administrator.
SXF7283E	<%s> Move failed: could not open workfile: cause=%d: %s	The file specified could not be opened. Be sure that FilePublisher has the proper authority.

Number	Message	Description
SXF7284E	<%s> Move failed: could not open target file <%s> cause=%d: %s	The specified file could not be opened. Make sure that FilePublisher has the proper authority.
SXF7285E	<%s> Move failed: could not close workfile: cause=%d: %s	The specified file could not be moved. Make sure that FilePublisher has the proper authority.
SXF7286E	%s	The %s identifies the problem found.
SXF7287E	%s	The %s identifies the problem found.
SXF7288E	<%s> Move failed: Unable to close target file <%s> cause=%d: %s	The specified file could not be moved. Make sure that FilePublisher has the proper authority.
SXF7289E	<%s> Move failed: could not close workfile: cause=%d: %s	The specified file could not be moved. Make sure that FilePublisher has the proper authority.
SXF7290E	<%s> Move failed: could not open workfile. cause=%d: %s	The open operation failed. Check that the target directory is available and that there are no authority problems.
SXF7291E	<%s> Move failed could not open target file <%s> using mode=%s Cause=%d:	The open operation failed. Check that the target directory is available and that there are no authority problems.
SXF7292E	<%s> Move failed: Unable to close workfile: cause=%d: %s	The close operation failed. Check that the target directory is available and that there are no authority problems.
SXF7293E	<%s> Move failed: could not fwrite to target file<%s> Cause=%d: %s	The write operation failed. Check that the target directory is available and that there are no authority problems, and there is sufficient space on the volume.
SXF7294E	<%s> Move failed: could not fprintf to target file <%s> cause=%d: %s	The write operation failed. Check that the target directory is available and that there are no authority problems, and there is sufficient space on the volume.

Number	Message	Description
SXF7295E	<%s> Move failed: could not write to target file <%s> Cause=%d: %s	The write operation failed. Check that the target directory is available and that there are no authority problems, and there is sufficient space on the volume.
SXF7296E	<%s> Move failed: Unable to close target file <%s> cause=%d: %s	The close operation failed. Check that the target directory is available and that there are no authority problems.
SXF7297E	<%s> Move failed: Unable to close workfile: cause=%d: %s	The close operation failed. Check that the target directory is available and that there are no authority problems.
SXF7298E	<%s> Error freeing GDG=<%s>: reason=%s	The GDG Deallocate operation failed. Check that the target directory is available and that there are no authority problems.
SXF7299E	<%s> Error moving working file to <%s>. Retry limit %d reached.	The target file was still locked out (in use by another job), after the max retry limit has been reached. The incoming file is converted to an ERR file, and trying to update the target file to be abandoned.
SXF7300E	<%s> Failed to retrieve sequence/block_id number from msg. cause=%s on Subject/Destination=%s	An attempt to retrieve a field from a RV message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF7301E	<%s>: Failed to extract Hostname from end-of-file message	An attempt to retrieve a field from a RV message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF7302E	<%s>: Failed to extract UserId from end-of-file message	An attempt to retrieve a field from an RV message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.

Number	Message	Description
SXF7303E	<%s> Error saving STERR num error status file=%s	A file write failed. It is usually because of an out of space error. See the previous error message in the trace log for the exact cause.
SXF7304E	<%s> VSAM fupdate failed for file. EOF returned.	A VSAM file update failed. It is usually because of a B37/D37 out of space error. See the previous error message in the trace log for the exact cause.
SXF7305E	<%s> VSAM fread for update failed for file.\n R15=%d Fdbk=%d op=%d ab_sys=0x%X ab_rc=0x%X\n errno=%d errno text=%s	The read of a VSAM file failed. The cause contains the reason for the failure.
SXF7306E	<%s> VSAM update failed for file. R15=%d Fdbk=%d op=%d ab_sys=0x%X ab_rc=0x%X\n errno=%d errno text=%s	A VSAM file update failed. It is usually because of a B37/D37 out of space error. The cause contains the reason for the failure.
SXF7307E	<%s> User=<%s> (Pub Id=%s) is not allowed WRITE access to target File=<%s>. Rejected. rc=%d subcode=0x%x rq=%d VOLSER=<%s> %s	Security checking was requested, and the security facility indicated the incoming File Transfer request from the specified user, was not allowed access to publish that file. Contact your security people, to determine why the user is not authorized to write to that file.
SXF7309E	<%s> Recovery for GDG file failed probably because Ops deleted CWK work file=%s. Progress file is being reset and recovery for that file has been abandoned. cause=%d: %s	Unable to recover a work file being used to update a GDG file. The file might be corrupted or might have been deleted. Contact the TIBCO administrator.
SXF7310E	<%s> Recovery for SEQ file failed probably because Ops deleted CWK work file=%s. Progress file is being reset and recovery for that file has been abandoned. cause=%d: %s	Unable to recover a work file being used to update a Sequential file. The file might be corrupted or might have been deleted. Contact the TIBCO administrator.
SXF7313W	FileType=<%s> Resetting PRG file %s to #NONE#	Denotes that a file transfer is complete, and that the file type is being reset.

