EDI Pros

Solid software with a clear future

AutoComm™ Enterprise

User's Guide

AUTOCOMM™ ENTERPRISE

User's Guide

© Copyright EDI Pros, Inc. 2006
238 Capitol Street • Salinas • CA • 93908
Phone 831.753.2352 • Fax 831.753.7762

Table of Contents

Introduction i

CHAPTER 1

Prerequisites 1

CHAPTER 2

Installation 2

CHAPTER 3

Quick Start 4

Main Menu 4
Scheduler Setup 4
Alert Action Setup 5
Connector API Packages 5
FTP Network Setup 5
Testing 5

CHAPTER 4

How it Works 7

General 7
Sending Data 8
Receiving Data 9

CHAPTER 5

Reference 11

Scheduler Setup 12
Scheduler Entry Parameters 16
Last Submission Details 18
Alert Action Setup 19
Alert Action Parameters 22
Batch Summary Display 24
Batch Detail Display 26
FTP Network Setup 28
FTP Network Parameters 30
Change FTP Script 39
Communications Error Conditions 41
FTP Network Statistics 44
Application Program Setup 45
Application Parameters Setup 45
Application Program Setup 45
Message Master Maintenance 45
Batch Purge Requestor 46
Connector API Packages 48
Connector APIs 50
Data Files 55
Calling AutoComm from other applications 58

APPENDIX A

Special Notices 59

Introduction

AutoComm™ will play a key role on your iSeries.

AutoComm™ is a powerful tool that allows you to send data to and receive data from your VAN (Value Added Network). AutoComm is powered by The Supervisor®, which includes

- An intuitive tool for scheduling when AutoComm will run
- Flexible alert action setup for alerting key personnel when AutoComm is in critical condition
- A comprehensive batch summary display for keeping a history of AutoComm processes.

Chapter

1

Prerequisites

Prerequisites for AutoComm to run properly.

Before installing AutoComm, be sure the following prerequisites have been met.

- Your AS/400 or iSeries is at an OS level of V5R1 or higher.
- For FTP, your iSeries must be able to connect to your FTP client(s).

Chapter 2

Installation

Installing AutoComm on your iSeries.

Installing AutoComm onto your iSeries is simple. Follow these steps to complete the installation.

1. Insert the CD-ROM into the optical device on your AS/400 or iSeries.
2. Sign on as QSECOFR.
3. At a command line, type the command **LODRUN DEV(*OPT)** and press ENTER.

After a short delay, the product install screen will appear.

4. Enter the name of the AS/400 or iSeries libraries to install the product into or leave the defaults. The installation process will create the new libraries.
5. Enter the name of a user profile that will own the product objects.[1]

The install process can be cancelled by pressing F3.

6. Press the F19 key (shift + F7) to begin the install process.

During the install process various prompts may appear, such as the creation of new libraries verification prompt, object ownership verification prompt and a duplicate object notification prompt. If you receive any prompts and do not want to proceed with the install process, press F12 from the prompt screen to restart the install.

[1] Defaults have been provided on the product install screen and the product can be installed using the default values, however it is recommended these values be reviewed prior to beginning the install process.

An information message appears when the install process has completed.

Product Installation is now complete

Chapter

3

Quick Start

After installing AutoComm™, getting up and running is a snap.

With a little help from The Supervisor®, you can have AutoComm communicating with your VAN or trading partner in minutes. Here's how.

Main Menu

Ensure that the installation libraries entered during installation are in your library list. This can be verified by typing "DSPLIBL" on a command line. To add the libraries, type "EDTLIBL" on a command line and enter the new libraries. See your iSeries administrator for further assistance and for adding the libraries permanently to your library list. Enter the main menu of The Supervisor® by typing "SUPER" at a command line. See your iSeries administrator for adding this as an option to your business applications.

Scheduler Setup

From the main menu, select option **1. Scheduler Setup**. To add an entry, press F6. From the App. Name field, press F4 and select AutoComm from the pop-up window. Enter the time, and enter an "X" for every day you want AutoComm to run at the entered time. Enter a user profile that has a job description that includes AutoComm's libraries in its library list. Or enter the qualifying job description to override the user profile's job description. If desired, enter the job queue that AutoComm should be submitted to when run. Press Enter. In the parameters pop-up window, press F4 to select a network to communicate with for the schedule entry. Select the network and enter either "Y" or "N" for the Send and Receive parameters. Press Enter to save the parameters and press F12 to close the window. Now press Enter to verify the schedule entry addition.

Alert Action Setup

From the main menu, select option **2. Alert Action Setup**. To add an entry, press F6. From the App. Name field, press F4 and select AutoComm from the pop-up window. Enter a message severity of 20. Position the cursor to the Action field, press F4 and select an alert action. Enter the appropriate recipient in the format that matches the type of alert action entered. Press Enter. In the parameters pop-up window, press F12 to close the window. Now press Enter to verify the alert action entry addition.

Connector API Packages

From the main menu, select option **30. Advanced Options**, then option **17. Connector API Packages**. If your EDI package is shown in the list, set it as the default package by using option 2 to change the Default field to "Y" and all the others to "N". Also, be sure that the default package is active ("A"). See your data extracting product's user guide for further setup that may be required.

If using an unlisted data extracting product and your data extractor does not have a flat-file interface for send data and receive data, then see the Connector API Packages chapter for advanced options.

If your data is available in a flat file, create a new API package. Press F6 and enter the name and description of your custom API package. Be sure it is the default package and it is active. Press Enter two times to add the package. Enter option 6 to enter the flat file and library information. Press Enter to save the information and press F12 to close the window.

FTP Network Setup

From the main menu, select option **17. FTP Network Setup**. Enter option 5 for the parameters pop-up window. Enter the parameters provided by your VAN or FTP client. Or press F9 to select a template, then enter the appropriate User ID and Password. Enter a Y or N for archiving the data. Press Enter to save the values and press F12 to close the window.

Your VAN may provide special parameters for testing. It is wise to test the data communication before communicating with live data. The FTP parameters may be changed at any time.

Testing

From the main menu, select option **1. Scheduler Setup**. Enter option 1 to immediately submit the schedule entry to batch. From the main menu, select option **3. Batch Summary Display**. Enter option 5 to view the batch detail. If there were any

errors, they will be listed in the batch detail with a high severity. The error messages should provide enough detail to troubleshoot the problem.

Chapter 4

How it Works

This section is important for advanced set up and may assist with troubleshooting.

AutoComm takes many fail-proof steps to ensure your data is safe and recovery is possible during unforeseen communications or setup failures. The more you know how AutoComm works, the easier it will be to troubleshoot connection and data problems.

General

Figure 1 is a general overview of the components in AutoComm™ and The Supervisor®. The Supervisor® monitors and controls AutoComm™. The Supervisor® has the following features:

- Scheduling
- Application control via API
- Alert Actions
- Logging

There are two major components in AutoComm™:

- FTP Parameters
- Connector API Packages

The FTP Parameters define how you connect with your client (or network). The other options available are also network-specific. Templates are available to help you set up popular VANs, banks and trading partners. The Connector API Packages define how the FTP data will integrate with your applications. Preconfigured packages are available

for connecting to some of the most popular applications. Source is provided in the CUSMP02SRC source file to help you create your own connector APIs.

When a network is created and the FTP parameters are populated, two subdirectories are created in the AutoComm directory, which is located in the root of the IFS. If the AutoComm directory does not exist, it is created as well. The two subdirectories are:

```
{Network ID}InBox (ie "D1InBox")

{Network ID}OutBox (ie "D1OutBox")
```

These "Local directories" may be overridden with a custom local directory. When a send and receive session is initiated, the send session is always processed before the receive session.

If either of the receive or send methods are *ALTCOMM, then AutoComm calls the Connector API Package **Alt Comm Valid'n Program** (if entered) before processing the send/receive steps. FTP communications are not performed when either *ALTCOMM or *NONE is selected as the send/receive method.

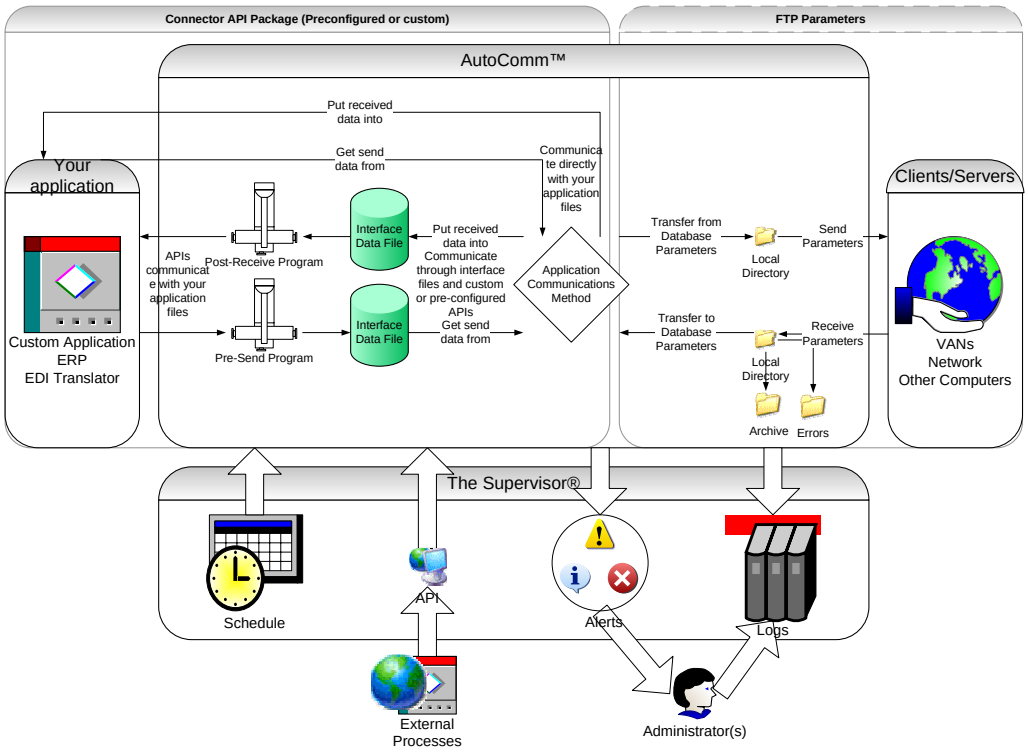


Figure 1

Sending Data

When AutoComm sends data, it follows these steps:

1. If the Package Code for the network is not *NONE and the **Connector API Package Data Files** Get Send Data From field is not *NONE:

    a. Calls the Connector API Package **Pre-Send Program** (if entered).

    b. Copies the the **Connector API Package Data Files** Get Send Data From file to the Local Directory.

2. If the send method is *FTP, then AutoComm FTPs the data file(s) to the network.

3. If the send method is *ALTCOMM, then AutoComm calls the Connector API Package **Alternate Comm Program**.

4. AutoComm compares the FTP log to the Error Conditions.

5. If no errors are detected during transmission and Local Delete = "Y", then AutoComm deletes the Send File in the Local Directory.

6. AutoComm Calls the Connector API Package **Post-Send Program** (if entered and the Send File existed in the Local Directory).

Receiving Data

When AutoComm receives data, it follows these steps:

1. If the Receive Method is *FTP:

    a. AutoComm FTPs the data file(s) from the remote system.

    b. If the receive was successful and Remote Delete option is "Y", then AutoComm FTP deletes the data file(s) on the remote system.

    c. If archiving is turned on, then AutoComm creates a subdirectory called {Network ID}Archive and copies the data file(s) to the Archive.

