



# bTrade TDAccess Release Notes

Release 2.x

© 2003 bTrade  
All rights reserved.  
Document 2003.R001.01  
Printed in the USA.

CONFIDENTIAL. All rights reserved. This document, including any writing, drawings, notes, or verbal representation made or shown in the course of this communication is confidential and proprietary to bTrade. No part of the materials included in this communication should be: a) reproduced; b) published in any form by any means, electronic or mechanical, including photocopy or information storage or retrieval system; or c) disclosed to third parties, without the express written authorization of bTrade.

# Contents

Contents .....	3
1 Purpose.....	6
2 Release Notes.....	7
2.1.1 Build 2.2.014 01/14/2003 .....	7
2.1.2 Build 2.2.013 01/08/2003 .....	7
2.1.3 Build 2.2.012d 12/18/2002 .....	7
2.1.4 Build 2.2.012 12/04/2002 .....	8
2.1.5 Build 2.2.011 11/08/2002 .....	12
2.1.6 Build 2.2.010 10/24/2002 .....	13
2.1.7 Build 2.2.009 10/16/2002 .....	14
2.1.8 Build 2.2.008 10/08/2002 .....	14
2.1.9 Build 2.2.007 10/03/2002 .....	15
2.1.10 Build 2.2.006 08/21/2002 .....	16
2.1.11 Build 2.2.005 07/24/2002 .....	17
2.1.12 Build 2.2.004 07/11/2002 .....	17
2.1.13 Build 2.2.003 07/02/2002 .....	17
2.1.14 Build 2.2.002 06/20/2002 .....	18
2.1.15 Build 2.2.001 05/30/2002 .....	18
2.1.16 Build 2.1.008 05/13/2002 .....	19
2.1.17 Build 2.1.007 04/30/2002 .....	19
2.1.18 Build 2.1.006 01/28/2002 .....	21
2.1.19 Build 2.1.005 01/16/2002 .....	21
2.1.20 Build 2.1.005 12/27/2001 .....	22
2.1.21 Build 2.1.004 12/03/2001 .....	22
2.1.22 Build 2.1.003 10/23/2001 .....	22
2.1.23 Build 2.1.002 10/15/2001 .....	24
2.1.24 Build 2.1.001 09/14/2001 .....	24
2.1.25 Build 2.18 07/13/2001 .....	25
2.1.26 Build 2.17 06/15/2001 .....	25
2.1.27 Build 2.16 06/15/2001 .....	25
2.1.28 Build 2.15 05/18/2001 .....	25
2.1.29 Build 2.14 05/03/2001 .....	25
2.1.30 Build 2.13 04/26/2001 .....	25
2.1.31 Build 2.12 04/24/2001 .....	26
2.1.32 Build 2.11 03/28/2001 .....	26
2.1.33 Build 2.10 03/24/2001 .....	26
2.1.34 Build 2.09 03/21/2001 .....	26
2.1.35 Build 2.08 03/14/2001 .....	26
2.1.36 Build 2.07 03/13/2001 .....	26
2.1.37 Build 2.06 03/12/2001 .....	26
2.1.38 Build 2.05 03/10/2001 .....	27

**bTrade TDAccess**  
Release Notes

---

2.1.39	Build 2.04 03/06/2001	27
2.1.40	Build 2.03 03/01/2001	27
2.1.41	Build 2.02 02/21/2001	28
2.1.42	Build 2.01 02/09/2001	28
2.1.43	Build 1.52 12/13/2001	28
2.1.44	Build 1.51 11/05/2001	29
2.1.45	Build 1.50 10/20/2001	29
2.1.46	Build 1.49 10/10/2001	29
2.1.47	Build 1.48 09/23/2001	31
2.1.48	Build 1.47 09/07/2001	31
2.1.49	Build 1.46 08/28/2001	31
2.1.50	Build 1.45 08/27/2001	32
2.1.51	Build 1.44 08/24/2001	32
2.1.52	Build 1.43 08/22/2001	32
2.1.53	Build 1.42 08/14/2001	32
2.1.54	Build 1.41 07/31/2001	32
2.1.55	Build 1.40 07/27/2001	32
2.1.56	Build 1.39 07/15/2001	33
2.1.57	Build 1.38 01/25/2001	33
2.1.58	Build 1.37 12/08/2000	33
2.1.59	Build 1.36 08/03/2000	34
2.1.60	Build 1.35 06/19/2000	35
2.1.61	Build 1.34 06/06/2000	35
2.1.62	Build 1.33 05/15/2000	36
2.1.63	Build 1.32 03/27/2000	36
2.1.64	Build 1.31 02/01/2000	37
2.1.65	Build 1.30 01/13/2000	38
2.1.66	Build 1.29 12/30/1999	38
2.1.67	Build 1.28 12/03/1999	38
2.1.68	Build 1.27 10/20/1999	39
2.1.69	Build 1.26 09/27/1999	42
2.1.70	Build 1.25 08/09/1999	43
2.1.71	Build 1.24 06/25/1999	43
2.1.72	Build 1.23 06/10/1999	43
2.1.73	Build 1.22 04/13/1999	44
2.1.74	Build 1.21 02/18/1999	44
2.1.75	Build 1.20 01/20/1999	45
2.1.76	Build 1.19 11/18/1998	45
2.1.77	Build 1.18 11/15/1998	45
2.1.78	Build 1.17 10/09/1998	46
2.1.79	Build 1.16 10/05/1998	46
2.1.80	Build 1.15 09/30/1998	46
2.1.81	Build 1.14 09/24/1998	47
2.1.82	Build 1.13 09/21/1998	48
2.1.83	Build 1.12 09/12/1998	48
2.1.84	Build 1.11 09/07/1998	48

**bTrade TDAccess**  
*Release Notes*

---

2.1.85	Build 1.10	09/02/1998	48
2.1.86	Build 1.9	08/24/1998	48
2.1.87	Build 1.8	07/27/1998	49
2.1.88	Build 1.7	07/21/1998	49
2.1.89	Build 1.6	07/17/1998	49
2.1.90	Build 1.5	07/06/1998	49
2.1.91	Build 1.4	05/22/1998	50
2.1.92	Build 1.3	04/15/1998	50
2.1.93	Build 1.2	03/23/1998	50
2.1.94	Build 1.1	03/18/1998	50
2.1.95	Build 1.0.0	03/09/1998	50

# 1 Purpose

The purpose of this document is to serve as a reference guide for the different releases of TDAccess 2.x.

## 2 Release Notes

The following are release notes for TDAccess 2.x Client from bTrade, Inc.

### 2.1.1 Build 2.2.014 01/14/2003

- Fixed another problem impacting the ability to send multiple AS2 messages per a single http connection to the server. Under certain circumstances the client would think it was still connected to the server when it was not, leading to a 'send' error.
- Fixed a problem with the GENERIC FTP network style which caused files to be sent to the FTP server in the wrong mode (ASCII or binary).

### 2.1.2 Build 2.2.013 01/08/2003

- Added support for images using the TIF format (TDImage).
- Fixed a problem with the EMail Address screen which caused the entered data to be forgotten in subsequent program runs.
- Fixed a problem with the Configuration / Setup Logging Options screen which caused the entered data to be forgotten in subsequent program runs.
- Modified the HTTP library to terminate the connection if a server responds as an HTTP 1.0 server, or if it includes the connection: close header in it's response. Previously, the HTTP library waited for the server to close the connection, and if the server waited to close the connection, the connection state became unknown to the client.
- Added support for a new method of managing encrypted password data used by TDNgine.
- This release contains TDCompress Build 0462.

### 2.1.3 Build 2.2.012d 12/18/2002

#### TDAccess Client:

- Fixed a problem writing the response log where under certain combinations of parameters the LOCALID() filename was incorrect. Also, will now create an empty response log if, during a receive transfer, no files are received.
- Fixed a problem with sending multiple AS2 messages in a single session. When executing a SendEDI transfer with X12 data, the message size was being incorrectly translated to the Content-Length header for the second and subsequent messages, due to a problem with the TDCompress component.
- Fixed a problem with the Failure Notification option for AS1/AS2/SMIME network styles when a duplicate messageID was encountered. This was causing multiple failure notification messages to be generated, instead of just one message. Also, cleaned up the Setup Retry and Notification screen - some of the screen literals were slightly truncated at some screen resolutions.
- Fixed a problem with the Fedex network style which was causing the APRF list to be duplicated in the Send Transfer APRF combo-box.
- Changed the way the original filename is record in the MIME header of AS1 and AS2 messages for the payload bodypart, to handle the file-naming requirements of some other vendors' AS2 servers which do not quite follow standards.

- Changed the way the Delete-after-Send option works when doing SendEDI for network styles which split out each interchange to a separate file. Now, the original file is not deleted until each interchange is successfully delivered.

#### **TDAccess Server:**

- Changed the server's file-naming convention (when retainOriginalNames = N) to include the AS2-To MIME header in the filename, to facilitate the multiple-personality server feature.

### **2.1.4 Build 2.2.012 12/04/2002**

#### **TDAccess Client:**

- Modified the Windows installation scripts so TDClient and TDServer use the same directory (the tdclient runtime sub-directory) for runtime files by default.
- Modified the Windows installation scripts to allow the AS2 Name to be specified. This value is used by both the TDClient and TDServer when doing AS2 communications.
- Modified the way Trading Partner information is obtained from the TDManager database so that the AS2 Name value for a trading partner is obtained from the User Name field on the Participant's Comm Profile screen. In addition, the HTTP Proxy user-name and password values are now taken from the Network Logon and Network Password fields on the Participant's Enabling screen.
- Fixed the STERLING-ENTERPRISE network style so that the SendAscii and ReceiveAscii flag values determine whether an EBCDIC-to-ASCII and ASCII-to-EBCDIC data conversion is performed for TDclients running on OS400 or MVS.
- Corrected a problem where AS1 and AS2 MDN processing reported "error -3" which is actually not an error condition (it was caused by the fact that an MDN did not exist yet for the given message, which is expected in most cases). The "error" message no longer is displayed.
- Added Failure Notification capability.  
This capability will generate and transmit a notification message whenever a transfer fails. The notification message is a multipart MIME message, and may be transmitted using any of the available (active) network instances in your tdclient.ini file. For example, the notification message may be transmitted via EMail, or using AS1 or AS2, or any of the FTP styles available in your tdclient.ini file. Since the notification message is a multipart MIME message, it is most easily viewed using a mail reader program, but it may also be viewed as plain text file.