Number	Message	Description
SXF7315E	<%s> There is an error in the configuration for this FileType. Cannot create the subscriber %s	Unable to create a listener for a subscriber because of errors previously reported in the configuration processing phase. Check the trace log to determine the problem in the configuration file and correct it.
SXF7316E	<%s> UserId=<%s> from node=%s was not found in Security Mapping Table. Request to update Subscriber file was rejected. Block# %d	Security checking was requested, and the security facility indicated the incoming File Transfer request from the specified user, was not allowed access to publish that file. Contact your security people, to determine why the user is not authorized to write to that file.
SXF7317W	FileType=<%s> Deleting PRG file	Denotes that a file transfer is complete, and that the PRG file for that file type is being deleted.
SXF7318W	File Type <%s> Error encountered. Closing all .CWK, .PRG, and .PRP files	A severe I/O error occurred when processing a incoming file transfer. See the previous message in the trace log for the cause of the problem
SXF7319E	<%s> Rename of workfile %s to .ERR error file %s failed for %.16s. Reason=%d: %s	The rename operation failed. Check that the target directory is available and that there are no authority problems.
SXF7320E	<%s> Got I/O error on .PRG file=%s. Reason=%d: %s	A file write failed on a PRG progress file. It is usually because of an out of space error. See the previous error message in the trace log for the exact cause.
SXF7324E	<%s> Error removing GDG file=<%s>. Reason=%d: %s	Unable to remove a file. Either the target data set is not available or it is invalid. Provide a valid data set name in the configuration file.
SXF7328E	<%s> Failed to retrieve sequence/block_id number from Record ECM msg. cause=%s on Subject/Destination=<%s>. useNT=%d	An attempt to retrieve a field from a RV message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.

Number	Message	Description
SXF7329E	<%s> Rename of workfile %s to target file %s failed. Reason=%d: %s	The specified file could not be renamed. Be sure that FilePublisher has the proper authority.
SXF7331E	<%s> User=<%s> (Pub Id %s) is not allowed WRITE access to target File=<%s>. Rejected. RC=%d racf_rc=%d subcode=0x%x VOLSER=<%s> %s	A file is being written and an error was flagged (for example, the FileAdapter user does not have ALTER/Update authority to that dataset file.). Contact your systems programmer.
SXF7332E	<%s> User=<%s> (Pub Id %s) is not allowed ALTER access to work file=<%s>. Rejected. RC=%d racf_rc=%d subcode=0x%x VOLSER=<%s> %s	A file is being written and an error was flagged (for example, the FileAdapter user does not have ALTER/Update authority to that dataset file.). Contact your systems programmer.
SXF7334E	<%s> Unable to write to target=<%s> and 'File Locked' retry timer failed. cause=%d: %s	A file was locked and the File Adapter was unable to create a retry timer. This usually indicates an out of memory condition, or a failure/crash in the underlying RV support.
SXF7337E	<%s> File open failed because unable to allocate dataset file. Dataset File is unavailable and/or allocated to another job. SVC99 codes=%d %d	The file is not available. The SVC99 codes contains the reason for the failure.
SXF7340E	<%s> Error writing CWK workfile. GDG target file=<%s> not created. file_rc=%d	A file write failed. It is usually because of a B37/D37 out of space error. The file_rc code gives the cause of the error.
SXF7349E	Error from the Svc99 function <%s>. File=<%s>	An error occurred when dynamically opening a file. The first %s denotes what the error was. Save this message and consult with your MVS programmer.
SXF7350E	Svc99 error: Duplicate Dataset Name=%s already exists on the system	This normally indicates that two different field types are trying to use the same file. This is not allowed.
SXF7351E	Svc99 error: Insufficient space on requested Volumes	A file is being created on a VOLUME that has run out of space.

Number	Message	Description
SXF7352E	Svc99 error: RACF/ACF2 Security problem	A file is being created and RACF has flagged an error (e.g. the Volume is password protected).
SXF7353E	Svc99 error: Cannot put a non-SMS dataset on an SMS volume	A VOLSER has been specified for a file that exists on a SMS managed volume
SXF7354E	Svc99 error: Catalog error - program not authorized to perform operation	A file is being created and RACF has flagged an error (e.g. the Volume is password protected).
SXF7355E	Svc99 error: RACF/ACF2 Security problem (Not Auth) or SMS error	A file is being created and RACF has flagged an error (e.g. the Volume is password protected).
SXF7356E	Svc99 error: File=<%s> already allocated and the tasks' usage attributes of 'a' and 'w' conflict	This normally indicates that two different field types are trying to use the same file. This is not allowed.
SXF7357E	Output File <%s> could not be opened using DD=<%s> mode=<%s> Type_alloc=%s errno=%d %s	The specified file could not be opened. The errno %s specifies the specific cause of the problem.
SXF7358E	Append Output File <%s> might not be currently cataloged on VOLSER=<%s ...>	The VOLSER parameter specified in the INI file might be incorrect for that file. Contact your systems programmer.
SXF7360E	Svc99 error: Invalid file name %s was rejected by Allocate	The FILENAME parameter specified in the INI file were incorrect. Contact your MVS systems programmer.
SXF7361E	Svc99 error: Invalid VolSer list was rejected by Allocate	The VOLSER parameter specified in the INI file were incorrect. Contact your MVS systems programmer.
SXF7362E	Svc99 error: Invalid UNIT name %s was rejected by Allocate	The UNIT parameter specified in the INI file were incorrect. Contact your MVS systems programmer.