2. If the Package Code for the network is not *NONE and the **Connector API Package Data Files** Put Received Data Into field is not *NONE:

a. AutoComm attempts to copy the data file from the Local Directory to the "Put received data into" data file defined in the Connector API Package data files:

b. If AutoComm successfully copies the data file, then AutoComm deletes the data file from the Local Directory.

c. If AutoComm cannot copy the data file (usually due to a setup issue), then it creates a subdirectory called {Network ID}Errors and moves the data file into the Errors subdirectory.

3. AutoComm calls the Connector API Package **Post-Receive Program** (if entered).

If either of the receive or send methods are not *FTP, then AutoComm calls the Connector API Package **Alternate Comm Program** (if entered) after processing the send/receive steps.

Chapter

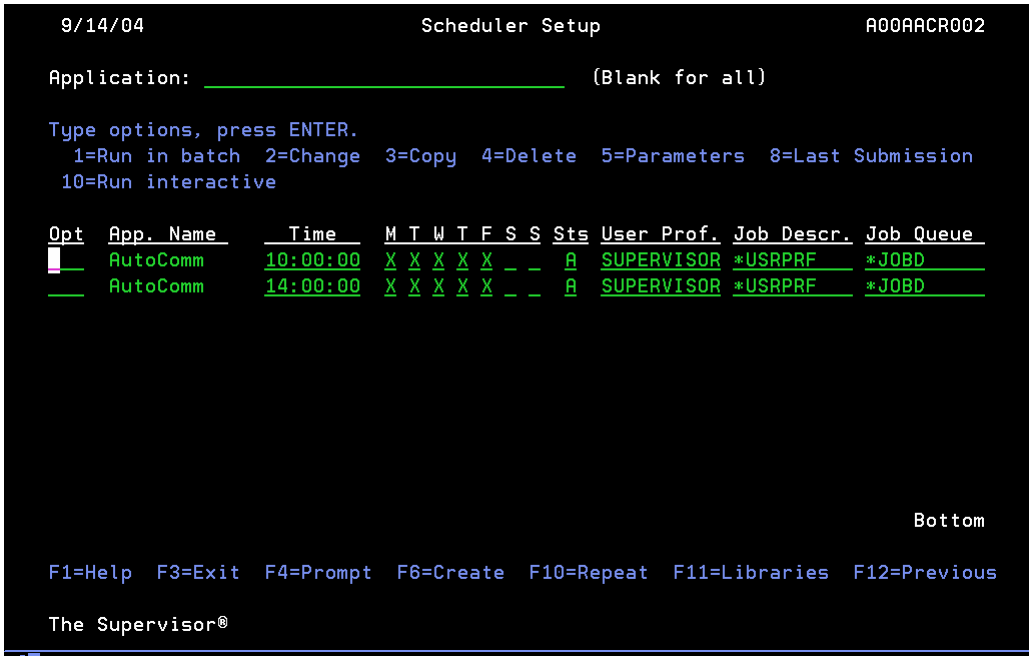
5

Reference

The following Reference section provides detailed information about every menu option available in AutoComm™. Use this section to fine-tune your understanding of AutoComm.

After understanding how AutoComm works (see the How it Works chapter), review the following reference information to learn about every option available in AutoComm. All the reference information is available in the help text by pressing F1 on any screen in AutoComm.

Scheduler Setup



The Scheduler Setup display shows the scheduling information for jobs to be executed using The Supervisor®. Entries are sorted by application code and time. Any entries that have application parameters that are required, but no parameter sets have been entered will display in red. Entering this display causes a check to take place to ensure that the SUPERVISOR subsystem is active. If it is not active, it is started.

It is important to note that the user profile's job description (when Job Descr. is *USRPRF) or the Job Descr. entered includes the libraries where The Supervisor resides in its library list.

From this display you can

- View scheduled job information
- Create scheduled jobs
- Change scheduled job information
- Delete scheduled job entries
- Enter specific application parameter values
- View the last submitted information
- Submit a job immediately

- Run a job interactively

Options

1 Submit a job immediately to the job queue that is specified on the schedule entry.

2 Change the days, status, user profile, job description or job queue of the schedule entry.

3 Copy the values from an existing job schedule entry and create a new job schedule entry.

4 Delete an existing job schedule entry.

5 View, add, change or delete application parameter values for the selected job schedule entry.

8 View the last submitted information of the job schedule entry.

9 Refresh the job schedule information.

10 Execute a job interactively. Note that jobs being executed interactively utilize the current user's user profile, job description and job queue instead of the scheduled entry's values.

Function Keys

F1 Help

F3 Exit program

F4 Prompt for values

F6 Create new schedule entries

F10 Enter option and press F10 to repeat the option

F11 Show job description and job queue libraries

F12 Close the subfile of data or exit the program

HEADER FIELDS

Application

The logs may be subset by an application by entering the application name or by pressing F4 and selecting an application. Leave blank to display all.

DETAIL FIELDS

Opt

Enter an option here (see "Options").

App. Name
Name of application to be processed. Press F4 to select an application while creating a new schedule entry.

Time
Enter military time with or without the colons for the schedule entry job to be submitted.

M T W T F S S (X, Blank)
Days of the week, Monday through Sunday. Enter an "X" for each day the schedule entry should be submitted.

Sts (A, I)
Actives or deactivates a schedule entry. Enter "A" for active or "I" for inactive.

User Prof.
Enter a valid user profile. The schedule entry job will run under the user profile entered. The reports will be created using the user profile entered.

Job Descr. (Name, *USRPRF)
Enter either a valid job description or "*USRPRF". When a job description is entered, the environment of the job description will be used for the schedule entry job as it is processing. If the location of the job description is not in the user profile's library list, press F11 to enter the library name. When *USRPRF is entered, the job description in the user profile field is used as the job description of the submitted job.

Job Queue (Name, *JOBD)
Enter either a valid job queue or "*JOBD". When a job queue is entered, the schedule entry job is placed in the job queue. If the location of the job queue is not in the user profile's library list, press F11 to enter the library name. When *JOBD is entered, the schedule entry job is placed in the job queue named in the specified job description.

Notes about the SUPERVISOR Subsystem

The SUPERVISOR subsystem is started by a QSYSWRK Autostart Job Entry called "STRSUPER", using job description STRSUPRSBS, which runs the following command:

STRSBS SBSD(<library>/SUPERVISOR)

The SUPERVISOR subsystem has an Autostart Job Entry called "SCHEDULER", using job description SUPERVISOR, which runs the following command:

CALL A00SCDPG01

Every 60 seconds, the SCHEDULER job checks the schedule for any schedule entries that need to run.

If The Supervisor® library is renamed or all the objects are moved to another library, it is important to change the library name in all the objects mentioned above.

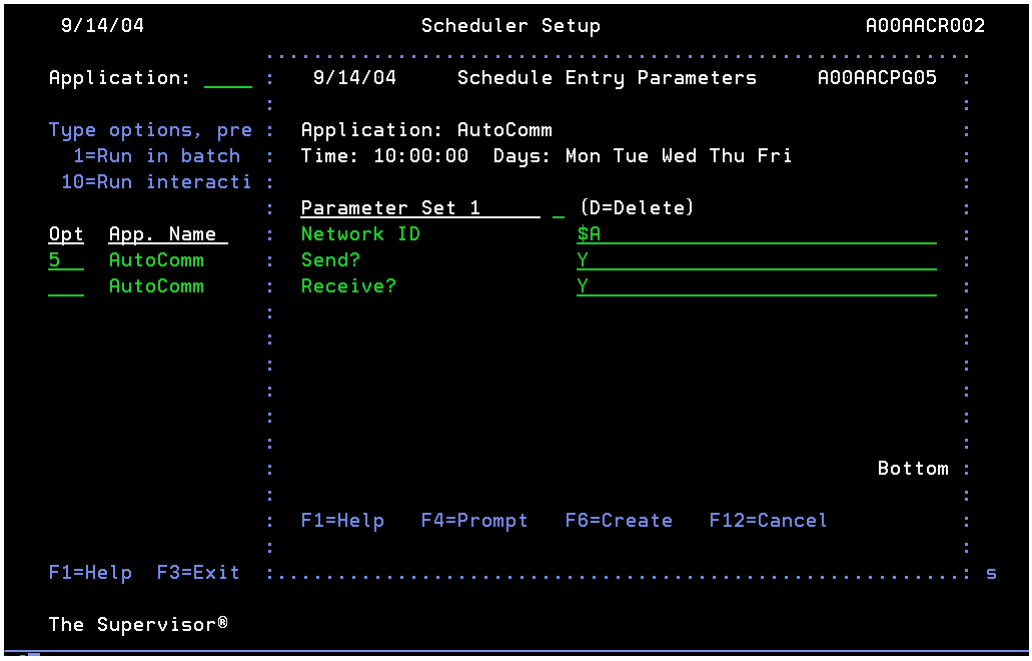
The Supervisor® cannot process schedule entries while its library (or some of the objects) is locked (possibly during backup). When this happens, The Supervisor® logs the following QSYSOPR message.

SUPERVISOR: OBJECT(S) LOCKED. CANNOT PROCESS REQUEST(S).

Once the lock is removed, The Supervisor® logs another message.

SUPERVISOR: OBJECT(S) UNLOCKED. PROCESSING CONTINUED.

Scheduler Entry Parameters



The Scheduler Entry Parameters window shows the values that will be used for each of the application's parameters. Multiple parameter sets may be entered to execute the job multiple times with different parameter set values.

From this display you can

- View schedule entry parameters
- Create new parameter sets
- Change values in existing parameter sets
- Delete parameter sets

Options

D Enter a "D" in the field to the left of "(D=Delete)" to delete a parameter set.

Function Keys

F1 Help

F3 Close window

F4 Prompt for values

F6 Create new parameter set

F12 Close window

FIELDS

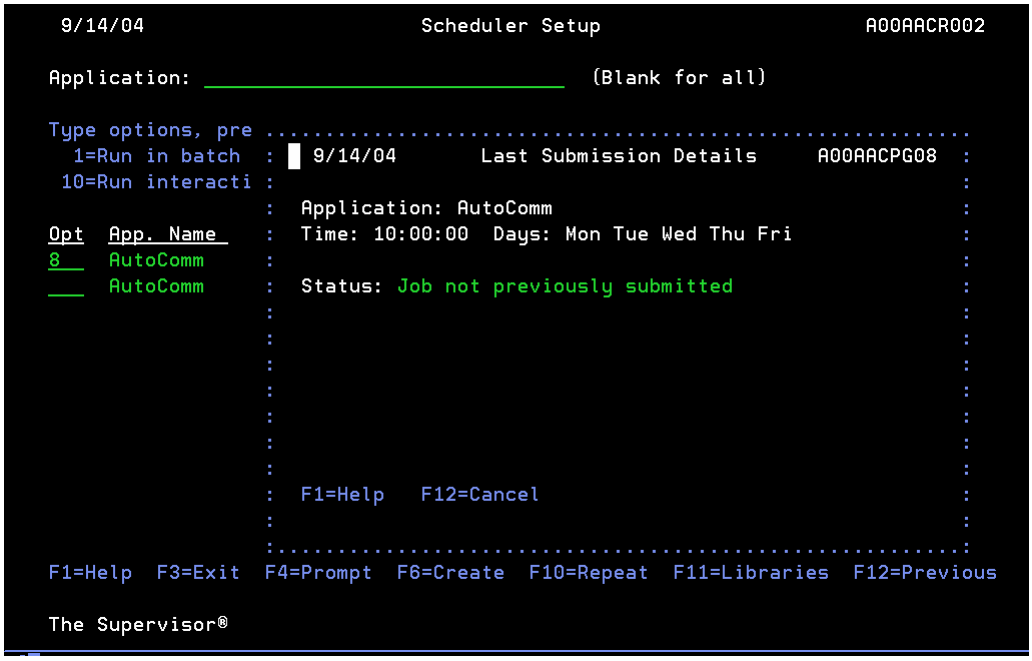
The fields will vary from one application to another. As an example, the fields that display for AutoComm (if purchased) are:

Network ID (see "FTP Network Setup")
Enter a valid 2-character network id to communicate with during execution of this parameter set. Press F4 to select a network.