To configure the notification options in the GUI, navigate to the Configuration / Setup Retry and Notification Options screen.

To configure the notification options from the command-line, use the following new keywords:

**NOTIFY\_CMDLINE\_USERS**= Enables notification messages for command-line users (only).  
**NOTIFY\_GUI\_USERS**= Enables notification messages for GUI users.  
**NOTIFY\_AFTER\_LAST\_RETRY**= Send failure message only after last retry  
**NOTIFY\_AFTER\_ITH\_RETRY**= Send failure message every 'i' retries.  
**NOTIFY\_ITH\_RETRY**= Specifies the value of 'i' for NOTIFY\_AFTER\_ITH\_RETRY.  
**NOTIFY\_INCLUDE\_LOG\_FILES**= Include any active log files with the failure message.

**NOTIFY\_INCLUDE\_CONFIG\_FILES=** Include all tdclient 'ini' files with failure message.

**NOTIFY\_INCLUDE\_TEMP\_FILES=** Include all relevant temporary files (in temp sub-directory) with the failure message.

**NOTIFY\_MAX\_MSG\_SIZE=** Specifies the maximum message size, in kilobytes.

**NOTIFY\_NETWORK=** Specifies which network instance is to be used to transmit the notification message. Any active network instance may be specified.

In addition to these values, you must configure the transfer used by the notification process to send the message. This is simply accomplished in the GUI by clicking the 'Configure notification delivery' button, causing the 'Setup SEND TDCLIENT NOTIFICATION' screen to display. You only need to provide two pieces of information to configure the transfer. Depending on the network selected, they are the destination address (mailbox for FTP, trading-partner address book entry for EMail, AS1, AS2), and the 'subject' (EMail, AS1, AS2), or 'class' (most FTP styles) or 'APRF' (some FTP styles).

Configuring the notification transfer using the command-line can be done with a separate run of the command-line program. For example:

```
tdclientc RESET NETWORK=EMAIL "TRANSFER=(NAME='SEND TDCLIENT NOTIFICATION'  
SEND=somefile SEND_VERIFY=N SENDSUBJECT='Failure!' SENDUSERID=myTPBookEntry)"  
SAVE_ONLY
```

Noteworthy points here are:

- (1) The NETWORK is set to the network instance to be used for transmission of the failure notification message. Here, EMAIL is being used.
- (2) The name of the transfer must be SEND TDCLIENT NOTIFICATION.
- (3) Although only the sendsubject and senduserid are actually used by the notification process (all other values are controlled by the program), you must include the SEND= keyword to define a 'legal' transfer. Use the SEND\_VERIFY=N keyword to allow the specification of any filename. The filename will not be used, since the notification process creates it's own file containing the notification message and uses that filename instead.
- (4) For other network styles, use the appropriate keyword in place of SENDSUBJECT=. For example, SENDCLASS=, or SENDAPRF= may be used for FTP styles.
- (5) The SAVE\_ONLY keyword is helpful since we do not want to save the transfer and not execute it.

- Fixed the SMIME network style to support wild-carding of filenames when encrypting or decrypting.
- Fixed problems with the Session History and Session Detail features in the GUI which caused, under certain odd conditions, the second-most-previous session's details to be omitted from the display.
- Extended the value-substitution capabilities for AS2 http URL addressing. The available substitutions now include:

**%filename%** - replaced with the base filename of file being transmitted

**%fromEdiName%** - replaced with the EDI Name of sender

**%fromEdiQual%** - replaced with the EDI Qualifier of sender

**%fromAS2Name%** - replaced with the AS2 Name of the sender

**%toEdiName%** - replaced with the EDI Name of receiver

**%toEdiQual%** - replaced with the EDI Qualifier of receiver

**%toAS2Name%** - replaced with the AS2 Name of the receiver

For example, if in the Trading Partner Address book the AS2 URI for a trading partner is `https://MyPartner.com:28080/%fromAS2Name%_file_is_%fileName%`, then when you send to this partner, the `%fromAS2Name%` and `%fileName%` values will be replaced with the corresponding values in effect for the transfer being executed. That is, the current AS2 Name of the sender and the actual filename being packaged and sent will appear in the http address sent to the receiving AS2 server.

- Modified the way the FTP change-working-directory command is used, so that two separate 'CD' commands are issued if both a mailbox and a 'class' or 'APRF' are specified. Previously, the mailbox and class were merged using '/' as a separator, and this merged string was used in a single 'CD' operation. However, some FTP servers do not recognize the '/' as a valid separator character. This problem is overcome by first doing a 'CD' into the mailbox, and then a 'CD' into the class.
- Both TDClient and TDServer will now successfully process SMIME, AS1 and AS2 messages encrypted with any certificate for which the certificate store contains a private key. The certificate store is either the TDManager database, or the runtime files, depending on how TDClient and TDServer are configured. Previously, the TDClient and TDServer would only process files encrypted for the certificate corresponding to the single EDI Name specified in the `tdclient.ini` file or `tdserver.cfg` file. This new capability provides for full support of multiple personalities in both the TDClient and TDServer when receiving SMIME'd messages.
- Added the ability to set the TDClient logging facilities from the GUI and from the command-line. To set the various logging options in the GUI, navigate to the Configuration / Setup Logging Options screen. The allowed logging levels are from 0 (off) to 6 (maximum). You may optionally select the 'Flush after every write' checkbox for each log option. This forces the log file contents to be current (and read-able) at each step in the program execution.  
The following logging options are available:

**Overall program run** - Produces file `ea2k.log` in the TDClient root directory, containing details of the entire program run.

**'Ini' file processing** - Produces file `eaini.log`, containing details of the start-up processing of the `tdclient.ini` file.

**Transfer processing** - Produces file `temp/eaxfer.log`, containing details of the last send or receive transfer, query\_list, or audit report request.

**Communications session** - Produces file `temp/eacomm.log`, containing details of the last FTP, AS1 (SMTP/POP + S/MIME), or AS2 (HTTP + S/MIME) communications session.

**SSL session** - Written into the `temp/eacomm.log` (in addition to the Communications session text), contains very low level details of the SSL encrypted communications session when using SSL with FTP or HTTP.

**Memory usage** - Produces the file `eamem.log` in the TDClient root directory, containing details of memory allocation/deallocation during the program run.

Please be aware of two facts about logging:

(1) When set to the maximum log level (6) a very large amount (megabytes) of logged text can result. This will slow down program operation.

(2) Setting the 'flush after every write' option will also slow down program operation (sometimes dramatically). So, while logging is useful to isolate configuration or operational problems, you probably do not want to run with extensive logging turned on all the time.

To setup the logging options from the command line, the following new command-line keywords are now available:

<b>LOG_EASYACC=</b>	Controls the Overall program run log.
<b>LOG_INI=</b>	Controls the 'Ini' file processing log.
<b>LOG_XFER=</b>	Controls the Transfer processing log
<b>LOG_FTP=</b>	Controls the Communications session log.
<b>LOG_SOCKET=</b>	Controls the SSL session log.
<b>LOG_MEM=</b>	Controls the Memory usage log.

For all of these keywords, the allowed values are 0 (off) to 6 (maximum). In addition, values of 'N' (off) and 'Y' (corresponds to numeric value 3) are supported. Finally, to enable 'flush after every write' logging, prefix the value with a minus sign. For example, LOG\_XFER=-6 means turn on maximum logging and flush after every write.

- Added the ability to specify the local host IP address. This is useful for machines which are multi-hosted, when operating in an environment using Network Address Translation (NAT). To specify the IP address of the local host in the GUI, navigate to the Configuration / Setup Firewall/Proxy Options screen and fill in the Local Host IP Address field.  
To accomplish from the command-line, use the new command-line keyword:

**LOCAL\_HOST\_IP\_ADDRESS=**

- Fixed a problem in the GUI for AS1/AS2/SMIME networks:  
On the Export User Certificates screen the drop-down list of trading partners for the Transmit PKCS7 message option was not being populated due to bug. This has been fixed.
- Added the ability to specify the user's AS2Name value using the new command-line keyword:

**AS2NAME=** This value appears in the 'AS2-To:' field of AS2 messages.

- Added the ability to specify the AS2Name values for trading partner entries from the command-line, using the new command-line keywords AS2NAME1= and AS2NAME2=. These keywords apply only in the context of creating a trading partner address book entry.  
For example:

```
TPBOOK=(TPNAME=My-Trading-Partners-Name
NETWORK1=AS2
MAILBOX1=http://my-Trading-Partners-Address
AS2NAME1=My-Trading-Partner-AS2Name
HTTP_USERID1=My-Trading-Partners-Server-UserId
HTTP_PASSWD1=My-Trading-Partners-Server-Password
SERVER_EDINAME1=EDIName-of-certificate-for-TP
NETWORK2=AS1
MAILBOX2=my-trading-partners-email-address@his-company.com
AS2NAME2=<not applicable since this is an AS1 instance...>
HTTP_USERID2=<ditto>
HTTP_PASSWD2=<ditto>
SERVER_EDINAME2=EDIName-of-certificate-for-TP)
```

- Added the ability to retrieve MDN-based audit logs for specific sender and receiver names (AS1/AS2/SMIME network styles). In the GUI, the Retrieve Audit Logs screen allows one to specify a sender and/or receiver (trading-partner address book entries). When these fields are specified, then only MDN records with matching sender and/or

receiver fields are returned and displayed. This is accomplished in the command-line by using the new command-line keywords:

**AUDIT\_SENDER=**  
**AUDIT\_RECEIVER=**

with your RECEIVE\_AUDIT\_LOGS program runs, and these new keywords work in addition to the existing AUDIT\_FILE=, AUDIT\_START\_DATE= and AUDIT\_END\_DATE= keywords.

- Fixed a problem using the database for certificates when using the IGN network style with SSL. The "SSL negotiation error" was due to a problem accessing the required certificate, and this has been fixed.
- Fixed a problem with the Windows InstallShield for TDAccess and TDPeer where the GUI configuration file (tdclient.znc) was not being updated when the product was installed into an existing product directory.
- Fixed a problem with the XFER class and some of its sub-classes which caused a program crash under some circumstances - some of the member variables were not being initialized properly, and if bad data was presented to the constructor, then when the destructor was called, sometimes the uninitialized member data caused problems (like invalid deallocations) leading to program crashes.
- Fixed an odd timing problem which led to a SQL error when an AS2 client and AS2 server located on the same machine and sharing the same TDManager database are involved in a message exchange. The client checks to see if an MDN exists, and then adds an MDN, but the add failed because the server had already added an MDN for the same message ID. This has been observed using an MS Access database, and is apparently due to the 'commit' lagging behind. This has been fixed by checking to see if the record exists when the add fails, and doing an update if it does.