Number	Message	Description
SXF7363E	Svc99 error: Invalid length was rejected by Allocate. Probably due to incorrect length on File name, Volser, or Unit	The FILENAME, UNIT, or VOLSER parameters specified in the INI file were incorrect. Contact your MVS systems programmer.
SXF7364E	Svc99 error: UNIT %s and Volser %s are incompatible or missing and were rejected by Allocate	The UNIT and VOLSER parameters specified in the INI file were incompatible for the file being processed. Contact your MVS systems programmer.
SXF7365E	Output File <%s> could not be opened using mode=<%s> errno=%d %s	A file open failed. The errno contains the cause of the error.
SXF7367E	<%s> Failed to create nested EPM Payload message of length=%d: %s	An attempt to create a RV or EMS message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF7368E	<%s> Unable to read source CWK file <%s> cause=%d: %s	The read operation failed. Check that the CWK workfile directory is available and that there are no authority problems.
SXF7369E	<%s> Unable to read target file <%s> cause=%d: %s	A file read failed. The errno contains the cause of the error.
SXF7370E	<%s> Subscriber I/O error caused File Transfer Abort: %d %s	A file write failed. It is usually because of an out of space error. The Abort code contains the cause of the error. The File Transfer is aborted.
SXF7371E	<%s> ConfBlock - Failed to create message of length=%d: %s	An attempt to create a RV or EMS message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.

Number	Message	Description
SXF7374E	<%s> File open, rename, or append write failed because unable to allocate dataset file. Dataset File is unavailable and/or allocated to another job. SVC99 codes=%d %d op=%d	A file open or rename failed. The SVC99 codes contains the cause of the error.
SXF7375W	<%s> Discarding Block # %d because of previous ECM I/O Error.	A new block was received after an I/O error was signalled to the Publisher. The block is discarded.
SXF7376E	<%s> Failed to close .PRG file=%s, aborting renameByCopy. error=%d: %s	A file close failed for a PRG progress file. The errno contains the cause of the error. The file transfer is aborted.
SXF7377E	<%s> Failed to open Progress file during Lock Retry: <%s> errno=%d errtext=%s	A file open failed during a locked file retry. The errno contains the cause of the error.
SXF7379E	<%s> There is an error in the configuration for this FileType. Will NOT perform the NT ECM Admin handshake for subscriber=%s ----	Check configuration phase error messages from FileAdapter startup and correct the error.
SXF7391E	<%s> <%s> Target file=<%s> not cataloged. Incoming file discarded/rejected.	File is not cataloged - reject the new incoming file.
SXF7392E	<%s> Error removing/deleting %s file=<%s>. Reason=%d: %s	Deletion of CWK file was unsuccessful.
SXF7394E	<%s> Svc99 error: Unable to open (for write) existing Dataset=%s. File may be use by someone else.	Unable to open dataset. Duplicate name on same volume.

## SXF8000 – SXF8999

Number	Message	Description
SXF8001E	FileType=<%s>: Subject=<%s> Unable to extract reply_subject back to Publisher	This is a Block Mode ECM internal or recovery type error. Contact TIBCO technical support.
SXF8007E	FileType=<%s>: Subject=<%s> Publisher indicates we are not registered for ECM mode. Flipping out to non-ECM mode.	This normally indicates a configuration mis-match between the Publisher and the Subscriber.
SXF8008W	FileType=<%s>: Subject=<%s> Publisher rejected our VERIFY_IFACTIVE_REQUEST - we are not registered. Issuing SUB_ACTIVATION_REQUEST	This indicates either that the Progress file was deleted at the Publisher, or a configuration mis-match between Publisher and Subscriber.
SXF8010E	FileType=<%s>: Subject=<%s> Publisher rejected our SUB_ACTIVATION_REQUEST	This indicates either that the Progress file was deleted at the Publisher, or a configuration mis-match between Publisher and Subscriber.
SXF8015W	FileType=<%s>: Subject=<%s> Publishing ECM VERIFY_IFREGISTERED_ REQUEST handshake message with SubscriberName=%s using GUID=%s retry#%d	This indicates that a restart/recovery sequence was performed between Publisher and Subscriber.
SXF8018E	<%s> Listener setup Failed for Admin Request subject <%s> cause=%s	An attempt to create a RV or EMS listener failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF8019E	<%s> Listener setup Failed for Admin Reply subject <%s> cause=%s	An attempt to create a RV or EMS listener failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.