Send? (Y, N)
Enter "Y" for Yes (execute send) or "N" for No (do not execute send).

Receive? (Y, N)
Enter "Y" for Yes (execute receive) or "N" for No (do not execute receive).

Last Submission Details



The Last Submission Details window shows information related to when the particular schedule entry was executed. This includes when the "Run in batch" and "Run interactive" options were used.

From this display you can

- View last submission details

Function Keys

F1 Help

F3 Close window

F12 Close window

FIELDS

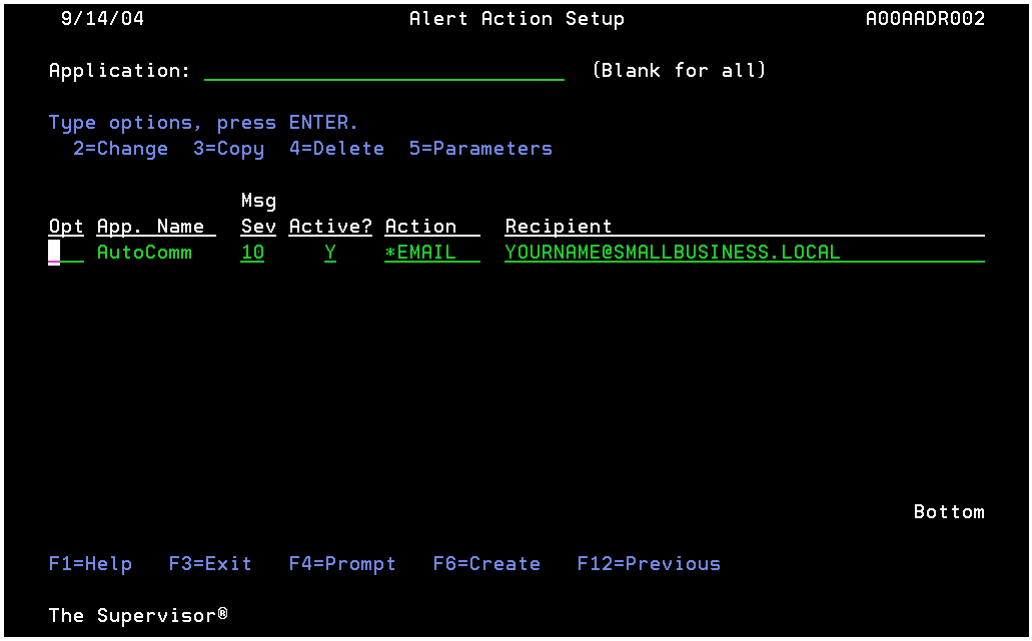
Status

Either "Job not previously submitted" or "Job successfully submitted" will be displayed.

Last Submitted on Date/Time

When the schedule entry was last executed. No values are displayed when the status = "Job not previously submitted".

Alert Action Setup



The Alert Action Setup display allows the creation and maintenance of alert actions. Alert actions are the types of responses that are sent to an individual in the event of a high-severity message while an application is running under the supervision of The Supervisor®. Alert actions are sent upon completion of an application's job. Entries are sorted by application code.

The following alert actions can be sent:

Action	Description	Recipient Type
*BRKINQ	Sends break inquiry message	Workstation
*BRKMSG	Sends break message	Workstation
*BRKMSGU	Sends break message	User
*EMAIL	Sends e-mail[2] of all messages that meet the message severity criteria	E-mail Address
*MSG	Sends program message	User

- From this display you can:

[2] Note that the *EMAIL option uses the iSeries' SMTP. Contact IBM for SMTP configuration instructions.

- View alert action information
- Create new alert actions
- Change alert action information
- Delete alert actions
- Enter specific application parameter values

Options

2 Change the message severity, status, action or recipient of the alert action.

3 Copy the values from an existing alert action and create a new alert action.

4 Delete an existing alert action.

5 View, add, change or delete application parameter values for the selected alert action. By entering parameters, the alert action is sent only if the parameters used to submit the job match the alert action parameters.

Function Keys

F1 Help

F3 Exit program

F4 Prompt for values

F6 Create new alert actions

F12 Close the subfile of data or exit the program

HEADER FIELDS

Application

The logs may be subset by an application by entering the application name or by pressing F4 and selecting an application. Leave blank to display all.

DETAIL FIELDS

Opt

Enter an option here (see "Options").

App. Name

Name of application to be processed. Press F4 to select an application while creating a new alert action.

Msg Sev (00-99)

Enter the lowest severity of a message that will trigger an alert action. Enter "00" to always receive an alert action. Message severities may be changed in the Message Master Maintenance display.

Active? (Y, N)
Activates or deactivates an alert action. Enter "Y" for active or "N" for inactive.

Action
The type of alert action that is being sent. Press F4 to select an alert action type. Examples are *BRKMSGU and *EMAIL. See alert action list for complete list of alert action types.

Recipient
The recipient of the alert action. The type of recipient is base upon the type of alert action. See alert action list for list of recipient types. Note that the values in this field are not validated.

Notes about *EMAIL

*EMAIL utilizes the iSeries SMTP server. For assistance on setting up your iSeries SMTP server, please contact IBM. To modify the from E-Mail address to send Internet E-Mail, use the following command:

CHGDTAARA DTAARA(A00EMLDA01) VALUE('<email address>')

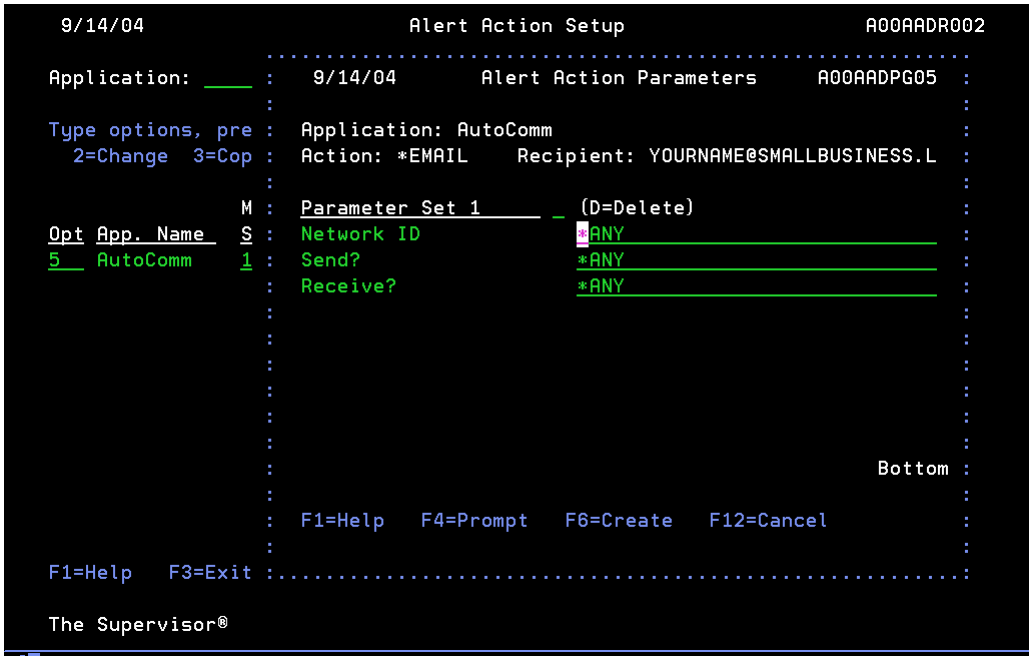
Where <email address> is the from E-Mail address.

*EMAIL also utilizes the time based on the Coordinated universal time offset. be sure the value is correct in order to receive e-mails with the proper timestamp. Pacific Coast is -0800. To change the value, use the following command:

CHGSYSVAL SYSVAL(QUTCOFFSET) VALUE('<Offset>')

Where <Offset> is the coordinated universal time offset.

Alert Action Parameters



The Alert Action Parameters window shows the values that will be used for filtering when a parameter is sent to an alert action recipient. The alert action parameter values are compared to the parameter values that were used for execution. If there is a match (and the maximum severity had been reached), then the alert action is sent. Multiple parameter sets may be entered for multiple filters.

From this display you can

- View alert action parameters
- Create new parameter sets
- Change values in existing parameter sets
- Delete parameter sets

Options

D Enter a "D" in the field to the left of "(D=Delete)" to delete a parameter set.

Function Keys

F1 Help

F3 Close window

F4 Prompt for values

F6 Create new parameter set

F12 Close window

FIELDS

The fields will vary from one application to another. The default value is of every field is "*ANY". *ANY accepts any value for the field to trigger the alert action. If a field is blank, the execution parameter value must be blank to meet the criteria.

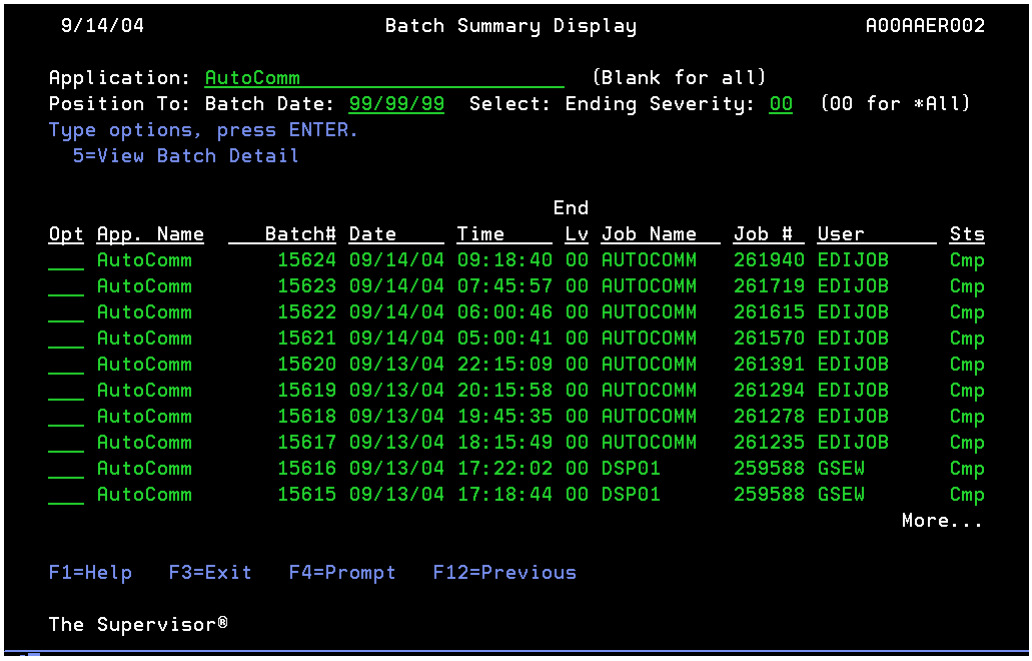
As an example, the fields that display for AutoComm (if purchased) are:

Network ID (*ANY, see "FTP Network Setup")
Enter a valid 2-character network id. Press F4 to select a network.