#### **TDAccess Server:**

- Added the ability to specify the local host IP address which is useful for machines which are multi-hosted, when operating in an environment using Network Address Translation (NAT). This option is set using the localhostIP= line in the \*Firewall and Proxy configuration\* section of the tdserver.cfg file.
- Corrected a problem where AS2 MDN processing reported "error -3" which is actually not an error condition (it was caused by the fact that an MDN did not exist yet for the given message, which is expected in most cases). The "error" message no longer is displayed.
- Added true multiple-personality support. The TDServer will now correctly process incoming messages which have been encrypted for any certificate for which a private key is available (in the runtimes files, or in the TDManager database if not using runtimes).
- Modified the Windows installation scripts so TDClient and TDServer use the same directory (the tdclient runtime sub-directory) for runtime files by default.

### **2.1.5 Build 2.2.011 11/08/2002**

#### **TDAccess Client:**

- Added support for firewalls using static network address translation (static NAT), for active (non-passive) FTP communications. This is setup via the Setup Firewall/Proxy

screen; enter the IP address or domain name of the firewall, and select the appropriate firewall type. The IP address of the firewall is passed to the server in the PORT command, allowing active FTP to work with the static NAT. To set this up in the command-line or library versions, use command-line keywords **FIREWALL\_TYPE=3** with **FIREWALL\_HOSTIP=<your firewall IP address or domain name>**.

- Fixed a problem sending from OS400 and MVS platforms to Windows platforms using any of the GENERIC FTP styles. The server-side filename was not properly formed from Windows point of view, resulting in files being overwritten or failing to be transmitted.
- This release contains TDCompress Build 0461.

#### **TDAccess Server:**

- For UNIX platforms, made the installation of the TDServer optional.

### **2.1.6 Build 2.2.010 10/24/2002**

#### **TDAccess Client:**

- AS1/AS2/SMIME network styles: Relaxed the requirement for a certificate for the sender. In previous releases, the 'sender' always had to have a certificate. Now, the sender's cert is required only if the message is to be signed.
- AS1/AS2/SMIME network styles: For SendEDI transfers, will now use the trading-partner address book entry for the Sender (as well as the Receiver). Thus, one may create address book entries for the Sender value(s) in the EDI data, and specify which certificate(s) is/are to be used for signing each interchange (envelope) to be sent.

This is a beginning for the support of a "multiple personality" client.

- Changed the GENERIC\* network styles so that no 'userid' or 'class' is required for any type of Send transfer.
- Added a new network style, STERLING-ENTERPRISE, to enable FTP sessions with the Sterling CONNECT:Enterprise FTP server. This style is very similar to the STERLING-COMMERCE style, but handles small differences between the two server's variants of FTP.
- Fixed an inefficiency in the utility method which renames files for Windows platforms.
- Fixed a bug on some Query Mailbox screens where the File/Close menu item was active, allowing the screen to be closed during the middle of a Query Mailbox session, causing TDClient to crash.
- Modified how the FTP mode (ASCII or binary) for a Query Download is determined for certain network styles - now the network's default receive parms (ASCII=N or Y) are used.
- Changed the Configuration/Network display for SMIME-style network instances to correctly show that the "Archive EDI Name" field is not relevant since this information is obtained from the Trading Partner Address Book.

#### **TDAccess Server:**

- No changes.

## **2.1.7 Build 2.2.009 10/16/2002**

### **TDAccess Client:**

- Added text decanonicalization when transporting AS1/AS2 or SMIME messages to UNIX platforms from other types of platforms. Prior to this, text documents retained the carriage return/line feed pair at the end of each line of text, instead of ending in a single line feed per UNIX text file representation.
- Added SIGNEDMDN() field to the response log to indicate if an AS1/AS2/SMIME message is requesting a signed MDN or not. Note that, when sending an AS1/AS2 message or constructing an SMIME message, the MESSAGEDIGEST() field in the response log only contains a value when a signed MDN is being requested.
- Changed the message the TDClient command-line program outputs as it is finishing a program run. It was outputting "Task completed with failure code <some number>" even when no error occurred (as of changes made in version 2.2.006). Now it outputs "Program run completed successfully!" when successful, and "Task completed with response code <some number>" when a problem occurred.
- Fixed a problem with the GUI "Delay Start" functionality in the Start Time window when running a transfer. When a deferred start was selected, the transfer ran immediately anyway.
- Fixed a problem with SSL negotiation (https and SFTP) when using a database for certificates.
- More changes to handle problems when using the TDManager database for certificates and trading-partner-address-book information.
- Fixed a problem with the MIME parser which caused it to overwrite memory in a very obscure case (which was caught during testing).

### **TDAccess Server:**

- Changed the default value of the retainOriginalFileNames keyword to false, to be consistent with current user settings.
- Changed the default value of the autoExtLen keyword to 1 (from 0) to handle a condition on AS400, which did not like a default of 0.

## **2.1.8 Build 2.2.008 10/08/2002**

### **TDAccess Client:**

- (AS1/AS2/SMIME network styles) Added to the Export User Certificates screen the ability to specify whether or not to base64-encode the certificate when exporting the certificate to a file.
- AS1/AS2/SMIME network styles only) Added to the command-line and API versions the ability to export certificates. One may export to a file, or one may choose to send a 'certs-only' message using a trading partner address book entry. The new command-line keywords to support these features are:

**EXPORT\_CERTS\_TO\_FILE=** - Specifies a fully-qualified file name into which TDClient will export the public certificate for the configured user. The configured user is identified by the EDINAME entry in the [IDENTIFY] section of the tdclient.ini file.

**FILTER\_EXPORTED\_CERTS=** - (Y or N) Specifies whether the exported certificate will

be filtered (base64 encoded) or not.

Applies only in conjunction with EXPORT\_CERTS\_TO\_FILE=.

**EXPORT\_CERTS\_TO\_TP=** - (Trading partner address book entry) Specifies that a PKCS-7 'certs-only' message is to be constructed containing the public certificate for the configured user, and then transmitted to the specified trading partner as specified in the trading partner address book. The configured user is identified by the EDINAME entry in the [IDENTIFY] section of the tdclient.ini file.

- Fixed a problem when sending multiple files using AS2 to an HTTP 1.0 AS2 server. The HTTP 1.0 protocol does not support persistent connections, and TDClient was not recognizing that the server was ending the session/connection after each file (AS2 message) was sent. This has been fixed so that the TDClient recognizes that it needs to automatically re-connect to the server in those cases where the session is ended by the server. Note that HTTP 1.1 servers do support persistent connections, so multiple files can be sent and multiple responses received all in a single session. This is more efficient, since the connect steps (including SSL negotiation) do not have to be repeated for each AS2 message file transmitted.
- This release includes TDCompress Build 0459.

#### **TDServer:**

- No changes

### **2.1.9 Build 2.2.007 10/03/2002**

#### **TDAccess Client:**

- Added the ability to cancel an HTTP-based (AS2) transfer from the command-line (only). This is done by creating a file named 'cancel.fil' in the client's temp sub-directory. This will cause the client to terminate it's connection with the server and end processing, if it is able to do so.  
Note this functionality is the same as that available for FTP-based transfers.
- Fixed a problem associated with obtaining certificates from a TDManager database during SSL negotiations with an FTP or HTTPS server.
- Improved the error reporting for AS1/AS2/SMIME transfers where the receiver's trading partner address book entry is missing. A transfer's execution should be terminated when no trading partner address book entry is found for these network styles. Prior to this change, the transfer continued to execute until one of several different error conditions occurred, generating error messages somewhat unrelated to the root cause of the problem.
- Fixed a problem when the trading partner address book data comes from a database instead of the local .ini file. The problem occurred only for AS2 style communications and resulted in the AS2-To name being absent in the created AS2 message.
- Fixed a problem with the use of '@' character in the login userId to navigate certain FTP Proxy Servers. During Query Mailbox downloads a second, unnecessary, login attempt was made using an incorrect login userId. The problem stemmed from an error in the special processing required to support the '@' character in the login userId. This problem was reported primarily by users of the FEDEX network style, apparently because they use this type of Proxy Server more predominantly than the general population of TDAccess users.
- Corrected a problem in 'receive' transfers where the receive filename is specified as a directory. The intent is to allow the specification of a directory, rather than a

- complete file-spec, to indicate that the original filename or server filename is to be retained and used to name the received file(s). This was not working for the EMAIL and AS1 network styles, but has been fixed in this release.
- Fixed a bug in 'receive' transfers for AS1 and EMAIL networks styles concerning the email-address filter. Now, one may specify this filter and only messages sent from the specified email-address will be received.
  - Changed the behaviour of the AS1 and EMAIL network styles in the way they respond to corrupted or un-processable messages on the mail server. Now, if a corrupted or un-processable message is detected during a Query Mailbox (GUI) or QUERY\_LIST (command-line or API) operation, the message is included in the list with the bad fields tagged with a value of "unknown". If a corrupt or un-processable message is received, it is moved the error sub-directory. Formerly, when such a message was detected, the client simply stopped processing and reported an error.
  - This release includes TDCompress Build 0458.

#### **TDServer:**

- Fixed a problem with the output file-naming when sending from OS400 to Windows. If the OS400 file containing '\*' or '/' (as in \*CURLIB/E0000001), then the Windows TDServer would fail to rename the payload into the ediDir directory. This is fixed by substituting '\_' (underscore) for characters considered 'illegal' by the Windows file-naming convention.
- Fixed a problem on AIX which caused a large amount of information to be dumped to the terminal window (stdout) during the shutdown of the TDServer (via a signal 16), caused by a failure to sleep within the loop which waits for child processing threads to complete.
- Added TDServer features to allow for the retention of the original filename associated with a received payload. If an AS2 message is received with a payload which contains in it's header information specifying it's original filename, then the TDServer will name the output payload file using this name. See the new tdserver.cfg file entry 'retainOriginalFileName=' keyword. To avoid overwriting existing files, newly received files may be 'auto-extended' using a incrementing numeric file extension. See the new tdserver.cfg file entry 'autoExtLen=' keyword.
- Improved the error reporting when the TDServer config file has incompletely specified database information.