Number	Message	Description
SXF8020E	NULL FileHandler in Admin timer callback	This is a Block Mode ECM internal or recovery type error. Contact TIBCO technical support.
SXF8021E	<%s>: Error on sending RV Admin message: cause=%s subject=%s	Unable to publish an ECM Admin message. Check that TIBCO EMS/Rendezvous is available.
SXF8022E	<%s>: Error on sending RV Admin message: cause=%s subject=%s	Unable to publish an ECM Admin message. Check that TIBCO EMS/Rendezvous is available.
SXF8023E	<%s>: Error on sending RV Admin message: cause=%s subject=%s	Unable to publish an ECM Admin message. Check that TIBCO EMS/Rendezvous is available.
SXF8024E	<%s>: Error on sending RV Admin message: cause=%s subject=%s	Unable to publish an ECM Admin message. Check that TIBCO EMS/Rendezvous is available.
SXF8025E	<%s>: Error on sending RV Admin message: cause=%s subject=%s	Unable to publish an ECM Admin message. Check that TIBCO EMS/Rendezvous is available.
SXF8026E	<%s>: Error on sending RV Admin message: cause=%s subject=%s	Unable to publish an ECM Admin message. Check that TIBCO EMS/Rendezvous is available.
SXF8027E	<%s>: Error on sending RV Admin message: cause=%s subject=%s	Unable to publish an ECM Admin message. Check that TIBCO EMS/Rendezvous is available.
SXF8028E	<%s> Admin Timer creation failed for Subject <%s>: cause=%s	An attempt to create a RV or EMS timer failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF8029E	FileType=<%s>: Received unknown ECM Admin config reply. msgtype=%d	This is a Block Mode ECM internal or recovery type error. Contact TIBCO technical support.

Number	Message	Description
SXF8030E	FileType=<%s>: SUB_IS_REG_REPLY = False. Startup Handshake rejected	This normally indicates a configuration mis-match between the Publisher and the Subscriber.
SXF8031E	FileType=<%s>: Received unknown ECM Admin config reply. msgtype=%d	This is a Block Mode ECM internal or recovery type error. Contact TIBCO technical support.
SXF8032E	<%s> Failed to create RV Admin message: %s	An attempt to create a RV or EMS message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF8033E	<%s> Failed to append MsgType to Admin message: %s	An attempt to add a new field to a RV or EMS message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF8034E	<%s> Failed to append MsgId/Guid to Admin message: %s	An attempt to add a new field to a RV or EMS message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF8035E	<%s> Failed to append Version to Admin message: %s	An attempt to add a new field to a RV or EMS message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF8036E	<%s> Failed to append bool flag to Admin message: %s	An attempt to add a new field to a RV or EMS message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.

Number	Message	Description
SXF8037E	<%s> Failed to append NumRetries to Admin message: %s	An attempt to add a new field to a RV or EMS message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF8038E	<%s> Failed to set RV Admin send subject: %s	An attempt to add a new field to a RV or EMS message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF8039E	<%s> Failed to set RV Admin reply subject: %s	An attempt to add a new field to a RV or EMS message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO tech support.
SXF8040E	<%s> Failed to send RV Admin msg: cause=%s subject=%s	Unable to publish an ECM Admin message. Check that TIBCO EMS/Rendezvous is available.
SXF8041E	<%s> Failed to create RV Admin SUB_ACTIVATION message: %s	An attempt to create a RV or EMS message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF8042E	<%s> Failed to append MsgType to Admin message: %s	An attempt to add a new field to a RV or EMS message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.

Number	Message	Description
SXF8043E	<%s> Failed to append MsgId/Guid to Admin message: %s	An attempt to add a new field to a RV or EMS message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF8044E	<%s> Failed to append SubscriberName to Admin message: %s	An attempt to add a new field to a RV or EMS message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF8045E	<%s> Failed to append Version to Admin message: %s	An attempt to add a new field to a RV or EMS message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF8046E	<%s> Failed to set RV Admin send subject: cause=%s subject=%s	An attempt to add a new field to a RV or EMS message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF8047E	<%s> Failed to set RV reply subject: cause=%s subject=%s	An attempt to add a new field to a RV or EMS message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF8048E	<%s> Failed to send RV Admin msg: cause=%s subject=%s	Unable to publish an ECM Admin message. Check that TIBCO EMS/Rendezvous is available.

Number	Message	Description
SXF8049E	<%s> Failed to create RV Admin REG_SUB_ALIVE_REPLY message: %s	An attempt to create a RV or EMS message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF8050E	<%s> Failed to append MsgType to Admin message: %s	An attempt to add a new field to a RV or EMS message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF8051E	<%s> Failed to append MsgId/Guid to Admin message: %s	An attempt to add a new field to a RV or EMS message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF8052E	<%s> Failed to append SubscriberName to Admin message: %s	An attempt to add a new field to a RV or EMS message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF8053E	<%s> Failed to append Version to Admin message: %s	An attempt to add a new field to a RV or EMS message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF8054E	<%s> Failed to set RV Admin send subject: cause=%s subject=%s	An attempt to add a new field to a RV or EMS message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.



Number	Message	Description
SXF8055E	<%s> Failed to set RV reply subject: cause=%s subject=%s	An attempt to add a new field to a RV or EMS message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF8056E	<%s> Failed to send RV Admin msg: cause=%s subject=%s	Unable to publish an ECM Admin message. Check that TIBCO EMS/Rendezvous is available.
SXF8057E	<%s> Failed to create RV Admin VERIFY_IFACT message: %s	An attempt to create a RV or EMS message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF8058E	<%s> Failed to append MsgType to Admin message: %s	An attempt to add a new field to a RV or EMS message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF8059E	<%s> Failed to append MsgId/Guid to Admin message: %s	An attempt to add a new field to a RV or EMS message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF8060E	<%s> Failed to append SubscriberName to Admin message: %s	An attempt to add a new field to a RV or EMS message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.