Send? (*ANY, Y, N)
Enter "Y" or "N".

Receive? (*ANY, Y, N)
Enter "Y" or "N".

Batch Summary Display



The Batch Summary Display shows the history of all jobs executed using The Supervisor®. Jobs are sorted by batch date, descending order.

From this display you can:

- View executed job information
- View the batch detail log entries

Options

5 View batch detail logs.

Function Keys

F1 Help

F3 Exit program

F4 Prompt for values

F12 Close the subfile of data or exit the program

HEADER FIELDS

Application

The logs may be subset by an application by entering the application name or by pressing F4 and selecting an application. Leave blank to display all.

Batch Date
Enter a batch date to position to the jobs that were executed on or before a particular date.

Ending Severity
Enter the lowest severity to be displayed. All jobs with equal or greater severity will be displayed.

DETAIL FIELDS

Opt
Enter an option here (see "Options").

App. Name
Name of application executed.

Batch#
The batch number for the executed job. Batch number sequence is dependant upon each application.

Date
The date the job was executed.

Time
The time the job was executed.

End Lv
The highest severity message logged during execution.

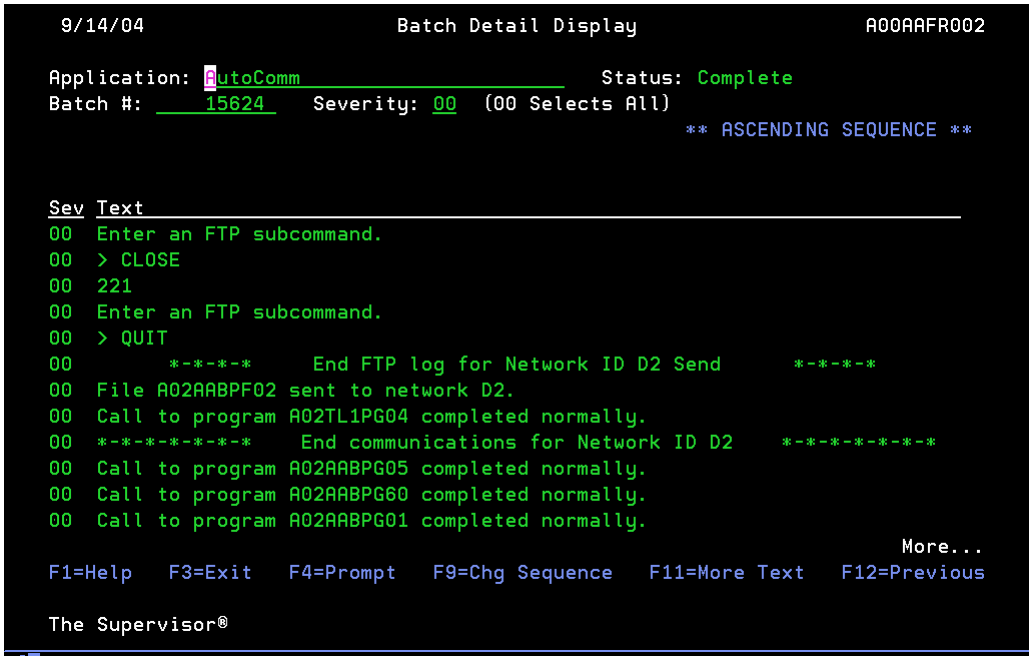
Job Name
The simple job name of the job from which this job was submitted. This is usually the name of the application. Jobs executed from the scheduler are prefixed with the letter "A".

Job Number
The job number of the job from which this job was submitted.

User
The name of the user who submitted this job.

Sts (Act, Cmp)
The status of the job. If the job is running, the status is "Act" (Active). If the job is complete, the status is "Cmp" (Complete).

Batch Detail Display



The Batch Detail Display shows the detail log entries for a particular batch. Jobs are sorted chronologically.

From this display you can

- View the batch detail log entries

Function Keys

F1 Help

F3 Exit program

F4 Prompt for values

F9 Toggle sort sequence by chronological order and reverse chronological order

F11 Unfold for more log entry detail

F12 Close the subfile of data or exit the program

HEADER FIELDS

Application

Enter the application name or press F4 and select an application.

Status (Active, Complete)
Status of job.

Batch #
Enter the batch number of the particular job.

Severity
Enter the lowest severity to be displayed. All log entries with equal or greater severity will be displayed.

DETAIL FIELDS

Sev
The severity of the message logged during execution.

Text
Message text.

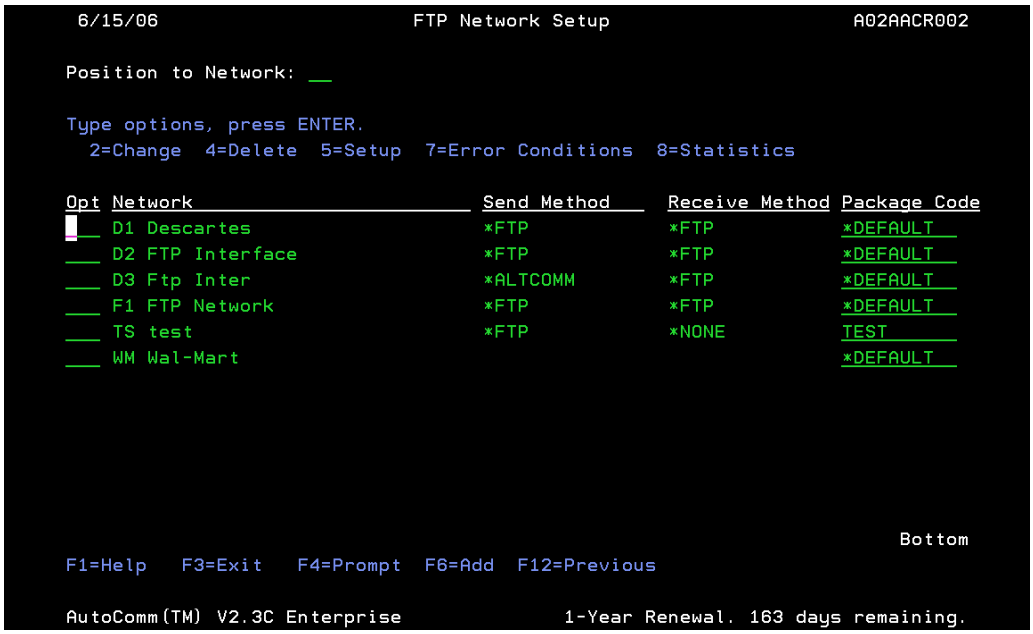
UNFOLDED DETAIL FIELDS

Message #
The number of the message logged. See Message Master Maintenance to maintain message text and severity.

Program
The name of the program that logged the message.

EDI Trans #
The transaction number. Only applies to jobs that can contain multiple transactions in a single batch (like EDI).

FTP Network Setup



The FTP Network Setup display shows the networks available for communications.

The display is sorted by network ID.

If any of the following is necessary, then the networks must be setup through the 3rd party's software and the network listing is loaded from the 3rd party's network listing automatically.

1. To utilize the active Connector API package's (see Connector API Packages) communications (like TL400 dial-up).

2. To interface with a 3rd party software that requires an internal network ID to send and receive electronic data (like TL400).

The networks are re-loaded every time the FTP Network Screen is displayed. For AutoComm to properly acquire the networks, be sure that the 3rd party's libraries are in the library list or are in the Connector API Packages API's Connect System and Database Libraries.

FTP communications are managed by AutoComm. Alternate communications methods (ie dial-up) are managed by your 3rd-party software.

From this display you can

- View available networks

- Create new networks

- View and change FTP network parameters
- View FTP network statistics
- Delete networks

Options

4 Delete an existing network.

5 View or change FTP network parameters.

7 View or change FTP network error conditions.

8 View FTP network statistics.

Function Keys

F1 Help

F3 Exit program

F6 Create new network

F12 Close the subfile of data or exit the program

HEADER FIELDS

Position to Network
Enter the network id to position to. Leave blank to position to the first network.

DETAIL FIELDS

Opt
Enter an option here (see "Options").

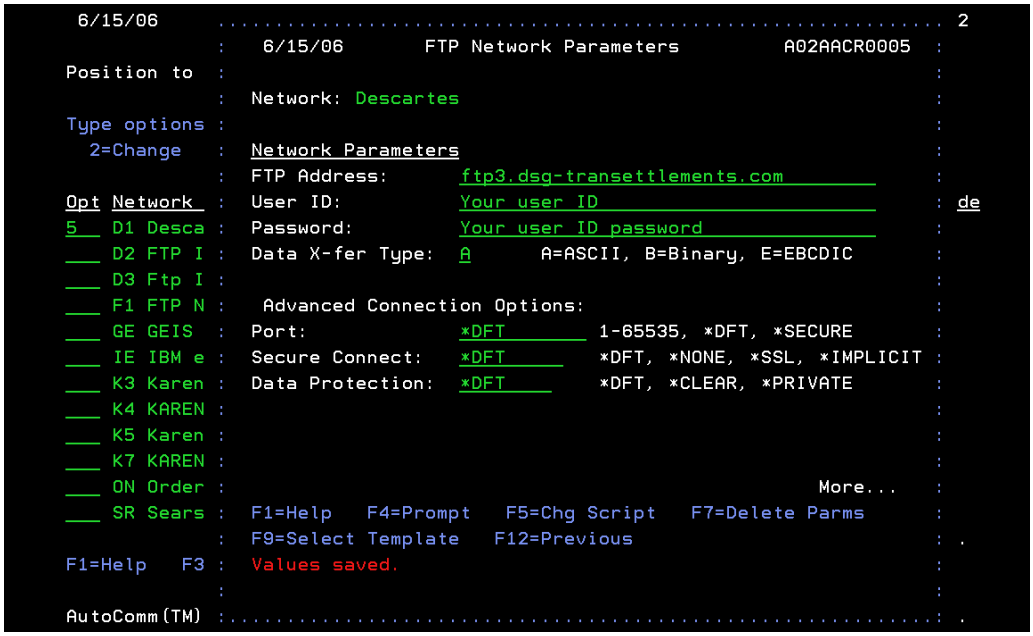
Network
Network ID and description.

Send Method (*FTP, *NONE, *ALTCOMM, blank)
*FTP is the default send method. The method is changed in FTP Network Parameters. A blank indicates that the network parameters have not been entered. See FTP Network Parameters Help for more information.

Receive Method (FTP, blank)
*FTP is the default receive method. The method is changed in FTP Network Parameters. A blank indicates that the network parameters have not been entered. See FTP Network Parameters Help for more information.

Package Code
Specifies which Connector API Package to use when sending and receiving data. The default is "*DEFAULT", which uses the first default API connector package (See Connector API Packages help). Press F4 to select a package code. The package code only applies to FTP transmissions.

FTP Network Parameters



The FTP Network Parameters display shows the FTP communications setup for a selected network.

From this display you can

- View, change and delete network parameters

Function Keys

F1 Help

F3 Close the window

F5 View send and receive script

F7 Clear data from all fields

F9 Select an FTP template to replace all the parameter values with the values in the template

F12 Close the window

FIELDS

Network Parameters

The Network Parameters include general information necessary for the connection.