#### **2.1.10 Build 2.2.006 08/21/2002**

- For Windows platforms, now including 2 required DLLs in the install package: msvcr7.dll and msvcp60.dll.
- Fixed a problem in the btdao library which impacted the integration of TDServer with TDNgine.
- Added support for the ENABLE\_ALIAS\_PROBE= keyword (IGN styles only):

**ENABLE\_ALIAS\_PROBE=** Y or N. IGN (FTP) network style only. If Y, then when sending EDI data to the IGM I/E FTP Gateway, the SITE EDIALIASPROBE 1 command is issued allowing the transmission of EDI files containing multiple envelopes, some of which are addressed to regular I/E mailboxes, and some of which are addressed to mailboxes via Alias Table(s). When set to N, then the SITE EDIALIASONLY 1 command is issued, as in previous releases of TDClient.

- Modified the argument parser used by the command-line and API versions of TDAccess to check for nested delimiters.
- Modified the return codes reported for FTP session errors to include the entire 3-digit response, instead of just the first digit of the response.
- Changed Database Setup screen to allow the database type value to be entered.
- Added support for the EDS sub-style for handling the EDS\*ELIT variant of SSL.
- Added support for obtaining certificate information directly from TDManager database when needed for SSL negotiations with an FTP or HTTPS server.

#### **2.1.11 Build 2.2.005 07/24/2002**

- Fix for threading problem on OS400 which caused file-naming collisions when simultaneous AS2 payloads were being moved to non-IFS files.
- Fix for threading problem in AS2 Server (all platforms) which caused log-file and terminal-message collisions between simultaneously executing threads.
- This release includes TDCompress Build 0457.

#### **2.1.12 Build 2.2.004 07/11/2002**

- Changed database layer to use BTDAO/SQLAPI DLL's.
- Fixed a bug in TDServer which caused an assertion failure when a certain type of SSL negotiation failure occurred.

#### **2.1.13 Build 2.2.003 07/02/2002**

- Added support to TDClient for specifying the database type via the new command-line keyword DB\_TYPE=, which takes allowed values:
  - Oracle** - database is an Oracle database.
  - DB2** - database is a DB2 database.
  - ODBC** - database is another type, being access via ODBC.Note that TDServer already has this feature, specified in the tdserver.cfg file using the dbType= value.
- Added support for controlling the AS2 session timeout value, in seconds. The keyword, AS2TIMEOUT=, specifies the time the client program will wait for an AS2 server to respond, after an AS2 message has been sent. A default value of 45 minutes is used if the timeout value is 0. The value for the AS2TIMEOUT= keyword may be modified via the command-line version. For example:

```
ea2kw95c reset "network= my AS2 network" as2Timeout=600 save_only
```

will permanently change the timeout to 5 minutes (600 seconds) for the network with name "my AS2 network". Program execution will stop after the change has been written into the tdclient.ini file (since 'save\_only' was specified).

Note that the following keywords now may be used interchangeably:  
TIMEOUT=, AS2TIMEOUT=, FTPTIMEOUT=  
since each of these keywords specifies the same value for the current network (or specified network instance).

- Fixed a problem with the use of the AS2 Server Login (Basic Authentication) information specified in the Trading Partner address book for AS2 partners. The information was not being used correctly, and the client was not successfully negotiation the logon challenge from an AS2 server.
- Fixed a problem for the AS2 server where the communications logs for the Asynchronous MDN processing thread were not being purged. Now, at the beginning of each day (at midnight), these communications logs are purged.
- Modified the AS2 server to respond with a multi-line response in the case where no MDN is requested. Previously, the server responded with a 1-line "200 OK" response which one competitor product could not process.
- Improved the efficiency of 'send' transfers for the EDS EDI\*ELIT network style by cache'ing the current 'directory' settings, thereby reducing the number of 'CD' operations when sending to the same 'directory'.
- Fixed a problem on the MVS and OS400 platforms when sending EDI data to non-EBCDIC platforms. Under some combinations of settings, the data was being incorrectly sent in binary mode, when it should be sent in ASCII mode.

#### **2.1.14 Build 2.2.002 06/20/2002**

- Improved usability of 'Retrieve Audit Logs' functionality for AS1 and AS2 network styles. The Cancel button did not actually cancel anything, and if the MDN database access failed, a previously retrieved set of MDNs was incorrectly displayed. Now, upon failure to access the MDN database, an error message is displayed and no 'old' set of MDNs is displayed.
- For AS1, AS2, SMIME, and EMail network styles, made the Subject field optional in a Send/SendEDI/SendXML transfer, and made Date field expressed in GMT time.
- For AS1, AS2 and SMIME network styles, disabled the TDCompress options fields on the Advanced Compression Options screen, since all compression for these network styles is governed by a specific trading-partner relationship record managed by the TDManager administrator, along with the type of encryption, signing algorithm, and type of MDN (signed, unsigned, or none) for any communications with the trading partner.
- Added support for EDS EDI\*ELIT FTP server (network style EDS\*ELIT in the tdclient.ini file)

#### **2.1.15 Build 2.2.001 05/30/2002**

- Added ability for TDServer (AS2 server) to retain only encrypted files, this feature is activated/deactivated by the setting the new tdserver.cfg file entry, keepEncryptedOnly, to Y or N. When activated, TDServer processes incoming messages normally, then saves the encrypted message rather than the decrypted payload, and all temporary files are automatically removed (except when a processing error occurs, such as failure to decrypt or verify a digital signature; then, the temporary files are retained for problem diagnosis).
- Fixed a problem with the Windows version which caused a GPF when receiving fragmented AS1 messages.
- Fixed a problem with the quoted-printable decoder which produced incorrectly decoded messages in some cases.

- Fixed a problem with the LOCAL\_ARCHIVE network style (DataGuard network instance) which caused the Restart screen to GPF when an 'archive' operation produced a TDCompress 106 error.
- Improved error logging on startup for TDServer so that it gives a better indication of what the underlying problem (i.e. typical install issues).
- Fixed a problem (OS400 platform only) where the TDServer temp files were not being deleted in all cases.
- Enhanced support for EDI qualifiers so that the trading-partner relationship lookup for AS1/AS2/SMIME now uses the sender and receiver qualifiers. Also added support for certificate indexing using EDI qualifiers, so that the same EDI name may have different certificates if they are associated with unique EDI qualifiers.
- Modified the SendEDI transfer feature to use the Sender's EDI name to specify the signing certificate when a data signature is requested (AS1/AS2/SMIME/LOCAL\_ARCHIVE/DATAGUARD and any network configured to use SECURE compression). To use this capability, the client must have a private key for each sending EDI name, or use a trading partner address book entry (when using runtimes) or TDManager certificate referral (when using a TDManager database) to provide a private key. EDI interchanges which contain an unrecognized Sender or Receiver are ignored (after a warning message has been issued).
- Added to tdserver.cfg the TDNgineDataDir keyword to allow for the specification of the TDNgine directory into which received AS2 payloads are to be placed for TDServers which are integrated into a TDNgine installation.
- Changed the filename filter used by the parsepfx certificate import utility program (for IGN network styles). Now, all files in the directory display.
- Fixed a problem with the display of the release notes file (release.txt) in the About box. On Windows 98, the About box window was not displaying properly if the release.txt file was larger than 64K bytes in size.
- This release includes TDCompress Build 0455.

#### **2.1.16 Build 2.1.008 05/13/2002**

- Reorganized some of the entries in the tdclient.ini file, grouping the firewall-specific data into a new [FIREWALL\_SETUP] section, the database-specific data into a new [DATABASE\_SETUP] section, the EMail-specific data into a new [EMAIL\_SETUP] section, and the AS1/AS2-specific data into a new [EDIINT\_SETUP] section. Some of this data was previously specified at the network level (for each network). Now, these data are specified at the program level, with one group of settings applying for all network instances. In addition, the GUI has been modified to move the setup windows from the Edit Network screen to the main window, under the Configuration menu. See the example command-file, tdclient.cmd, for additional details.
- This release includes TDCompress Build 0455.

#### **2.1.17 Build 2.1.007 04/30/2002**

- Added a new network-style, SMIME, which composes and decomposes S/MIME messages (with no transport), and includes MDN support. The SMIME network style is like the LOCAL\_ARCHIVE style in that it does no transport at all, and is like the AS1 and AS2 styles in the way it supports S/MIME messages.

- Added ability to specify that AS1/EMail/SMIME messages larger than a specified size are to be fragmented (per RFC 2046) and sent separately, and then re-assembled by the receiving client. For AS1/EMAIL/SMIME network styles the Edit Networks screen has an additional button for configuring how message fragmentation and re-assembly is to be performed. For TDClient command-line usage, see the new keywords governing this behaviour in the tdclient.cmd file:  
**ALL\_PARTS\_ONLY**, **AUTO\_COMBINE**, **BREAK\_APART**, and **BREAK\_APART\_SIZE**.
- Changed default filename of a received PKCS7 certs-only message (AS1/AS2/SMIME) so that it is more easily viewed (on Windows) using the default (Microsoft) certificate viewer.
- Added SendXML button to Transfer screens for the AS1/SMIME network-styles.
- Converted all MIME parsing logic to use the bTrade mimeparse library.
- Added support for quoted-printable content encoding/decoding for EMAIL/AS1/SMIME network styles
- Added support for ZLIB compression, for AS1/AS2/SMIME network styles, for the Windows platform only - ZLIB will be supported for other platforms in the next product release.
- Fixed a problem with the generation of PKCS7 certs-only messages for the AS1/SMIME network style (they were OK for AS2). The messages were not being constructed properly for AS1/SMIME and could not be processed to exchange certificates.
- Fixed a problem with the btPOP3 library; it was adding a blank line to the end of the message data in some cases.
- Localized all S/MIME file I/O into new BTFILE class, to facilitate cross-platform portability .
- Added command-line (only) keyword SEQUENCE\_NUMBER, for IGN network styles only, which will set the CDH sequence number for files being sent to IGN-style networks. This feature was added primarily to support the TDNgine product in it's reconciliation of files being sent to IGN networks.

Usage:

**NETWORK**=<an IGN-style network>

"**TRANSFER**=(**NAME**=<name> ... **SEQUENCE\_NUMBER**=<up to 5 alpha-numeric> ...)"