Number	Message	Description
SXF8061E	<%s> Failed to append Version to Admin message: %s	An attempt to add a new field to a RV or EMS message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF8062E	<%s> Failed to set RV Admin send subject: cause=%s subject=%s	An attempt to add a new field to a RV or EMS message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF8063E	<%s> Failed to set RV reply subject: cause=%s subject=%s	An attempt to add a new field to a RV or EMS message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF8064E	<%s> Failed to send RV Admin msg: cause=%s subject=%s	Unable to publish an ECM Admin message. Check that TIBCO EMS/Rendezvous is available.
SXF8065E	<%s> Failed to create RV Admin VERIFY_IFREG message: %s	An attempt to create a RV or EMS message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF8066E	<%s> Failed to append MsgType to Admin message: %s	An attempt to add a new field to a RV or EMS message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.

Number	Message	Description
SXF8067E	<%s> Failed to append MsgId/Guid to Admin message: %s	An attempt to add a new field to a RV or EMS message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF8068E	<%s> Failed to append SubscriberName to Admin message: %s	An attempt to add a new field to a RV or EMS message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF8069E	<%s> Failed to append Version to Admin message: %s	An attempt to add a new field to a RV or EMS message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF8070E	<%s> Failed to set RV Admin send subject: cause=%s subject=%s	An attempt to add a new field to a RV or EMS message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF8071E	<%s> Failed to set RV reply subject: cause=%s subject=%s	An attempt to add a new field to a RV or EMS message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF8072E	<%s> Failed to send RV Admin msg: cause=%s subject=%s	Unable to publish an ECM Admin message. Check that TIBCO EMS/Rendezvous is available.

Number	Message	Description
SXF8073E	<%s> Failed to create RV Admin VERIFY_IFACTIVE_REPLY message: %s	An attempt to create an RV or EMS message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF8074E	<%s> Failed to append MsgType to Admin message: %s	An attempt to add a new field to a RV or EMS message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF8075E	<%s> Failed to append Version to Admin message: %s	An attempt to add a new field to a RV or EMS message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF8076E	<%s> Failed to append 'isactive' to Admin message: %s	An attempt to add a new field to a RV or EMS message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF8077E	<%s> Failed to set RV Admin send subject: cause=%s subject=%s	An attempt to add a new field to a RV or EMS message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF8078E	<%s> Failed to send RV Admin msg: cause=%s subject=%s	Unable to publish an ECM Admin message. Check that TIBCO EMS/Rendezvous is available.

Number	Message	Description
SXF8079E	<%s> Failed to create RV Admin SUB_ACTIVATION_REPLY message: %s	An attempt to create a RV or EMS message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF8080E	<%s> Failed to append MsgType to Admin message: %s	An attempt to add a new field to a RV or EMS message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF8081E	<%s> Failed to append Version to Admin message: %s	An attempt to add a new field to a RV or EMS message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF8082E	<%s> Failed to append 'isactivated' to Admin message: %s	An attempt to add a new field to a RV or EMS message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF8083E	<%s> Failed to set RV Admin send subject: cause=%s subject=%s	An attempt to add a new field to a RV or EMS message failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF8084E	<%s> Failed to send RV Admin msg: cause=%s subject=%s	Unable to publish an ECM Admin message. Check that TIBCO EMS/Rendezvous is available.

**SXF9000 – SXF9999**

Number	Message	Description
SXF9002W	Library not found: %s	A library needed by the FileAdapter was not found.
SXF9003E	API %s failed with exception %s	A TIBCO API failed. Note the return code and contact your TIBCO administrator.
SXF9004E	Input file <%s> does not exist	The input file specified did not exist. Correct the file name, or the library name.
SXF9005E	File <%s> is already present in process library	The specified file already existed in the process directory library. Move the specified file out of the process directory library to prevent Adapter from overwriting the file.
SXF9006E	Library not found: %s	Could not find the specified library. Supply a name of a library that exists on the system.
SXF9007E	Input file <%s> does not exist	The input file specified did not exist. Correct the file name, or the library name.
SXF9008E	File <%s> is present in process library.	The specified file already existed in the process directory library. Move the specified file out of the process directory library to prevent Adapter from overwriting the file.
SXF9502E	Generic error: %s %d %d %d %d	File Adapter is reporting an error it encountered. The %s details the cause of the error.
SXF9503E	Generic error: %s %s %d %d %d	File Adapter is reporting an error it encountered. The %s details the cause of the error.
SXF9504E	NULL SaveTimer in timer callback	An internal FileAdapter error occurred. Contact TIBCO technical support.

Number	Message	Description
SXF9505E	SaveTimer Timer creation failed: cause=%s	An attempt to create a RV or EMS timer failed. The %s identifies the specific cause. It normally indicates insufficient memory for the REGION. If increasing the REGION size does not solve the problem, then contact TIBCO technical support.
SXF9506E	SaveTimer Timer removal failed: cause=%s	An attempt to free a RV or EMS timer failed. If the problem persists, then contact TIBCO technical support.
SXF9507E	Error renaming Sub log file <%s> to <%s> cause: %d %s. File logging turned off...	The File Adapter was unable to rename and open a new trace log file. The second %s details the cause of the problem.
SXF9508E	Failed to open Sub log file <%s> cause: %d %s. File logging turned off...	The File Adapter was unable to open the trace log file. The second %s details the cause of the problem.
SXF9509E	Failed to open Sub log file <%s> cause: %d %s. File logging turned off...	The File Adapter was unable to open the trace log file. The second %s details the cause of the problem.