FTP Address (systemname [portnumber])

The systemname is the name or Internet address of the remote system. The portnumber is the port number to use for this session until the connection is closed. This is optional and if you do not specify a port number, one is chosen.

User ID
Enter the user id for the connection.

Password
Enter the password for the connection.

Data X-fer Type (A, B, E)
Specifies how data will be transferred. Data representation specifies how the data is represented when transferred.

- A - ASCII is the default transfer type. It is normally used except when both client and server use EBCDIC data representation. IBM AS/400, VM, and MVS use EBCDIC.

- B - IMAGE transfer type is used to send data as contiguous bits packed in 8-bit bytes. This transfer type should always be used for transferring binary data. Binary image transfer is required for save files and hierarchical file system (HFS) files.

- E - EBCDIC is the most efficient way to transfer data between machines that use this data representation.

Advanced Connection Options

Port (1-65535, *DFT, *SECURE)
Specifies the port number used for connecting to the FTP server. Normally the "well-known" port value of 21 is used to connect to the FTP server. Under some circumstances, the FTP server may be contacted at a port other than port 21. In those situations, the port parameter may be used to specify the server port to connect to.

- *DFT - The value 00021 is used.

- *SECURE (V5R2 or higher) - The value 00990 is used. Port 990 is reserved for secure FTP servers which immediately use Transport Layer Security (TLS) or Secure Sockets Layer (SSL) protocols to encrypt data.

- port_value - The requested port value is used. This value is validated to ensure it is in the proper range. If 990 is specified, the FTP client will perform the same functions as if *SECURE were specified.

Secure Connect (*DFT, *NONE, *SSL, *IMPLICIT) (V5R2 or higher)
Specifies the type of security mechanism to be used for protecting information transferred on the FTP control connection (which includes the password used to authenticate the session with the FTP server). Transport Layer Security (TLS) and Secure Sockets Layer (SSL) are compatible protocols which use encryption to protect data from being viewed during transmission and verify that data loss or corruption does not occur.

- *DFT - If the Port parameter specifies *SECURE or 990, *IMPLICIT is used; otherwise, *NONE is used.

- *IMPLICIT - The FTP client immediately attempts to use TLS/SSL when connecting to the specified FTP server (without sending an AUTH subcommand to the server). If the server does not support implicit TLS/SSL on the specified port, or the TLS/SSL negotiation fails for any reason, the connection is closed.

- *SSL - After connecting to the specified FTP server, the FTP client sends an AUTH (authorization) subcommand requesting a TLS/SSL protected session. If the server supports TLS/SSL, a TLS/SSL negotiation performed. If the server does not support TLS/SSL or the TLS/SSL negotiation fails, the connection is closed.

- *NONE - The FTP client does not use encryption when connecting to the specified FTP server.

Data Protection (*DFT, *CLEAR, *PRIVATE) (V5R2 or higher)
Specifies the type of data protection to be used for information transferred on the FTP data connection. This connection is used to transfer file data and directory listings. The FTP protocol does not allow protection of the data connection if the control connection is not protected.

- *DFT - If the Security Connect parameter specifies a protected control connection, *PRIVATE is used; otherwise, *CLEAR is used.

- *PRIVATE - Information sent on the FTP data connection is encrypted. If the Secure Connect parameter specifies that the FTP control connection is not encrypted, *PRIVATE cannot be specified.

- *CLEAR - Information sent on the FTP data connection is not encrypted.

Send Parameters ("put")
The Send Parameters allow entry of the required FTP parameters for sending (PUTing) a file to a remote location. AutoComm sends files in the Local Directory. By default, the Local Directory is "/AutoComm/{Network ID}OutBox". If there is a Connector API package defined for the network, then the file is copied into the Local Directory from the location specified in the Connector API Package before communications. See Connector API Packages help for information on how to specify and populate the file. AutoComm™ will ensure that files exist which match the Send File criteria in the specified Local Directory before starting an FTP Send session.

Send Method (*FTP, *NONE, *ALTCOMM)
The Send Method specifies how the communications will be processed.

- *FTP - communicate using FTP.

- *NONE - No communications will take place. This is a good option when other parties are initiating the FTP sessions. Note that the connector API will still execute.

- *ALTCOMM - Launches the Alternate Communication Program defined in the Connector API Package.

Local directory
The name of a file directory, library, or other system-dependent file-group designator, on the local system. The default is "/AUTOCOMM/{Network ID}OUTBOX". The default directory will be created automatically. A custom directory will not be created automatically.

Remote directory
The name of a file directory, library, or other system-dependent file-group designator, on the remote system.

Send file (*, *IFSNAME)
Specifies the file(s) in the Local Directory to be sent. "*" is the default. Specify one or more AS/400 file members, documents, or other local AS/400 files to be transferred to the remote system. The name given to the file on the remote system is generated automatically by the client. You can use an asterisk to send all the members in a file or all the documents in a folder. For example, MYFILE.* transfers all the members of file MYFILE. Enter "*IFSNAME" to send only the file that was transferred from the IFS.

Remote filename
The name of the delivered file on the remote system if different from the "Send file". If the file name does not already exist, the FTP client provides a default name. If a remotefile of that name already exists, the contents are replaced by the contents of the localfile. Enter "*SNDFILE" to not specify a name and to use the same name as the local file name.

Translator Options
Options used for communicating with your selected API Package. Pre-defined API packages utilize this information to send instructions to the translator.

Wrap code (W, U)
Enter "W" for wrapped or "U" for unwrapped. The wrap code is not used during transmission, but may be passed to your 3rd party software for processing. This allows the software to create the data in the format requested by the receiver.

Record length (80, 132)
This is the length of the maximum number of character per line being transmitted. Enter "80" or "132". If End of Line Char is *FIXED, then the record length is used as the record length of the Send Data file (A02AABPF02). The record length may also be passed to your 3rd party software for processing. This allows the software to create the data in the format requested by the receiver.

Transfer from Database Parameters

When a Connector API Package is selected, options used for copying files from DB2 to the Send Local Directory.

End of Line Char (*CRLF, *LF, *CR, *LFCR, *FIXED)

Specifies the end-of-line characters to insert into the outbound file. If one of the end-of-line character options is selected ((*FIXED) is not specified) the database file records are transformed to variable-length text lines before they are sent. Each database file record is trimmed of any trailing blanks. Then, the end-of-line character is appended to the end of the text line. The text line is copied to the stream file. Use *FIXED for unwrapped data when the send file is A02AABPF02.

- *CRLF - Carriage-return followed by line-feed is appended to the end of each line.
- *LF - Line-feed is appended to the end of each line.
- *CR - Carriage-return is appended to the end of each line.
- *LFCR - Line-feed followed by carriage-return is appended to the end of each line.
- *FIXED - The lines in the send file are written as fixed length records. CR and LF characters are not added at the end of each line, trailing blanks are not removed from the end of each record. The length of the file records equals the length of the database file records. Only stream files and database files with compatible encoding schemes which do not result in the expansion of data record lengths are valid with this option. If an incompatible combination of encoding schemes is detected, the operation will fail. The following encoding scheme combinations are supported.
    - Single-byte to single-byte
    - Double-byte to double-byte
    - Mixed ASCII to mixed ASCII
    - Mixed EBCDIC to mixed EBCDIC

IFS Filename (name, *AUTO)

Then name of the file in the Send Local Directory when copying a file from DB2.. *AUTO allows AutoComm to automatically create a remote filename. The name it creates is "ED{timestamp}" where timestamp is the time/date of transmission, in the format of hhmmssDDDDDDDD. If a remote filename of that name already exists, the contents are replaced by the contents of the local file.

Data Handling

Local Delete (Y, N)
Enter "Y" to delete the file(s) on the local system after successfully sending them. Only the file(s) sent will be deleted.

Receive Parameters ("get")
The Receive Parameters allow entry of the required FTP parameters for retrieving (GETing) a file from a remote location. The file(s) being retrieved are placed in the Local Directory, and then copied to the file defined in the Connector API Package Data Files. By default, the Local Directory is "/AutoComm/{Network ID}InBox". See Connector API Packages help for information on how to specify and acquire data from the file.

Receive Method (*FTP, *NONE, *ALTCOMM)
The Receive Method specifies how the communications will be processed.

- *FTP - communicate using FTP.
- *NONE - No communications will take place. This is a good option when other parties are initiating the FTP sessions. Note that the connector API will still execute.
- *ALTCOMM - Launches the Alternate Communication Program defined in the Connector API Package.

Remote directory
The name of a file directory, library, or other system-dependent file-group designator, on the remote system. The directory may be entered here or combined with the Receive File.

Local directory
The name of a file directory, library, or other system-dependent file-group designator, on the remote system. The default is "/AUTOCOMM/{Network ID}INBOX". The default directory will be created automatically. A custom directory will not be created automatically.

Receive File
The Receive File is the file to be retrieved from the remote system. All received files are received into the /AutoComm/{Network ID}InBox directory. You can use a "*" to retrieve all members in a file or all members starting with a particular letter. For example, if the remote system is an AS/400 system,

- "MYFILE.*" retrieves all the members of file MYFILE on the remote system.
- "*" retrieves all the files in the Remote Directory.
- "A*" retrieves all documents starting with the letter A.

Local filename (*RCVFILE, *AUTO)
The local file member, hierarchical file system (HFS) file or document, or other file to be used or created. Enter "*RCVFILE" to not specify a name and to use the same name as the remote file name. Enter "*AUTO" to rename the file to ED{timestamp}.

Local replace (Y, N)
Select "Y" to replace a file with the same name in the Local Directory. Select "N" to not replace the file.

Translator Options
Options used for communicating with your selected API Package. Pre-defined API packages utilize this information to send instructions to the translator.

Wrap code (W, U)
Enter "W" for wrapped or "U" for unwrapped. The wrap code is not used during transmission, but may be passed to your 3rd party software for processing. This allows the software to process the data in the format created by the receiver.

Record length (80, 132)
This is the length of the maximum number of character per line being transmitted. Enter "80" or "132". If End of Line Char is *FIXED and Create File is "Y" (see Data Files in Connector API packages), then the record length is used as the record length for the specified Receive Data file. The record length may be passed to your 3rd party software for processing. This allows the software to process the data in the format created by the sender.

Transfer to Database Parameters:
When a Connector API Package is selected, options used for copying files from the Receive Local Directory to DB2.

End of Line Char (*ALL, *CRLF, *LF, *CR, *LFCR, *FIXED)
Specifies the end-of-line characters of the received data. All records are transformed to a fixed-length format as they are received into the database file. The fixed-length is equal to the length of the database file records to which they are copied. If one of the end-of-line character options is selected (*FIXED is not specified) the received data is read up to first occurrence of that character. The end-of-line character is stripped from the record. The remainder of the record is padded with blanks. The data is copied to the database file member. If a record is too long to fit in the fixed-length format, it is truncated, then copied.

- *ALL - Any single or double character combination of carriage-return and line-feed is appended to the end of each line.
- *CRLF - Carriage-return followed by line-feed is appended to the end of each line.