- Added HFS\_DIR= command-line keywords for the MVS platform, to specify the HFS directory used for dynamic temp-files:

**HFS\_DIR**= - Specifies the HFS directory for temp files; may be up to 256 characters in length.

For TDServer, see the new entry in the tdserver.cfg file:

hfsDir=<your HFS directory path for temp files>

- Removed the second database specification from TDServer. Prior to this release, one could specify up to two databases for an MDN database, a TDManager database (for runtimes support), and a TDNgine database (for integration to a TDNgine). In this release, we have collapsed these into a single database specification. The software figures out whether your certificates are in the database or in static runtime files for you.
- Fixed a problem in the NT Service version of TDServer - it was terminating after running for 30 minutes and 20 seconds due to a silly bug.

- Added DIVERT\_DUP\_DATA keyword for EMAIL/AS1/SMIME network styles only. This keyword specifies whether or not messages received which contain either identical data or an identical message ID as that received in a previous message are to be diverted to the error sub-directory.
- For AS1/AS2/SMIME network styles, modified MDN creation so that the Disposition includes a Warning component when a duplicate message or duplicate data is received.
- For AS1/AS2/SMIME network styles, modified MDN archiving so that when an MDN file-system is specified, the MDN is saved in it's original form. Prior to this, only the MDN payload (multipart/report bodypart) was saved. By saving the entire MDN as it was received, any signatures are also retained.
- For all network styles supporting both SendEDI transfers and the use of the Trading Partner Address Book, fixed a problem which had prevented each envelope in an EDI file from being routed according to its receiver EDIName information. With this fix, if your EDI file has 10 envelopes and each has a distinct receiver EDI Name, and you have a Trading Partner Address Book entry for each receiver, then each envelop will be routed using the network information specified in the TP Address book. Note that envelopes with no matching Trading Partner entries are not processed, and a warning is issued when this occurs.
- Fixed a problem with the Query Mailbox window so that it will not respond to button clicking until it has finished it's current task of getting a file list, downloading file(s), or deleting file(s). In previous versions it was possible to crash the GUI by rapidly clicking buttons while it was executing its tasks.
- This release includes TDCompress Build 0455, including new versions of the inmsgp and outmsgp programs.

#### **2.1.18 Build 2.1.006 01/28/2002**

- Fixed a problem with the generation and transmission of asynchronous HTTP MDNs in the TDServer. An internal flag which manages the state of HTTP connections was not being set properly, causing subsequent connect attempts to fail.
- Fixed a TDClient problem for the EAFTP network style when using 'data-over-command' mode and the session goes through an FTP Proxy. The DOVC command was issued after the connect but before the first (proxy) login, and was not being presented to the destination server.
- Fixed a problem with the Trading Partner Setup screens when adding or modifying a Trading Partner using an AS2 network. Under certain conditions, the URL entered on the AS2 Trading Partner Setup screen was not carried back to the Setup Trading Partner screen.
- Fixed a problem with accessing stored transfers for networks added manually to the ini files due to case sensitivity.
- This release includes TDCompress Build 0453.

#### **2.1.19 Build 2.1.005 01/16/2002**

- Build to complete the migration of the TDAccess products to the TDClient product, and the migration of the Comm-Press utility to TDCompress. This mostly involved changes to some filenames (most notably, the executable and library filenames), and some externally displayed text labels and window headings.

- Added command-line keywords to allow for specification of ACCOUNT keyword for use during the FTP login sequence. See the example file, tdclient.cmd, for full details.
- Added command-line keywords to allow for full specification of trading-partner address book entries from the command-line. See the example file, tdclient.cmd, for full details.
- Includes all functionality in TDClient Version 1.xx through build V1.52 (see below for details of changes).
- This release includes TDCompress Build 0452

#### **2.1.20 Build 2.1.005 12/27/2001**

- Build to complete the migration of the TDAccess products to the TDClient product, and the migration of the Comm-Press utility to TDCompress. This mostly involved changes to some filenames (most notably, the executable and library filenames), and some externally displayed text labels and window headings.
- Includes all functionality in TDClient Version 1.xx through build V1.52 (see below for details of changes).
- This release includes TDCompress Build 0451.

#### **2.1.21 Build 2.1.004 12/03/2001**

- Includes all functionality in EasyAccess 2000 Client Version 1.xx through build V1.51 (see below for details of changes).
- Fixed a problem where the original file name was not being correctly written into the MIME header Content-Type line for AS1- and AS2-style messages.
- Fixed a problem with the AS1- and EMail-style Query Mailbox function which caused the program to crash if the CC: line in the (possibly wrapped) MIME header was longer than 512 bytes.
- This release includes the Comm-Press product, version 4.42o.015.

#### **2.1.22 Build 2.1.003 10/23/2001**

- Modified the MDN table to include the original message ID to support automated retransmission of messages which have not received an MDN within a specified time period (in conjunction with the bTrade SecurePortal). For command-line and API users, the original message ID may be specified via the command-line or API argument list using the new ORIGINAL\_MESSAGE\_ID= keyword. For example, ORIGINAL\_MESSAGE\_ID=20010910121657\_579CED8634@bTrade. This keyword should be used to specify the message ID of the message being re-transmitted. Note that a new message ID will be generated for the re-transmitted message. By specifying the original message ID, the new and original message are related in the MDN database so that when an MDN is finally received for one of the messages, then all related messages (those sharing the same original message ID) may be reconciled.
- Includes all functionality in EasyAccess 2000 Client Version 1.xx through build V1.50 (see below for details of changes).
- Added support for the following new NETWORK-related command-line keywords (which are already supported via the GUI). The following keyword descriptions are taken from file tdaccess.cmd, which provides descriptions and examples for all supported

keywords:

**AUTO\_DELETE**= Y or N. For EMAIL and AS1 network styles only. Specifies whether a message is to be auto-deleted from the mail (POP3) server after it has been retrieved.  
**SUNIQUE**= Numeric value. FTP-based network styles only. Determines whether or not to tell the server to use unique filenames.

Valid values include:

- 0 => do not use storeUnique functionality
- 1 => use storeUnique WITH serverFileSpec as arg on STOU command
- 2 => use storeUnique WITHOUT serverFileSpec as arg on STOU command

**FIREWALL\_TYPE**= Numeric value. AS2 network style only. Specifies if client must pass thru a proxy server or not to get out to the target AS2 server.

Allowed values:

- 2 => Proxy server in use
- 3 => No proxy server

The TDAccess client supports Basic Authentication, Digest Authentication and (Windows only) NTLM Authentication protocols.

**FIREWALL\_HOSTIPNAME**= Text. AS2 network style only; applicable only for FIREWALL\_TYPE=2. Specifies IP address or domain name of Proxy server.

**FIREWALL\_USERID**= Text. AS2 network style only; applicable only for FIREWALL\_TYPE=2. Specifies User ID for logging onto Proxy server.

**FIREWALL\_PASSWD**= Text. AS2 network style only; applicable only for FIREWALL\_TYPE=2. Specifies Password for logging onto Proxy server.

**FIREWALL\_PORT**= Numeric value. AS2 network style only; applicable only for FIREWALL\_TYPE=2. Specifies the port on which the Proxy server is listening.

**CERT\_IMPORT\_DIR**= Text. AS1 and AS2 network styles only. Specifies the directory which is to receive certificates sent to the user's AS1 mailbox or the user's AS2 server via "certificate-only" messages. The extracted (untrusted) certificate is copied into the directory in PKCS7 format, for further (manual or automated) processing by the user or the user's application.

**MDN\_TYPE**= Numeric value. AS1 and AS2 network styles only. Specifies what sort of MDN is to be requested from the recipient of messages sent by TDAccess.

Allowed values include:

- 1 => Asynchronous MDN using SMTP (AS1-style MDN)
- 2 => Asynchronous MDN using HTTP or HTTPS (Async. AS2-style MDN)
- 3 => Synchronous MDN using HTTP or HTTPS ('normal' AS2-style MDN)

**MDN\_DISP\_TO**= Text. AS1 and AS2 network styles only. Specifies the address to which MDNs are to be sent in response to messages sent by TDAccess. For AS1 network styles, this is just an EMail address (for example: myname@mybusiness.com). For AS2, the form of the address depends on the MDN\_TYPE value as follows:

MDN\_TYPE=1 (AS1-style MDN):mailto:myname@mybusiness.com

MDN\_TYPE=2 (Asynchronous AS2-style MDN):http://myURI:myport/myURI

or

https://myURI:myport/myURI

**MDN\_FILE\_SYSTEM=** Y or N. AS1 and AS2 network styles only. Specifies whether or not MDNs are to be stored in a directory on disk. You may choose to keep the MDNs on disk and/or keep them in the MDN database.

**MDN\_DIR=** Text. AS1 and AS2 network styles only. Applicable only for MDN\_FILE\_SYSTEM=Y. Specifies the directory in which MDNs are to be stored.

**MDN\_CUSTOM\_FILENAMES=** Y or N. AS1 and AS2 network styles only. Applicable only for MDN\_FILE\_SYSTEM=Y. Specifies the type of filename to be used when moving MDNs into the MDN\_DIR. When MDN\_CUSTOM\_FILENAMES=N, then a default naming convention is used; when MDN\_CUSTOM\_FILENAMES=Y, then the MDN\_BASE\_FILENAME= keyword is used to specify the base filenames used.

**MDN\_BASE\_FILENAME=** Text. AS1 and AS2 network styles only. Applicable only when MDN\_FILE\_SYSTEM=Y and MDN\_CUSTOM\_FILENAMES=Y. Specifies the base filename to be used when moving MDNs into the MDN\_DIR. This filename is used in conjunction with the MDN\_AUTOEXT and MDN\_APPEND flags to construct the actual filenames given to the MDNs.

**MDN\_AUTOEXT=** Y or N. AS1 and AS2 network styles only. Applicable only when MDN\_FILE\_SYSTEM=Y and MDN\_CUSTOM\_FILENAMES=Y. Specifies that the filename given to an MDN is derived from the base filename by adding a unique numeric extension to the filename.

**MDN\_APPEND=** Y or N. AS1 and AS2 network styles only. Applicable only when MDN\_FILE\_SYSTEM=Y and MDN\_CUSTOM\_FILENAMES=Y. Specifies that an incoming MDN is to be appended to the base filename, rather than writing it to a separate file.

**MDN\_DATABASE=** Y or N. AS1 and AS2 network styles only. Specifies that AS1 and/or AS2 file transmission activity be recorded in the MDN database. Received MDNs are reconciled against previously sent messages. Sent MDNs are reconciled against previously received messages.