# Index

## Symbols

.LOG files [50, 82](#)

## A

adapter

- function overview [2](#)
- relationship to IBM AS/400 [2](#)
- typical usage scenario [2](#)
- usage scenarios [2](#)

ADAPTER\_NAME element [51, 83](#)

AE wire format [3](#)

AppendDateTime parameter [95](#)

appendDateTime parameter [120](#)

appendToExistingFile parameter [97](#)

autoGenerateFile parameter [96, 120](#)

## B

binary datatypes [131](#)

binary format [14](#)

block transfer mode [120](#)

blocksizeAlloc parameter [94](#)

blockTransferMode parameter [62, 101, 120](#)

blockTransferSize parameter [62, 120](#)

BusinessEvents

- in introduction [15](#)
- subject element for subscriber [83](#)

## C

certified messages delivery [132](#)

checking trace logs [38](#)

CL program importance [125](#)

COBOL numeric data types [133](#)

COBOL numeric datatypes [66](#)

COBOL, emulating LOW-VALUES and  
HIGH-VALUES [67](#)

codepage support [53](#)

commands

- call [125](#)
- DLT [40](#)
- SBMJOB for FilePublisher [116](#)
- SBMJOB for FileSubscriber [117](#)

configuration file examples

- delimited file [70, 106](#)
- different order formats [72, 108](#)
- fixed-length file [71, 107](#)

configuration files

- examples [44](#)

configuration information, printing at startup [54, 85](#)

configuring data sets [56](#)

confirmationSubject parameter [63, 102](#)

constant tag [67](#)

constraint parameter [65, 65, 103](#)

- tags [68](#)

constructing subject name from data [131](#)

containerName tag [68, 105](#)

CONTINUE\_ON\_CONFIG\_ERROR element [51, 83](#)

convertTo tag [104](#)

convertToString tag [66](#)

customer support [xiii](#)

## D

daemon parameter [86](#)

data blocks, limiting the number of [86](#)

data sets

- configuration [56](#)

- dataSetType parameter [63](#)
- datasetType parameter [93](#)
- defaultTimeLimit parameter [87](#)
- DELETE\_PUB\_PRG\_FILES element [51](#)
- DELETE\_SUB\_PRG\_FILES element [83](#)
- delimited file
  - configuration file examples [70](#), [106](#)
  - example [70](#)
- delimiter parameter [64](#), [101](#)
- delivery options [3](#)
- deliveryMode parameter [58](#), [58](#)
- description tag [103](#)
- different order formats [72](#), [108](#)
- double values
  - publishing [132](#)

## E

- ECM [10](#)
  - and RVCN [11](#)
  - error/restart handling [121](#)
  - FilePublisher [132](#)
  - RV subjects [122](#)
- ECM\_NESTED\_TEXT\_LINES element [52](#)
- ECMSubscriberName parameter [63](#), [101](#), [123](#)
- EEM\_DESTINATION element [52](#), [83](#)
- EEM\_SUBJECT element [52](#), [83](#)
- EEM, support for [15](#)
- EMS\_SESSION element [52](#), [55](#), [83](#), [84](#)
- endPublishDestination [61](#)
- endPublishSubject parameter [61](#), [120](#)
- EOL\_CHARACTER element [56](#)
- EOL\_ON\_SUBSCRIPTION element [84](#)
- ERROR\_DESTINATION element [52](#), [53](#), [84](#)
- ERROR\_EXIT\_CC element [52](#), [84](#)
- ERROR\_SUBJECT element [52](#), [84](#)
- errors, writing to SYSLOG [91](#)

## examples

- configuration file for a delimited file [70](#)
- configuration file for a fixed-length file [71](#)
- configuration file for different order formats [72](#)
- failed FilePublisher session [142](#)
- failed FileSubscriber session [151](#)
- FilePublisher [70](#)
- FilePublisher configuration file for numeric data
  - types [74](#)
- FileSubscriber configuration file supporting
  - numeric data types [109](#)
- header record in an output file [134](#)
- pre-processing [125](#)
- SBMJOB for starting [117](#)
- starting with sbmjob [116](#)
- successful FilePublisher session [138](#)
- successful FileSubscriber session [145](#)
- executeAfterProcess parameter [64](#), [100](#)
- executeBeforeProcess parameter [64](#), [100](#)
- exitOnFileSaveError parameter [97](#)
- explicit confirmation mode (ECM) [10](#)

## F

- field parameter [103](#)
- fieldStart tag [67](#), [104](#)
- file transfer
  - ECM [15](#)
- FILE\_COUNT element [47](#), [79](#)
- FILE\_LIMIT element [47](#), [79](#)
- FILE\_LINE element [103](#)
- FILE\_NAME element [47](#), [79](#)
- FILE\_OPTIONS element [57](#), [92](#), [133](#)
- fileHeader parameter [100](#), [134](#)
  - example [134](#)
- filePrefix parameter [57](#), [92](#)