- *LF - Line-feed is appended to the end of each line.
- *CR - Carriage-return is appended to the end of each line.
- *LFCR - Line-feed followed by carriage-return is appended to the end of each line.
- *FIXED - Text lines in the received data are considered fixed-length records of the same length as the database file records to which they are being received into. Any encountered CR, LF, or EOF characters are not stripped from the received data. Tab expansion is not allowed and the TABEXPN parameter is not valid. If the last record in the received data does not fill the database file record, that record is padded with blanks. Only stream files and database files with compatible encoding schemes, which do not result in the expansion of data record lengths, are valid with this option. If an incompatible encoding scheme combination is detected, the operation will fail. The following encoding scheme combinations are supported.
    - Single-byte to single-byte
    - Double-byte to double-byte
    - Mixed ASCII to mixed ASCII
    - Mixed EBCDIC to mixed EBCDIC

Data Handling

Remote Delete (Y, N)

Enter "Y" to delete the file(s) on the remote system after successfully retrieving them. Only the file(s) received will be deleted.

Archive Data (Y, N)

Enter "Y" to archive received files in the /AutoComm/{Network ID}Archive directory. The filenames will be prefixed with a YYMMDDHHMMSS timestamp.

ERRORS DIRECTORY

If there is ever a problem while copying the data from the Receive Local Directory (typically "/AutoComm/{Network ID}InBox") to the destination file, the file is placed in the /AutoComm/{Network ID}Errors directory and a high severity message is logged. This can happen when specifying a non-existent filename for putting received data into. If this should occur for business-critical data then

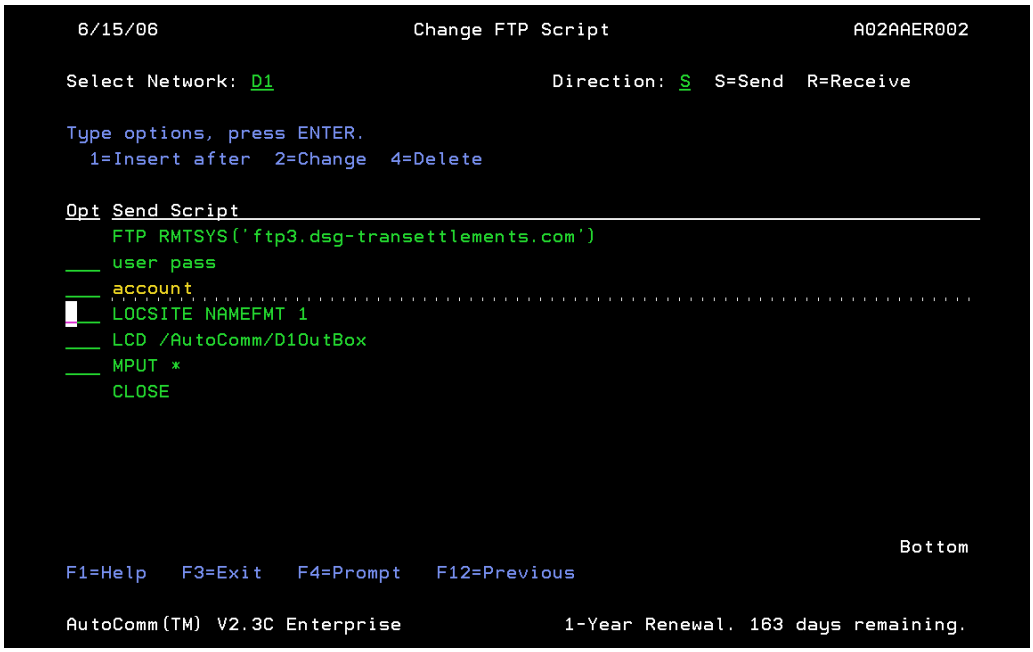
1. Analyze the logs to determine the problem.
2. Once the problem is fixed, copy the received file from the Errors directory into the appropriate Receive Local Directory.

The next time a receive FTP transmission is executed, AutoComm will attempt to copy the data from the Local Directory to the destination file.

References

IBM V5R2 FTP help text.

Change FTP Script



The Change FTP Script screen allows modification to the script that will be executed during an FTP session.

From this display you can

- Insert new script
- Change inserted script
- Delete inserted script

Options

1 Insert new blank line of script after the line with option 1.

2 Change the inserted line of script.

4 Delete the inserted line of script.

Function Keys

F1 Help

F3 Close the window

F7 Reset statistics

F12 Close the window

HEADER FIELDS

Network
Enter the network ID. Press F4 to select from a list.

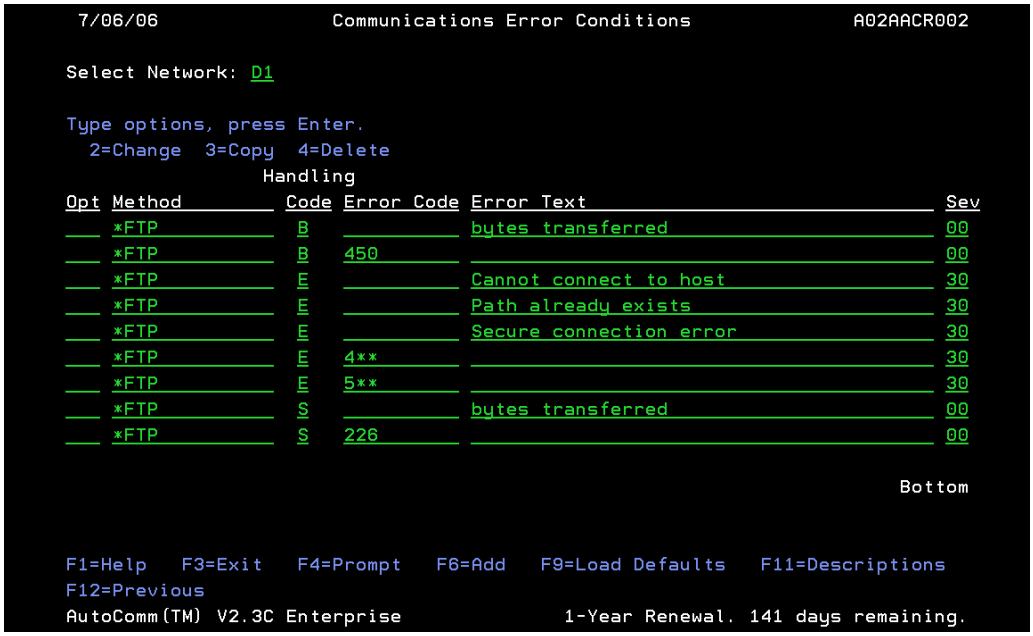
Direction (S, R)
Choose the direction for the FTP script that needs modification. S = Send script (PUT), R = Receive script (GET).

DETAIL FIELDS

Direction (S, R)
Enter an option here (see "Options").

Send Script
Existing and inserted FTP script. Inserted FTP script will be underlined and in yellow. The lines in green are a direct result of entries made on the FTP Parameters screen.

Communications Error Conditions



The Communications Error Conditions screen allows maintenance of communications error handling.

Network selection is mandatory. The display is sorted by Method and Handling Code. This screen is accessed from the FTP Network Setup screen. Default entries are automatically loaded. The defaults may be modified by entering "**..**" in the Select Network field. Modifications to the **..** network will only take affect for new networks and when the Load Defaults function is executed.

From this display you can

- Change error conditions
- Copy error conditions
- Add new error conditions
- Delete error conditions
- Load default error conditions

Options

2 Change an error condition.

3 Copy an error condition.

4 Delete an error condition.

Function Keys

F1 Help

F3 Exit program

F4 Prompt

F6 Add new error condition

F9 Load default error conditions

F11 Fold/Unfold to view optional descriptions for each line

F12 Close the subfile of data or exit the program

HEADER FIELDS

Select Network

Enter the network id. This is mandatory to view all the error conditions for the network. Press F4 to select a network.

DETAIL FIELDS

Opt

Enter an option here (see "Options").

Method (*FTP)

The communications method. If *FTP is selected as the Receive/Send Method in Network setup, then only the *FTP error conditions will apply. Other methods will apply when those methods are selected as the Send/Receive Method in Network Setup. Press F4 to select from a list.

Handling Code (B, E, S)

Enter the appropriate Handling Code for the condition being entered. Press F4 to select from a list.

B - Bypass Error

Error codes entered with a "B", "Bypass Error", are logged as severity "00". Bypass error codes take precedence over "E", "Flag As Error" codes. For example, if one condition specifies "E" with error code "4**" and a severity of "30", and another condition specifies "B" with error code "450", error codes "400" thru "449" and error codes "451" thru "499" will be logged as severity "30", while error code "450" will be logged as severity "00".

E - Flag as Error

Entering an "E" condition will set the severity of any matching error codes to the entered severity.

S - Summary OK Flag

If AutoComm does not find any Summary OK Flag conditions during a SEND FTP transmission, the transmission will be flagged as in error.

Error Code

Enter the Error Code (or the Error Text) for the condition. Enter a "*" to replace the position with a wildcard. Valid FTP code values are XYZ, XY* and X**, where X, Y, Z are arbitrary characters. Press F4 to select codes from a list.

Error Text

Enter the Error Text (or the Error Code) for the condition. During analysis, AutoComm scans the entire FTP log entry for the text entered.

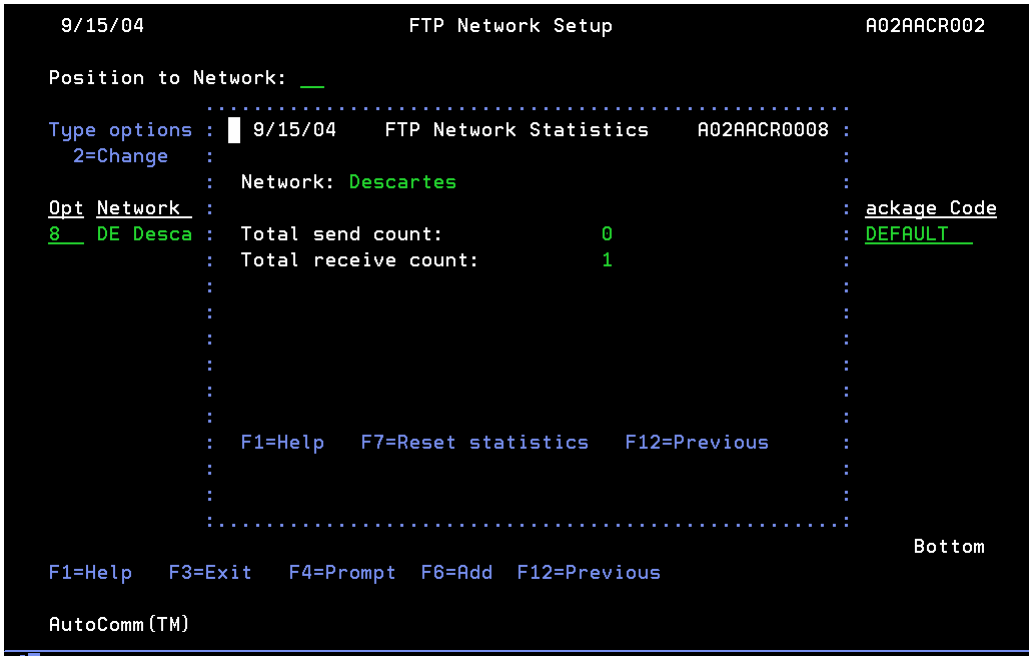
Sev (00, 10, 20, 30, 99)

The Severity only applies to the E handling codes. When an error condition is met, the FTP log entry will be logged at the selected severity. Press F4 to select from a list.

Description

The optional condition description may be viewed and hidden by pressing F11.