**MDN\_DB\_DSN=** Text. AS1 and AS2 network styles only. Applicable only when MDN\_DATABASE=Y. Specifies the ODBC Data Source Name (DSN) for the MDN database.

**MDN\_DB\_SCHEMA=** Text. AS1 and AS2 network styles only. Applicable only when MDN\_DATABASE=Y. Specifies the (optional) database schema name for the MDN database.

**MDN\_DB\_USERID=** Text. AS1 and AS2 network styles only. Applicable only when MDN\_DATABASE=Y. Specifies the (optional) User ID for the MDN database.

**MDN\_DB\_PASSWD=** Text. AS1 and AS2 network styles only. Applicable only when MDN\_DATABASE=Y. Specifies the (optional) Password for the MDN database.

### **2.1.23 Build 2.1.002 10/15/2001**

- Changed file naming on AS2 Server for incoming data files to include the original filename as the last part of the server-side filename. That is, if the Content-Type or Content-Disposition MIME header includes the original filename for the data, it is included as the last part of the server-side filename, including the file extension.
- Includes all functionality in EasyAccess 2000 Client Version 1.xx through build V1.49 (see below for details of changes).

### **2.1.24 Build 2.1.001 09/14/2001**

- Major release.
- Includes all functionality in EasyAccess 2000 Client Version 1.xx through build V1.47.

- Includes changes to include AS1/AS2 MDN information in the response log.

#### **2.1.25 Build 2.18 07/13/2001**

- Product Name change
- Name change from EasyAccess to TDAccess Client

#### **2.1.26 Build 2.17 06/15/2001**

- GA Builds.
- Now include support for Basic and NTLM Proxy authorization.

#### **2.1.27 Build 2.16 06/15/2001**

- Maintenance release with the latest bTrade socket API library version.
- Now include ability to select certificate-less Diffie-Helman algorithm for FTP with SSL and HTTPS.
- Modified MDN database table and associated code to remove the TRANSID field (it was basically a duplicate of MESSAGEID), and to increase the size of the MESSAGEID field from 64 to 254 bytes.

#### **2.1.28 Build 2.15 05/18/2001**

- Added support for the GEIS Open\*Net Immediate FTP service.
- Added support for RECEIVE\_SERVER\_FILE= and DELETE\_SERVER\_FILE= command-line keywords. These keywords are used only within a Receive or ReceiveEDI transfer.
  - The RECEIVE\_SERVER\_FILE= keyword allows the user to specify that only the specified file be retrieved from the server.
  - The DELETE\_SERVER\_FILE= keyword specifies that the specified file be deleted on the server.

In both cases, the filename specified with these keywords is the server's name for the file.

- Added support for SITE command EDIREPLYBUF for IGN network styles. This site command fixes a problem when sending an EDI file with a large number of interchanges, which caused the FTP session to hang due to a backlog of server responses.
- Externalized the help filename; now specified in [IDENTIFY] in easyacc.ini.
- This release includes the Comm-Press product, version 4.42o.008.

#### **2.1.29 Build 2.14 05/03/2001**

- Now includes bTrade, Inc.'s ssock API V2.03.

#### **2.1.30 Build 2.13 04/26/2001**

**2.1.31 Build 2.12 04/24/2001**

**2.1.32 Build 2.11 03/28/2001**

**2.1.33 Build 2.10 03/24/2001**

**2.1.34 Build 2.09 03/21/2001**

**2.1.35 Build 2.08 03/14/2001**

**2.1.36 Build 2.07 03/13/2001**

**2.1.37 Build 2.06 03/12/2001**

- Release-candidate builds sent to QA test. A variety of problems were found, fixed, and returned to QA for verification. In addition, new requirements were defined during this process and were added to the code; these are noted below.
- Added special handling for duplicate received AS1 messages. Duplicate messages are recognized and handled as follows: Duplicate data messages are moved to the (new) error sub-directory, an MDN is sent (if requested), and (command-line and EA API) an error return code is generated. Duplicate MDNs are reported to the user, but are processed normally.
- Added special handling for negative MDNs. When a negative MDN is received, the user is notified, and (command-line or EA API) an error return code is returned.
- Added the ability to require a minimum level of security on incoming AS1 messages.
- Defined security levels are specified via the SECURITY\_LEVEL keyword in the specified network's section of the easyacc.ini file; allowed values are:
  - 1 => accept any data
  - 2 => required data be signed
  - 3 => require data be encrypted
  - 4 => require data be signed and encrypted

If a file is received without the required level of security, then a negative MDN is generated (if requested).

- Added ability to specify a range of client-side ports or a single port (FTP, AS2). This is useful for navigating through certain firewall configurations. The port range is specified via the LOW\_CLIENT\_PORT and HIGH\_CLIENT\_PORT keywords in the specified network's section of the easyacc.ini file; syntax rules are as follows:
  - If LOW\_CLIENT\_PORT=0, then any available client port will be used else if HIGH\_CLIENT\_PORT=0, or LOW\_CLIENT\_PORT=HIGH\_CLIENT\_PORT, then only the port specified by LOW\_CLIENT\_PORT will be used. Else, a port in the range from LOW\_CLIENT\_PORT to HIGH\_CLIENT\_PORT (inclusive) will be used. If no port is available in the specified range, then a failure is reported and the communications session is ended.
- Added ability to use the TManager database as the data repository for certificates, trading-partner relationships, and other information formerly only available in the static TDAccess runtime files. A specified TManager database will be used instead of

static runtimes if the following keywords are specified in the SECURITY section of the easyacc.ini file:

- SM\_DSN= Specifies the DSN for the database (required)
- SM\_SCHEMA= Specifies the database schema (optional)
- SM\_USERID= Specifies the database logon user ID (optional)
- SM\_PASSWD= Specifies the database logon password (optional)
- (Command-line and EA API) Added ability to receive or delete a single specified file on a server (FTP, AS1, EMAIL) by specifying the file using the server's name for it. These actions are accomplished via the RECEIVE\_SERVER\_FILE= and DELETE\_SERVER\_FILE= keywords which are used within a Receive or ReceiveEDI transfer.

### **2.1.38 Build 2.05 03/10/2001**

- Added wild-card file support for compression on UNIX platforms.
- Revised/reinstated the GUI TDAccess 2.0 Client product for UNIX, including AIX, HPUX, Sun Solaris and SCO UNIX. During 3Q99 the GUI was discontinued due to small demand for a UNIX GUI product and due to problems with a toolkit upgrade. The toolkit problems have been resolved in response to increased interest in the UNIX GUI product.

### **2.1.39 Build 2.04 03/06/2001**

- Fixed a number of problems pertaining to mis-configured easyacc.ini files by providing default values. This includes the MDN configuration information for AS1 and AS2 styles, as well as the Proxy Server info, and the ADDRESS\_BOOK flag for EMAIL, AS1, and AS2 styles. Note that only partial Proxy Server support is available in this version of the TDAccess product.
- Moved all log and restart files to the TDAccess "temp" directory.
- Added response-file creating for EMAIL, AS1, and AS2 network styles. The response file contains the unique message-id of the message file being transported to/from the server.
- This release includes the Comm-Press library (version 4.42o.004), which has corrections to the following problems:
  - (004) SMIME - moved recovered digest file to temp directory.
  - SMIME - added base64 flag to create text and signed messages

### **2.1.40 Build 2.03 03/01/2001**

- This release includes the Comm-Press library (version 4.42o.003), which has corrections to the following problems:
  - (003) COMPRESS- Fixed HASHUPDATE call if SECFILE and HEADERCOMP(Y).
  - COMPRESS- Fixed duplicate records if SECFILE and SENDEDI.
  - COMPRESS- Fix to save g->gs\_ver after ParseISAGS.

#### **2.1.41 Build 2.02 02/21/2001**

- Added two new network styles to support the GE InterLinX FTP service and the GEIS Enterprise FTP service.

#### **2.1.42 Build 2.01 02/09/2001**

- Major release includes AS1/AS2 EDI-INT functionality. The EDI-INT engine has been certified to be interoperable for AS2 by the Uniform Code Council per Interoperability Testing carried out under the auspices of the Drummond Group, Inc.
- This version of the TDAccess client supports only the following network styles:
  - EMAIL - Performs SMTP/POP3 Internet Mail transfers (i.e. sends and receives Email).
  - AS1 - Performs S/MIME encryption/decryption with Message Disposition Notifications (MDN), using SMTP and POP3. Supports in-band or out-of-band certificate exchange using the PKCS7 messages.
  - AS2 - Performs S/MIME encryption/decryption with synchronous or asynchronous Message Disposition Notifications (MDN), using HTTP or HTTPS. Supports in-band or out-of-band certificate exchange using the PKCS7 messages. (bTrade, Inc. AS2 Server available separately).
- A new utility program, parsedlg.exe, allows for the local import of generic X.509 certificates into the TDAccess runtimes (i.e. the certificate store).

#### **2.1.43 Build 1.52 12/13/2001**

- Fixed an FTP library bug which intermittantly caused the data transfers to fail when using the command-over-data feature (EAFTP network style only).
- Added support for controlling the FTP session timeout value, in seconds. This is supported on a network-by-network basis via a keyword, FTPTIMEOUT=, in each network's section within the easyacc.ini file. The value for the FTPTIMEOUT= keyword may be modified via the command-line version. For example: `tdclientc reset "network= my network" ftpTimeout=120 save_only` will permanently change the timeout to 2 minutes (120 seconds) for the network with name "my network". Program execution will stop after the change has been written into the tdclient.ini file (since 'save\_only' was specified).
- Added support for controlling server-side response buffering (IGN network style only), when sending EDI data. This is accomplished using the new command-line keyword, SERVER\_RESPONSE\_BUFFERING=, which also appears in each network section within the easyacc.ini file. Appropriate values for this keyword are Y or N. For example, to enable server response buffering, use:

```
ea2kw95c reset "network= my IGN network" server_response_buffering=Y
save_only
```

When set to N, the IGN server will parse the EDI file and send a response for each interchange as it is received. When set to Y, the IGN server will accumulate the responses and send them only after receiving all of the EDI data.

- This release includes the Comm-Press product, version 4.42o.015.