FilePublisher  
 configuration file 47  
 ECM 132  
 example 74  
 example of failed session 142  
 example of successful session 138  
 examples 70  
 numeric data types 74  
 overview 5, 5  
 polling method 6  
 post-processing of files 126  
 pre-processing of files 125  
 sample configuration files 45  
 starting 116, 116  
 stopping 118  
 usage guidelines 70

FileSubscriber 8  
 configuration file 79  
 example for COBOL numeric data types 109  
 example of failed session 151  
 example of successful session 145  
 examples 106  
 numeric data types example 109  
 post-processing of files 126  
 pre-processing of files 125  
 sample configuration files 45  
 starting 117, 117  
 stopping 119  
 usage guidelines 133

fileTrailer parameter 100, 134

FileType section 57, 92

fixed-length file  
 configuration file examples 71, 107  
 example 71

floating-point datatypes 131

forcePublishedFileName parameter 98

fromMessage tag 104

functional components 4

## G

GDG  
 output files, storing 90

generateFileDestinationName parameter 99  
 generateFileFieldName parameter 61, 99  
 generateFileOnNumberOfMessages parameter 96  
 generateFileOnNumMsgs parameter 120  
 generateFileSubjectName parameter 99, 120  
 genFilePublishDestinationName parameter 98  
 genFilePublishSubject parameter 98

grouping  
 fields 68  
 progress (.PRG) files 85  
 work (.CWK) files 91

## H

header record in an output file 133

heartbeat messages 127  
 format 127

HEARTBEAT\_FILE\_INFO element 53, 84

HEARTBEAT\_TIME element 53, 84, 127

HIGH-VALUES (COBOL) 67

HOST\_CODEPAGE element 53, 85

## I

info messages 160

INPUT\_DATASET element 53

inputDataset parameter 63

installation  
 requirements 22

isBinary parameter 65, 101

isCertified parameter 64, 99

## J

JCL consideration 79

JMS\_TIBCO\_MSG\_TRACE parameter 58, 58, 92

**K**

keepTrailingBlanks parameter [64](#)  
 KILL\_PUBLISHER element [53](#)

**L**

label tag [66](#)  
 ledgerFile parameter [87](#)  
 length tag [67](#), [68](#), [104](#)  
 lineLength parameter [63](#), [94](#)  
 lineLength tag [105](#)  
 listenerName parameter [56](#)  
 listeningSubject  
   parameter [56](#)  
 location tag [67](#), [69](#)  
 logMatched parameter [65](#)  
 logs  
   trace [38](#)  
 logUnmatched parameter [65](#)  
 LOW-VALUES (COBOL) [67](#)

**M**

mapping user ids [88](#)  
 MAX\_CONCURRENT\_JOBS element [53](#)  
 memory  
   preventing exhaustion [86](#)  
 MESSAGE\_FIELDS element [65](#)  
 messageContainer parameter [65](#)  
   tags [68](#)  
 messageItem parameter [65](#)  
 messagesPerTransaction parameter [60](#)  
 mode processing, recording [88](#)  
 multiple record formats [68](#)  
 MVS volume [50](#), [82](#), [90](#), [90](#), [90](#), [91](#)

**N**

name parameter [86](#)  
 name tag [68](#)  
 nested message in a container [68](#)  
 network parameter [86](#)  
 NETWORK\_CODEPAGE element [54](#), [85](#)  
 noOfRetries parameter [96](#)  
 noWaitAfterConfirmations [122](#)  
 noWaitAfterConfirmations parameter [63](#)  
 numeric data types [133](#)

**O**

opaque data [131](#)  
 opaque data types [131](#)  
 operating system supported [22](#)  
 Options section [51](#), [82](#)  
 order formats [72](#)  
   configuration file example [108](#)  
   configuration file examples [72](#)  
   example [72](#)  
 output file  
   creation [9](#)  
   format [9](#)  
   header and trailer [133](#)  
 output file creation [96](#)  
 OUTPUT\_DATASET element [54](#), [85](#)  
 outputDataset parameter [63](#), [94](#)

**P**

packed datatypes [131](#)  
 packed format [14](#)  
   formula for [14](#)  
 padCharacter parameter [100](#)  
 padCharacter tag [104](#)  
 padDirection parameter [100](#)  
 padDirection tag [104](#)

## parameters

- alwmltthd 117, 117, 118
- executeAfterProcess 125
- executeBeforeProcess 125
- iscertified 99
- jobmsgqfl 116, 117, 118
- PERSIST\_EMS\_ECM\_MSGS element 54
- polling method 6
- pollInterval parameter 60
- position tag 67, 68, 104
- precision 13
- precision tag 66, 104
- Pre-Register section 56
- pre-registering subscribers 132
- primaryAlloc parameter 94
- PRINT\_FILE\_OPTIONS element 54, 85
- PRINT\_STDOUT element 47, 79
- PROCESS\_DATASET element 54
- processDataset parameter 63
- progress (.PRG) files
  - storing 91
- progress (.PRG) files, grouping 85
- PROGRESS\_DATASET element 85
- PUBLISH\_HEARTBEAT element 54, 85, 127
- publishDestinationName parameter 57, 58
- publishDestinationType parameter 57, 58
- publishing double values 132
- publishSubjectName parameter 57