FTP Network Statistics



The FTP Network Statistics display shows the FTP statistical data for a selected network.

From this display you can

- View FTP network statistics

Function Keys

F1 Help

F3 Close the window

F7 Reset statistics

F12 Close the window

FIELDS

Total Send Count
Total number of files sent via FTP since last reset.

Total Receive Count
Total number of files received via FTP since last reset.

Application Program Setup

This option is available in the Enterprise version of The Supervisor®. Refer to the User's Guide for The Supervisor® for information.

Application Parameters Setup

This option is available in the Enterprise version of The Supervisor®. Refer to the User's Guide for The Supervisor® for information.

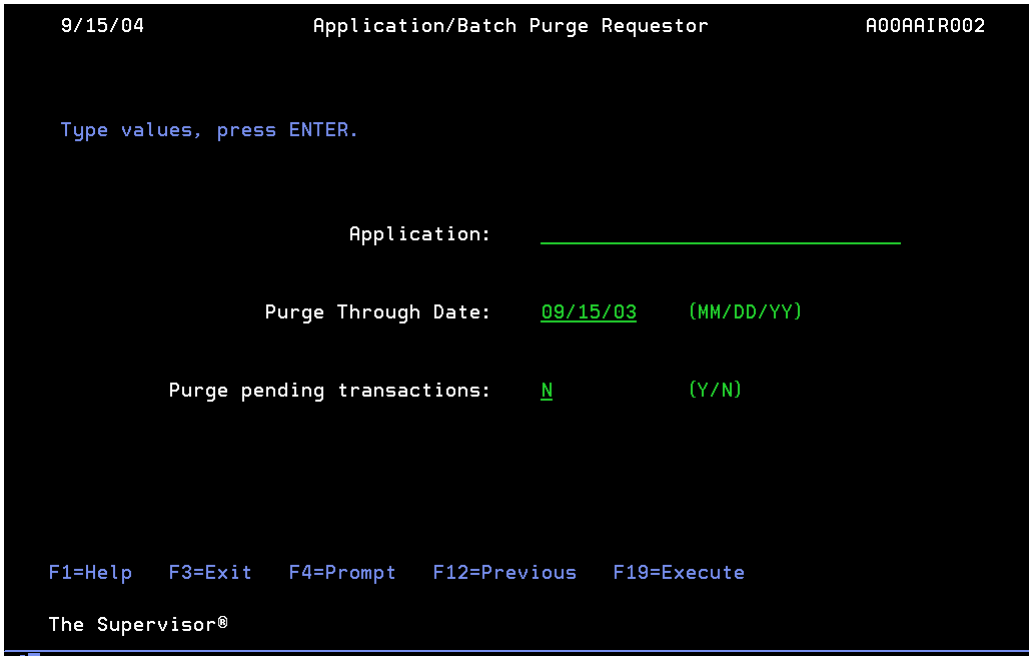
Application Program Setup

This option is available in the Enterprise version of The Supervisor®. Refer to the User's Guide for The Supervisor® for information.

Message Master Maintenance

This option is available in the Enterprise version of The Supervisor®. Refer to the User's Guide for The Supervisor® for information.

Batch Purge Requestor



The Application/Batch Purge Requestor allows you to purge by application. Each application has its own set of data to purge. A purge of an application does the following.

Purge the application's data. For example, AutoComm's purge will purge received FTP data stored for recovery purposes. Electronic data applications will purge their own data plus any transaction message logs from The Supervisor®.

Purge The Supervisor® batch logs for the selected application. The purge of the batch log will take place only if all transaction message logs have already been purged.

From this display you can

- Purge by application through selected date

Function Keys

F1 Help

F3 Exit program

F4 Prompt for values

F12 Close the subfile of data or exit the program

F19 Execute the purge

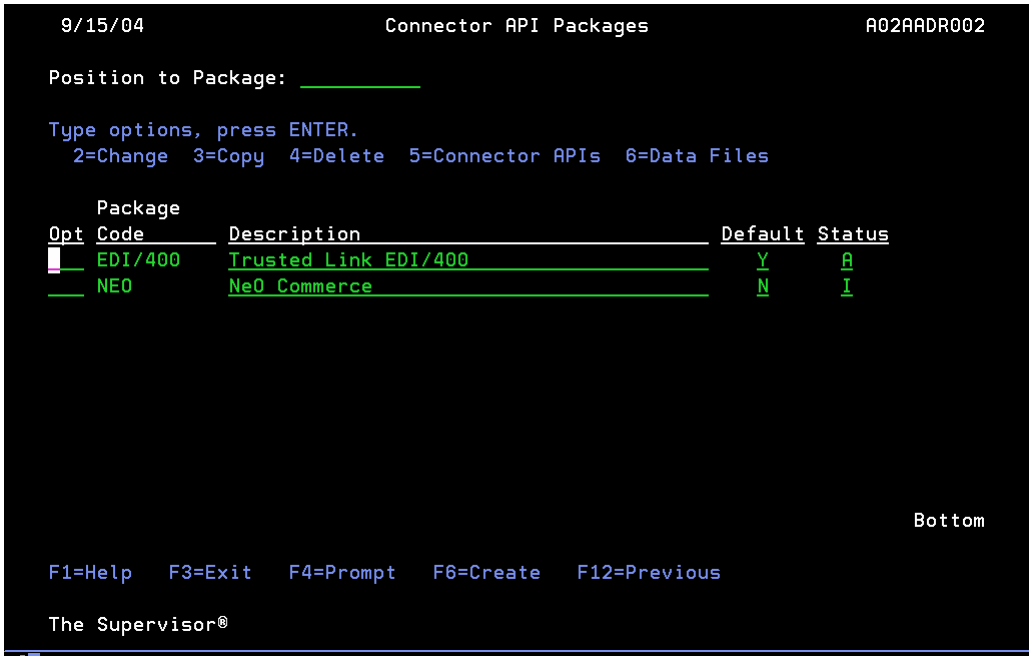
FIELDS

Application
The application to be purged must be entered. Press F4 to select an application from a list.

Purge Through Date
All data will be purged up to the date entered.

Purge pending transactions (Y, N)
This field is used by applications that log transaction-specific messages. If a "Y" is entered, the transaction is purged whether it has or has not completed processing. If an "N" is entered, pending transaction message logs are left alone.

Connector API Packages



A connector API package is a package of connector APIs. A connector API is a program interface that connects AutoComm to your data-handling product. When AutoComm calls a connector API, it sends key information that could be used by your data-handling product to perform data processing or communications. The connector API package may be chosen during schedule entry. See help for the Connector APIs for more information regarding the function of each connector API. From this screen, you may also specify the data files for each connector API package. The data files allow you to choose where you want the data to go after receives and where you want to get the data from for sends.

From this display you can

- View connector API package information
- Create connector API packages
- Change connector API package information
- Delete connector API packages
- View, change connector APIs
- View, change data files

Options

2 Change the description, default or status of a connector API package.

3 Copy the values from an existing connector API package and create a new connector API package.

4 Delete an existing connector API package.

5 View, add, change or clear connector APIs.

6 View or change data files.

Function Keys

F1 Help

F3 Exit program

F4 Prompt for values

F6 Create new connector API packages

F12 Close the subfile of data or exit the program

HEADER FIELDS

Position to Package
Positions to the entered package.

DETAIL FIELDS

Opt
Enter an option here (see "Options").

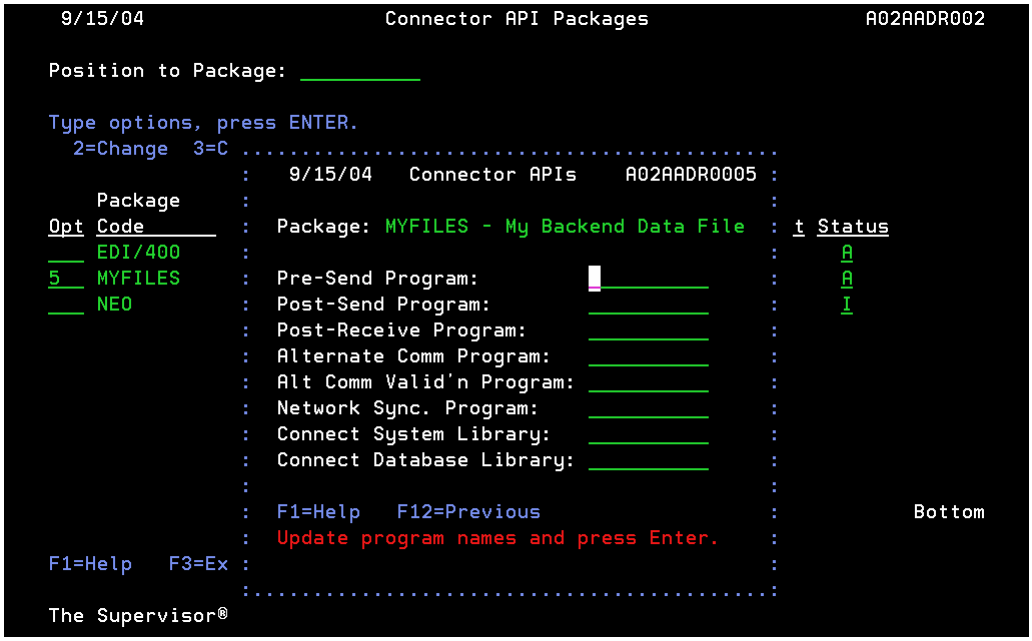
Package Code
This is the code that is entered when populating the schedule entry parameters. Enter a descriptive code.

Description
Enter a description for the connector package.

Default (Y, N)
The first default package will run when the if the package is blank in the schedule parameters. If all packages are "N", no package will run if the package is blank in the schedule parameters.

Status (A, I)
Actives or deactivates a connector package. Enter "A" for active or "I" for inactive. Only active packages will show while prompting an the packages during AutoComm schedule parameters entry.

Connector APIs



A Connector API is a program interface that connects AutoComm to your data-handling product. When AutoComm calls a connector API, it sends key information that could be used by your data-handling product to perform data processing or communications. Connector APIs may be created by your IT department for connecting to any product that includes data APIs.

From this display you can

- View connector API entries
- Create connector API entries
- Change connector API entries

Function Keys

F1 Help

F12 Close the window

CONNECTOR API PROGRAMMING TIPS

The parameter being sent is a single string containing multiple values in the formats listed for each connector API. The 4-character dictionary codes are contained in The Supervisor® dictionary, ADict. ADict may be defined in a connector API as an external data structure. You can create a data structure with fields that are defined like the dictionary fields and move the single *ENTRY parameter into the data structure without having to hard-code lengths of every inbound field. If a file specified in the Data Files is to be created, the file may be reproduced for connector API compiling purposes with the following command:

CRTPF FILE({library}/{file}) RCDLEN(256) MBR(&MBR)
*MAXMBRS(*NOMAX) SIZE(*NOMAX)*

FIELDS

Pre-Send Program

Before AutoComm does a send ("put"), it calls this API. This API is usually used to copy your product's electronic data into AutoComm's send data file (QTEMP/A02AABPF02 by default). Any data that is copied to this file at this time will be transmitted. At this point, AutoComm can store your product's session key for use after sending the data. Store the session key in the auto-created data area, QTEMP/SESSKEY. The parameters fed to your API follow.