#### **2.1.44 Build 1.51 11/05/2001**

- Fixed a problem with the parsepfx utility which prevented it from importing an IBM certificate if a certificate had been previously imported into the runtime/ign sub-directory. Basically, the program will now delete the certificate files in this sub-directory prior to processing the IBM PFX or PKCS12 file.
- Fixed a problem which occurred when the INIPATH= keyword was used for the GUI for the IGN network style; the program could not find the parsepfx program (launched from the Security Menu) because it looked for it in the INIPATH directory rather than the program directory. Also, modified the way the GenKeys program is launched, to handle a similar problem in the command-line program (which launches GenKeys in response to the GENKEYS keyword) when used in conjunction with the INIPATH= keyword.
- Fixed a problem with the GENERIC\_SSL network style which resulted in a program crash when attempting to connect to a server using SSL. The problem was introduced when the new Socket API library was introduced and was only recently detected (apparently the EAFTP or IGN network styles are used most often to do FTP with SSL).

#### **2.1.45 Build 1.50 10/20/2001**

- Extended the use of the USE\_ORIGINAL\_NAME keyword introduced in Build 1.49 to include compressed files, and to include the LOCAL\_ARCHIVE network style. Now, if the USE\_ORIGINAL\_NAME keyword is used in a Receive or ReceiveEDI transfer for any IGN or EAFTP network style server, or with an Unsecure or UnsecureEDI transfer with the LOCAL\_ARCHIVE network style, then compressed files will be decompressed in a way so that the original file names stored within the compressed files will be used to name the output files. For example, if a Windows user compresses and sends 10 files in a directory by specifying a wild-carded filename, say \mydata\\*.txt, as the file-spec to be sent or archived, then the resulting compressed file which is transmitted to the server or local-archive contains the original file names of the 10 compressed files. If the file is received from the server or local-archive using the USE\_ORIGINAL\_NAME keyword, then when the file is decompressed, the 10 files are delivered into the directory specified in the transfer, and are given their original names. NOTE: USE AUTOEXT OR APPEND WHEN USING THE 'USE\_ORIGINAL\_NAME' KEYWORD; otherwise, your files may be overwritten by newly arriving files.
- For the FEDEX network style, added a 5-second pause (sleep) just prior to ending the FTP connection, to handle some sort of timing problems experienced by a small number of FEDEXNET users.

#### **2.1.46 Build 1.49 10/10/2001**

- Fixed a problem with the command-line parser which required that certain keywords be followed by a space in order to be correctly recognized.
- Added ability to specify in a Receive or ReceiveEDI transfer that the file(s) received should retain either their original file-name, or their name on the server, or should use a user-specified name. This only applies to Windows and UNIX platforms only (not applicable for OS400 or MVS). The original file-name is the name the file had before it was sent to the server. The IGN and EAFTP network styles are the only styles which support original file-names. The server file-name is the unique name the server gives to the file when it is sent to the server. This change ties together loose bits of functionality previously available for some network styles and generalizes them for all network styles in a simple way, although original file-names are only available for IGN

& EAFTP styles. In the DEFAULT\_RECEIVEPARMS section for each network instance (in the easyacc.ini file), the default behaviour for the network is defined with new entries:

**USE\_ORIGINAL\_NAME=N or =Y** and **USE\_SERVER\_NAME=N or =Y**. (Missing entries are defaulted to 'N').

If the user specifies a fully-qualified fileName to receive the file, then neither of the new keywords apply, and the file is named as specified.

If the user specifies only a directory to receive the file, then the values of the new keywords come into play. In the case where a directory is specified and both **USE\_ORIGINAL\_NAME** and **USE\_SERVER\_NAME** are set to 'N', then the server fileName is used. The file-naming behavior may be overridden for a particular transfer using the new 'Retain original fileName' and 'Retain server fileName' checkboxes on the Other Receive Options screen (GUI), or using the new command-line keywords

#### **USE\_SERVER\_NAME and USE\_ORIGINAL\_NAME**

For example, a command-file transfer definition such as:

```
TRANSFER=(NAME=my_receive RECEIVE=. USE_ORIGINAL_NAME)
```

will receive files into the current directory ('.') using the file(s) original name(s).

- Fixed small GUI problem where the main menu was accessible while the Receive Runtimes process was executing.
- Fixed a problem with the auto-Dial feature (Windows only), for the IGN network style, which caused a dial-attempt loop when Query Mailbox was selected when the AGNS dialer program, IDIALER.EXE, was incorrectly specified.
- Tweaked the look of the AGNS Setup Dialer screen (IGN network style) to clearly display the field labels under different screen resolutions.
- Fixed a bug in the reporting of the file receiving decompressed data during transfer execution, when auto-extended filenames are specified. The reported filename(s) were constructed using a value of the 'autoext' parameter one less than actually used, resulting in a mismatch between the reported filename and the filename actually created and populated with the data.
- Changed default value of LOG\_SOCKET value to the newly supported value of zero which was added to the socket API layer recently LOG\_SOCKET=0 (no logging)(see also notes for Build 1.47).
- Fixed a long-standing problem with the IGN network style where compressed EDI files were not properly handled by a Receive transfer (however, ReceiveEDI worked fine). That is, in the IGN network style, a Receive transfer retrieves all files in the user's mailbox, but compressed EDI files were not processed correctly - the EDI parameter was not used in the decompress step. This caused the EDI data to be output into one file for the EDI header and a separate file for the body of the EDI data. Now this has been fixed using the Common Data Header fields which indicate if a file is an EDI file or not.

### **2.1.47 Build 1.48 09/23/2001**

- Modified the way ASCII to EBCDIC character translation is done to make for a more flexible design; this is in anticipation of allowing customer-specified translation tables to be used in a future release.
- Modified the way CRLF handling is performed when using FTP transport across platforms; this was done to allow MVS users to specify CF or LF as a segment terminator when sending cross-platform.
- Fixed a problem on MVS involving the statist.txt restart file, which was not being written before it was opened for read.
- Fixed a problem with the restart files when a single file was being sent. The problem caused the restart files to be incorrectly reset when a problem occurred if the file was being compressed.
- Fixed a problem with the new LOG\_SOCKET= IDENTIFY-section keyword which forced an entry to be present when using the LOCAL\_ARCHIVE network style.
- Fixed a problem which caused the Transfer dialog box to display "... EDI interchange processed by server ..." as a message from the IBM Information Exchange FTP server, for non-EDI data. This message is displayed in response to a keyword in the Common Data Header sent by the IBM I/E FTP server, but it only meant for EDI transfers.

### **2.1.48 Build 1.47 09/07/2001**

- Added support for the default alias table when sending EDI data using the IGN network style. In the GUI, when a SendEDI transfer is being constructed, the Alias Table drop-down list now contains the entry "Use default alias table". When selected this entry causes the default Global Alias Table (GX...) to be used by the IGN-style server when it processes the EDI data. To accomplish this using the command-line or library versions of EasyAccess, specify the command-line keyword phrase: "SENDUSERID=Use default alias table" within your SendEDI transfer definition statement. (Note use of double quotes; could use single quotes, parentheses, or square brackets, as well).
- Added support for the LOG\_SOCKET= entry in the IDENTIFY section of the easyacc.ini file. This keyword enables or disables logging for the low-level transport libraries, including the SSL and socketapi libraries.  
Valid values are:

LOG\_SOCKET=1 (minimal logging)  
LOG\_SOCKET=2 (extended logging, no data traffic)  
LOG\_SOCKET=3 (extended logging, including all command and data traffic)  
LOG\_SOCKET=4 (maximum logging, including certificate manipulation)

### **2.1.49 Build 1.46 08/28/2001**

- Fixed a problem in the response log for the GEIS Open\*Net and GEIS Enterprise network styles where the remote filename field was not being populated properly.
- Fixed a problem with the ini-file updating scheme introduced in Build 1.45 which affected only the GUI version (new transfers were not being written to the exfer.ini file).

### **2.1.50 Build 1.45 08/27/2001**

- Improved performance (esp. on UNIX platforms) by reducing the number of times the exfer.ini and easyacc.ini files are updated during argument processing. In simple benchmarks involving the definition and execution of 100 transfers from a command-file, overall execution time was reduced from about 10 minutes to about 2.5 minutes.
- Fixed a problem in the management of an FTP session if the connect failed. This problem caused intermittent crashes of the program.
- Changed the logic governing the creation of the response-log file. As of this release, the response log file will be produced ONLY if the 'RESPLOG=' command-line argument is used.

### **2.1.51 Build 1.44 08/24/2001**

- Changed the EAFTP network style so that a SendEDI transfer will not apply a default Class/APRF value of "EDI" to the transfer if no Class/APRF is provided by the user.
- Now interpreting a decomp return code of 69 as a successful return code. This return code implies that one or more segments of SAFE-mode compressed data failed to decompress and were saved to a specified error file. (See comm-press/decomp documentation). Prior to this change, a return code of 69 was treated as a failure condition, requiring a restart of the transfer(s) in order to complete the specified file downloads. Now, a restart is no longer required.

### **2.1.52 Build 1.43 08/22/2001**

- Change for all network styles so that a Send transfer with specified SECFILE= does NOT split proprietary EDI interchanges into separate files which are sent separately. Note that sendEDI will still split EDI data into separate interchanges for separate sending for the EAFTP, GEIS Open\*Net, GEIS EDI\*Switch, EDI-INT, GISB-CLIENT, and GISB-SERVER network styles, and if SECFILE= is specified in the Advanced Compress parms, then it will do this for proprietary EDI data.
- Changed EAFTP network style so that the Class/APRF field is optional for a Send transfer.
- This release includes the Comm-Press product, version 4.42o.011.

### **2.1.53 Build 1.42 08/14/2001**

- Fixed problem which prevented the FTP 'QUIT' command from being issued for the IGN (SSL) and EAFTP network styles.
- This release includes the Comm-Press product, version 4.42o.010.

### **2.1.54 Build 1.41 07/31/2001**

- For EAFTP network style, added support for embedded spaces in the Class field and sender and receiver mailbox name fields when doing a List command.

### **2.1.55 Build 1.40 07/27/2001**

- (EAFTP network style only) Added support for new LIST command flags for improved performance.

- Added the QUERY\_STATUS= command-line keyword (for EAFTP network-style only). This keyword specifies that action QUERY\_LIST should return a list of files with the specified status only. The allowed status values are:
  - **AVAILABLE** - files which have not yet been received
  - **RECEIVED** - files which have been already received
  - **DELETED** - files which have been deleted

If not present, then file list will contain entries for any status (just as before the introduction of the QUERY\_STATUS keyword).

- Added NAME() field to the response log; this field will contain the transfer name without the network-name prefix. The response log is an optionally produced log file which details file send and receive activity for external applications.
- This release includes the Comm-Press product, version 4.42o.009.