**Q**

- QUEUE\_LIMIT element, subscriber 86
- queueing of files 5

**R**

- receiving COBOL numeric data types 133
- removeAfterProcess parameter 64
- removeLeadingBlanks parameter 64
- removeTrailingBlanks parameter 64
- requireOldMessage parameter 56

- requireOldMessages parameter 87, 132
- retransmissionDelayTicks 122
- retransmissionDelayTicks parameter 62, 102
- retryInterval parameter 97
- RV vs RVCN 3
- RV\_SESSION element 54, 86
- RVCN
  - vs. ECM 11
- RVCN\_SESSION element 55, 87
- rvcmTimeLimit parameter 64

**S**

- sample configuration files 44
- sample data files 46, 46
- sample files for configuration 44
- saveFileInterval parameter 96
- secondaryAlloc parameter 94
- SECURITY\_CHECK\_FILE element 88
- sending COBOL numeric data types 130
- sending data untranslated (opaque) 131
- SEQ (sequential)
  - files, storing 90
- service parameter 86
- session type 51, 51
- skipPadding parameter 100
- software requirements 22
- startAtLine parameter 65
- starting FilePublisher 116
- starting FileSubscriber 117
- startNewMessage tag 68
- startPublishDestination parameter 61
- startPublishSubject parameter 60
- stopping FilePublisher 118
- stopping FileSubscriber 119
- subject name
  - constructing 131
- subscribeDestinationType parameter 92
- subscribeDestinationName parameter 92
- subscriber
  - pre-registering 132
- SUBSCRIBER element 56
- subscribeSubjectName parameter 92

support, contacting [xiii](#)  
 supported data types [42](#)  
 supported file types [42](#)  
 syncLedger parameter [87](#)  
 SYSLOG, writing erros to [91](#)

## T

tags  
   in the constraint parameter [68](#)  
   in the messageContainer parameter [68](#)  
   in the messageitem parameter [66](#)  
   precision [13](#)  
 technical support [xiii](#)  
 TERMINATE\_ON\_RV\_SEND\_FAIL element [55, 88](#)  
 testing the adapter [31](#)  
 TIBCO BusinessEvents [15](#)  
 TIBCO Rendezvous [4](#)  
   subjects used by ECM [122](#)  
 tibrvsend API, cannot call [88](#)  
 trace files [50, 82](#)  
 trace logs [38](#)  
 Trace section [47, 79](#)  
 TRACE\_EMS\_EPM\_ERROR\_MSGS element [48, 80](#)  
 TRACE\_EMS\_HEARTBEAT\_MSGS element [48, 80](#)  
 TRACE\_HEAP element [48, 80](#)  
 TRACE\_LEVEL element [49, 81](#)  
 TRACE\_SWITCHES element [50, 82](#)  
 trackingIdDestination Name parameter [59, 93](#)  
 trackingIdSubject parameter [59, 93](#)  
 trailer record  
   in output file [133](#)  
 transactionDelay parameter [60](#)  
 transferType parameter [62](#)  
 translation table, specifying [53](#)  
 transport mechanisms [3](#)  
 triggerDestinationName parameter [60](#)  
 triggerDestinationType parameter [60](#)  
 triggerFieldName parameter [60](#)  
 trigger-subject method [7](#)  
 triggerSubjectName parameter [60](#)  
 truncateRecords parameter [95](#)  
 type tag [66, 104](#)

## U

UNIT\_CWK element [89](#)  
 UNIT\_GDG element [89](#)  
 UNIT\_OUTPUT element [89](#)  
 UNIT\_PRG element, subscriber [89](#)  
 UNIT\_TRACE element, publisher [50](#)  
 UNIT\_TRACE element, subscriber [82](#)  
 usage guidelines  
   FilePublisher [70](#)  
   for publisher and subscriber [125](#)  
 useExplicitConfirmation [122](#)  
 useExplicitConfirmation parameter [62, 101, 122](#)  
 useFieldWidth parameter [64](#)  
 useFilePolling parameter [60](#)  
 useFixedRecordFile parameter [94](#)  
 userids, mapping [88](#)  
 useTrackingId parameter [59, 93](#)

## V

value tag [67, 68, 104](#)  
 value tag for MESSAGE\_FIELDS parameter [67](#)  
 VOLSER\_CWK element [90](#)  
 VOLSER\_GDG element [90](#)  
 VOLSER\_OUTPUT element [90](#)  
 VOLSER\_PRG element, subscriber [91](#)  
 VOLSER\_TRACE element, publisher [50](#)  
 VOLSER\_TRACE element, subscriber [82](#)  
 volume, MVS [50, 82](#)  
 vsamFileMode parameter [102](#)  
 vsamLogFile parameter [102](#)  
 vsamUseLog parameter [103](#)

## W

wire format [3](#)  
 work (.CWK) files  
   grouping [91](#)  
   storing [90](#)  
 WORKFILE\_DATASET element [91](#)

WRITE\_TO\_SYSLOG element [55](#), [91](#)

## Z

zoned datatypes [131](#)

zoned format [13](#)