Code	Description	Fr-To	Type(Len,Dec)
APPC	Application Code	1-3	A(3)
BATN	Batch Number	4-12	S(9,0)
SNRC	Send/Receive	13-13	A(1)
NTID	Network ID	14-15	A(2)
FWRP	Wrap Code	16-16	A(1)
FRCD	Record Length	17-19	S(3,0)
OBND	File/Library/Member	20-49	A(30)

Post-Send Program

After AutoComm does a send ("put"), it calls this API. This API is usually used to either take action based on the results of the transmission (passed/failed) or to communicate the transmission results to your product. This may require the session key that was stored during the pre-send process. The session key may be acquired from the data area QTEMP/SESSKEY. The parameters fed to your API follow.

Code	Description	Fr-To	Type(Len,Dec)
APPC	Application Code	1-3	A(3)
BATN	Batch Number	4-12	S(9,0)
SNRC	Send/Receive	13-13	A(1)
CSTS	Completion Status	14-15	A(2)
FBYT	Bytes sent	16-30	S(15,0)
TDUR	Time duration	31-37	S(7,0)

Post-Receive Program

After AutoComm does a receive ("get"), it calls this API. This API is usually used to copy received data from AutoComm's receive data file (QTEMP/A02AABPF01 by default) into your product. The parameters fed to your API follow.

Code	Description	Fr-To	Type(Len,Dec)
APPC	Application Code	1-3	A(3)
BATN	Batch Number	4-12	S(9,0)
SNRC	Send/Receive	13-13	A(1)
NTID	Network ID	14-15	A(2)
OBND	File/Library/Member	16-45	A(30)
FBYT	Bytes Received	46-60	S(15,0)
TDUR	Time duration	61-67	S(7,0)
FWRP	Wrap Code	68-68	A(1)
FRCD	Record Length	69-71	S(3,0)

Alternate Comm Program

AutoComm does not utilize modem communications. If needed, modem communications must be a function of your data-handling product. AutoComm can launch this API to call your product's communications API. The parameters fed to your API follow.

Code	Description	Fr-To	Type(Len,Dec)
APPC	Application Code	1-3	A(3)
BATN	Batch Number	4-12	S(9,0)
NTID	Network ID	13-14	A(2)
IVAL	Identifier Value	15-22	A(8)
TYPE	Session Type	23-23	A(1)

Alt Comm Valid'n Program

This API may be used to validate modem resources before launching the modem communications API. If the validation fails, a high-severity message is logged and the modem communications are suspended. The parameters fed to your API follow.

Code	Description	Fr-To	Type(Len,Dec)
APPC	Application Code	1-3	A(3)
BATN	Batch Number	4-12	S(9,0)
NTID	Network ID	13-14	A(2)

Network Sync. Program

This API is used to synchronize your product's network listing (if available) with AutoComm's network listing. The API is launched every time AutoComm's networks screen is displayed. Note that this option is not available in AutoComm™ *Signature*.

Connect System Library

This is the system library name of your application that the Connector APIs must communicate with. The library is added to the library list (after QTEMP) while AutoComm is running. Leave blank to not add a library to the library list.

Connect Database Library

This is the database library name of your application that the Connector APIs must communicate with. The library is added to the library list (after QTEMP) while AutoComm is running. Leave blank to not add a library to the library list.

PARAMETER DESCRIPTIONS

APPC Application Code A(3)
AutoComm's application code (usually "A02"). Used for logging messages to The Supervisor® from with a connector API.

BATN Batch Number S(9,0)
The batch number created by The Supervisor®. Used for logging messages to The Supervisor® from with a connector API.

CSTS Completion Status A(2)
Data sent without errors = "00". With errors = "99".

FBYT Bytes Sent/Received S(15,0)
Number of bytes sent or received.

FRCD Record Length S(3,0)
Either 80 or 132. Value entered in FTP Network Setup Parameters.

FWRP Wrap Code A(1)
Either "W" for wrapped or "U" for unwrapped. Value entered in FTP Network Setup Parameters.

IVAL Identifier Value A(8)
Used for the type of alternate communications. The value may be stored in AutoComm's network file during network synchronization. The value may be used by the alternate communications program to select which of your product's APIs to use for communications.

NTID Network ID A(2)
This is the network ID being executed.

OBND File/Library/Member A(30)
The value is in the following format:

Description	Fr-To	Type(Len)
File	1-10	A(10)
Library	11-20	A(10)
Member	21-30	A(10)

During pre-send, the values are discovered and come from the "Get send data from" in your data file definition ("A02AABPF02QTEMP " by default). During post-receive, the values are discovered and come from the "Put received data into" in your data file definition ("A02AABPF01QTEMP " by default).

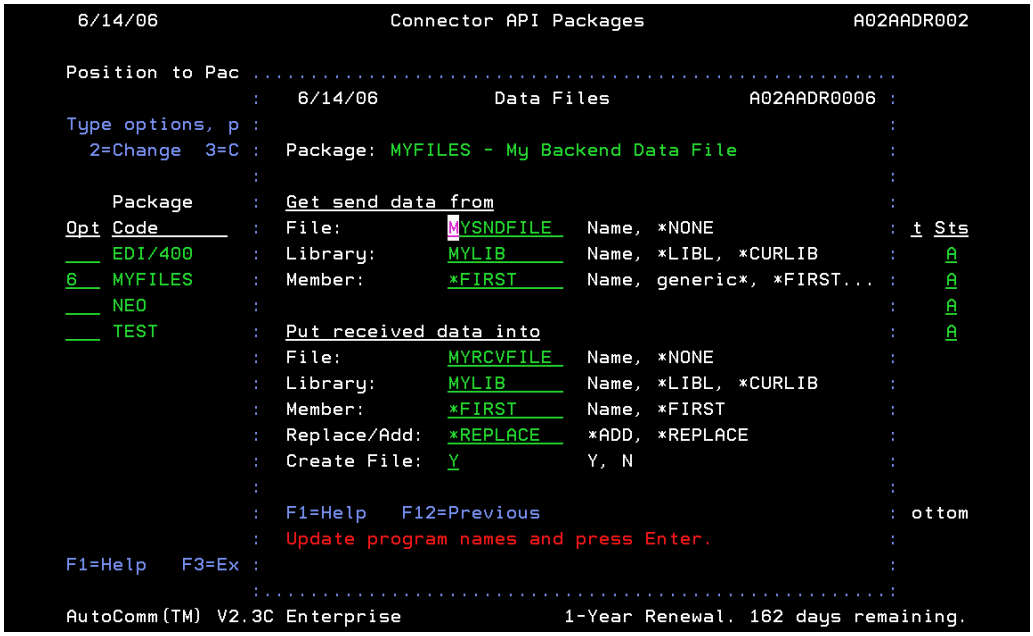
SNRC Send/Receive A(1)
"S" for pre/post-send and "R" for post-receive.

TDUR Time duration S(7,0)
Amount of time for communications.

TYPE Session Type A(1)
This is the session type that may be used by your product's modem communications API to determine the type of transmission to perform.

Alt Comm Send	Alt Comm Receive	Session Type
Yes	No	"1"
No	Yes	"2"
Yes	Yes	"3"

Data Files



The purpose of the data files window is to specify the files for received data to be copied into and sent data to be acquired from.

From this display you can

- View data file information
- Change data file information

Function Keys

F1 Help

F12 Close the window

FIELDS

Get send data from...
Specify the file to put in the OutBox.

File (Name, *NONE)
The name of the file that will be copied to the OutBox. *NONE is used when another process copies the files to the OutBox.

Library (Name, *LIBL, *CURLIB)
The library where the file exists. Enter "*LIBL" if the file is in the job description's or user's library list. Enter "*CURLIB" to specify the job's library. If the file does not exist, QGPL is used.

Member (Name, generic*, *FIRST, *LAST)
The name of the member in the file to send. Enter "*FIRST" to send the first member in a date-ordered list. Enter "*LAST" to send the last member in a date-ordered list. Specify the starting characters of the member name followed by an "*" to send the first member in the name ordered list that starts with the specified characters.

Put received data into...
Specify the file to copy InBox data into.

File (Name, *NONE)
The name of the file to copy InBox data into. The file must exist when Create File is "N". If the file does not exist, the received data will be copied to the /AutoComm/{Network ID}Errors directory and a high severity message will be logged. The file may be either a source physical file or a program-described physical file. Source physical files with multiple data fields are not supported. If the database file is a source physical file, a sequence number and a date stamp of zeros is added to each record as it is copied to the database file member.

Library (Name, *LIBL, *CURLIB)
The library where the file exists. Enter "*LIBL" if the file is in the job description's or user's library list. "*LIBL" is not valid when Create File is "Y". Enter "*CURLIB" to specify the job's library. If the file does not exist, QGPL is used.

Member (Name, *FIRST)
The name of the member in the file to receive the data. Enter "*FIRST" to copy the data to the first member in a date-ordered list. If the member does not exist, it is created.

Replace/Add (*ADD, *REPLACE)
Specifies whether the copy operation replaces, adds, or fails to copy the records if the object specified on the TOMBR parameter already exists. If the object does not exist, it is created. Enter "*ADD" to add the copied records to the end of the existing records. Enter "*REPLACE" to replace the existing records with the copied records. When multiple files are received, the first file's records will follow the value entered (*ADD or *REPLACE). The remaining files' records are will follow the *ADD rule.

Create File (Y, N)
Enter "Y" to create the file in the specified library or "N" if the file already exists. The file is created during the receive process and will have the following attributes.

- One member with the same name as the file.
- Single field in the member with the same name as the file and a record length of 256 characters.
- Maximum members and size is *NOMAX.

DEFAULTS

These files are created during transmission in QTEMP and may be used by connector APIs to copy data from or into.

Get send data from

File: A02AABPF02

Library: QTEMP

Member: *FIRST

Put received data into

File: A02AABPF01

Library: QTEMP

Member: *FIRST

Replace/Add: *REPLACE

Create File: Y

Calling AutoComm from other applications

It's easy to call AutoComm from another application. Just enter one of the following command from a command line or in your CL application:

Run Interactively:

CALL PGM(A00APIPG01) PARM('<Application Code>' '<Network ID>' '<Send>' '<Receive>')

Submit to Batch:

CALL PGM(A00BATPG01) PARM('<Application Code>' '<Job Description>' '<Job Queue>' '<Network ID>' '<Send>' '<Receive>')

Where the parameters are defined as follows:

Parameter	Description	Value
Application Code	The 3-character code for the application (AutoComm = "A02").	A02
Job Description	Used in the Submit to Batch API. Enter a valid OS400 Job Description. Be sure the job description includes the necessary library list to support AutoComm.	Go to a command line and type "WRKJOBD *ALL" to view all the available job descriptions.
Job Queue	Used in the Submit to Batch API. Enter a valid OS400 Job Queue where the job is to be queued for processing.	Go to a command line and type "WRKJOBQ *ALL" to view all the available job queues.
Network ID	Enter a valid 2-character network id to communicate with. AutoComm Signature supports one network.	Go to your FTP Network Setup to see a list of networks.
Send	Enter "Y" for Yes (execute send) or "N" for No (do not execute send).	Y, N
Receive	Enter "Y" for Yes (execute receive) or "N" for No (do not execute receive).	Y, N

Appendix

A

Special Notices

This publication is intended for the customer licensing the software and may not be duplicated in any form for distribution or sales to entities that are not licensing the software from EDI Pros, Inc.

AUTOCOMM™ is a trademark of EDI Pros, Inc.

THE SUPERVISOR® is a registered trademark of EDI Pros, Inc.

IBM® is a registered trademark of the International Business Machines Corporation of the United States and/or other countries.