### **2.1.56 Build 1.39 07/15/2001**

- Added support for tight integration of the EasyAccess client with the bTrade TDNgine Portal, including the generation of a "response log" for each program run. The response log details all file transmission activity.
- Added command-line keywords MAILBOXUSERID= and MAILBOXPASSWD= for the OPEN\*NET network style.
- Added command-line keyword DHONLY= to specify whether or not the client is to do ONLY Diffie-Helman non-certificate-based SSL.
- Added check to ensure transfers being defined on the command-line or in the command-file have unique names.
- Fixed a problem parsing the listfile for IGN-IE style networks, where mailbox entries with no Filename in their Common Data Header were being ignored. EasyAccess uses the CType CDH field to determine if the file is to be received in ASCII or binary mode. The DFormat field is used to determine if the file is an EDI file or not; note that the presence of any CDH field beginning with 'EDI' also signifies that the file is an EDI file. The Filename field is used to display the original filename to the user, and to allow the user to give received files their original filenames; this is done by specifying only the directory to receive into, rather than a complete file specification.
- Increased the number of transfers which may be created or referenced from the command-line or from a command-file from 25 to 200.

### **2.1.57 Build 1.38 01/25/2001**

- This release includes the Comm-Press library (version 4.42o), which now has support for direct communication with the TDManager database via native SQL calls.

### **2.1.58 Build 1.37 12/08/2000**

- Added support for the GEIS eXchange Service (GXS).

- Fixed a problem with the display of certificates in the Security/Registration screen which caused current certificates to incorrectly display as 'Pending' under some circumstances.
- Added functionality to import PKCS12-format certificates (IGN network-style only).
- Changed the Configuration/Setup Network/Edit Network screen to allow the control port to be editable for all network styles which use a control port.
- Major changes in this release are included in the Comm-Press library (version 4.42n) and FTP and SSL libraries (version 2.0). The Comm-Press V4.42n library has a number of bug fixes. The FTP and SSL libraries now support 'data-over-command' FTP, which requires only a single socket connection (port) to perform FTP.
- Changed the Transfer screen for GENERIC and GENERIC\_DOS network styles so that the "APRF/Class" field is now labelled "SubDirectory", and is no longer a required field.
- Fixed a bug in the "xfer" library so that the FTP LIST or NLST commands may be issued with no parameter. Previously, a trailing space was appended to the LIST or NLST command, which caused a problem for some FTP servers.

### **2.1.59 Build 1.36 08/03/2000**

- Added support for a backup Dial method (Windows only). If your first dialer selection fails to connect, then EasyAccess will invoke your second choice.
- Added command-line keywords for specifying a Dialer program, instead of a DialUp Networking entry, for your primary Dial method (Windows only). The new keyword is:

**DIAL\_PROGRAM=** Text filespec (fully qualified filename) of a Dialer program that you want launched (instead of DialUp Networking). For example, the AT&T Global Network Services Dialer program IDIALER.EXE (plus directory path) may be specified. If both **DIAL=** and **DIAL\_PROGRAM=** are specified, then the Dialer program will be used.

See the example command-line file, `easyacc.cmd`, for a usage example.

- Added command-line keywords for specifying the backup Dial method (Windows only). The new keywords are:

**BACKUP\_DIAL=**Text name of a backup DialUp Networking entry you have previously set up on your computer. The backup Dial entry is used if the primary DialUp Networking entry or Dialer program fails to connect.

**BACKUP\_DIAL\_PROGRAM=**Text filespec (fully qualified filename) of a Dialer program that you want launched (instead of DialUp Networking) as your backup. For example, the AT&T Global Network Services Dialer program IDIALER.EXE (plus directory path) may be specified. The backup Dialer program is used if the primary DialUp Networking entry or Dialer program fails to connect. If both **BACKUP\_DIAL=** and **BACKUP\_DIAL\_PROGRAM=** are specified, then the Dialer program will be used as the backup. See the example command-line file, `easyacc.cmd`, for usage examples.

- Modified the auto-dialer so that it reports the dialer progress in the main transfer window, and so that it is invoked during a restart.
- Added support third-party Dialer Program support for the FEDEXNET network style. The AGNS Dialer program is the default.
- Corrected the Change Password logic for the FEDEXNET network style.

- Corrected the server-side filename generated when sending wild-carded files, so the server-file is not the same as the client wild-carded filename.
- Improved performance for transfers which send or receive many files by splitting out the file-list(s) formerly in the EasyAccess restart file (eastatus.txt). Since the eastatus.txt file is updated frequently, it substantially reduces this overhead by moving the static list of files to a separate file: statfile.fil in the EasyAccess temp directory.
- Enhanced the GEIS EDISwitch Send Transfers for EDIFACT data. The message type from the first UNH segment is used as the APRF value, and the control number from the UNB segment is used as the SNRF value. These values are part of the filename used during the FTP put. For example:

STOR %RECEIVER%MSGTYP%CTLNO%E

### **2.1.60 Build 1.35 06/19/2000**

- Added EDI envelope-splitting capability to (GEIS) EDISwitch network style. When a SendEDI transfer is specified, this change will cause EDI (X12 and EDIFACT) files to be split into separate files, one file for each envelope (interchange), and sent to the mailbox specified in the EDI header, or via the ALIAS table (in the EasyAccess runtime directory) as follows: if the EDI header specifies the receiver to be 'R', and the ALIAS table has an entry 'M=R' in the section labelled 'NALIAS', where N is the name of the current network, then the envelope will be sent to the mailbox M on the N network. For example, if EDISWITCH is the name of the current network and the ALIAS table contains the entries  
[EDISWITCHALIAS]  
JEFF=3013401234  
and the EDI header of an envelope specifies the receiver to be '3013401234', then the envelope is routed to the EDISWITCH mailbox 'JEFF'; if no such ALIAS table entry is found, then the envelope will be sent to the EDISWITCH mailbox '3013401234'.
- Modified support for sending EDI data using proprietary headers. As of this release, a Send transfer should be used (rather than a SendEDI transfer) to send EDI files using proprietary headers. This is accomplished by specifying the location of a file describing the layout of the proprietary header using the SECFILE= keyword in the Advanced Compress Options section of the Send/Compress Options screen, (or in the OTHER\_COMP\_PARMS section of your command-line transfer) and by selecting a Send rather than a SendEDI transfer. This change was necessary since when such files are compressed, they must be decompressed without the EDI decomp option, and so must be sent using a Send rather than SendEDI transfer. This option is supported for all network-styles which display the Compress Options button on the Add Transfer, Edit Transfer, and Run Transfer screens. Please refer to the Comm-Press User's Guide for more details on the SECFILE= keyword and the supported formats for the header layout file.
- Added support for overriding the Send Class (known as APRF, APRF%SNRF, or Subject for some network styles) when using proprietary EDI headers. If the proprietary header layout file specified by the SECFILE= keyword specifies the location of Class data (using the CLASSSTART and CLASSLENGTH keywords), then the Class is taken from the EDI header, rather than the value entered in the Class (APRF/APRF%SNRF/Subject) field on screens noted above (or in the SENDCLASS keyword in your command-line transfer).

### **2.1.61 Build 1.34 06/06/2000**

- Added functionality to IGN-I/E network style so that during a SendEDI transfer, if IGN rejects an interchange for any reason, the user is made aware of the error. The GUI

displays a message dialog; the command-line displays a text message. Such an error does not cause transfer processing to cease. When all transfers are completed, the command-line exits with a return code 129 and logs the following message:

Exit Status: FAILURE (99999).

- Corrected problem sending EDIFACT data to EAFTP SecurePortal. The problem caused the 'Put' operation to fail when the EDIFACT data used single-tick (') delimiters due to a parsing problem of the UNB header.
- Corrected problem sending EDI data to EDISwitch, where the SNRF was not conveniently specified via the GUI transfer screens.

### **2.1.62 Build 1.33 05/15/2000**

- Added proprietary EDI-header support, for sending EDI files using non-X12/non-EDIFACT EDI headers. This is done using keyword SECFILE=<your header template file> in the Compression Options/Advanced Options line for a SendEDI Transfer.
- Added support for new network style 'ICC-NET' for communicating with Internet Commerce Corporation's ICC.NET service.
- For EDI\_INT network-style, now using MIME Content-Types defined in RFC 1767 when sending EDI data (types EDI-X12, EDIFACT, or EDI-Consent).
- Fixed a problem with the transmission to/from bTrade, Inc. SecurePortal of EDI data which uses non-printable segment terminators.
- Enhanced GENERIC\_MVS network style to allow use of single quotes in the MVS dataset name.
- Fixed bug present in UNIX versions (only) which prevented wild-card expansion of files from working when performing SendEDI transfers to EAFTP, EDI\_INT, GISB\_CLIENT and GISB\_SERVER network-style servers.
- Completed port of SMTP/POP3-based EDI\_INT, GISB\_CLIENT and GISB\_SERVER network-styles to all UNIX platforms (EasyAccess command-line only).
- Improved MIME parsing for EDI\_INT, GISB\_CLIENT and GISB\_SERVER network styles to allow payload data to be sent in-line or as an attachment.
- Fixed a memory leak in the XFER class in ea2kw95d DLL (or library) due to the base class destructor not being declared virtual.
- Fixed all memory leaks in the SSL library.
- Added new entry to easyacc.ini file which allows the Windows RAS Dialer functionality to be completely turned off within the EasyAccess 2000 Client (Windows 95/NT only). To disable the Windows RAS Dialer, modify the DISABLE\_DIALER line in the IDENTIFY section of the easyacc.ini file to be:  
DISABLE\_DIALER=Y  
To enable the Dialer, make sure it has it's default setting:  
DISABLE\_DIALER=N

### **2.1.63 Build 1.32 03/27/2000**

- Added support for Persistent Retry, which specifies that the Send or Receive portion of a transfer repeat after a pause of specified duration. See the new keywords PERPETUAL\_SEND= and PERPETUAL\_RECEIVE= in the easyacc.cmd example command-file.

- Added support for the FTP variant 'command-over-data', which combines the two standard FTP socket connections into a single socket connection. This feature is supported by bTrade, Inc.'s SecurePortal server product. This feature is quite useful when intervening firewalls cause problems for standard FTP traffic.
- Added support for new network style 'GENERIC\_MVS' which is tailored for sending files to an MVS-based FTP server.
- Modified the way the Advanced Compression and Advanced decompression options are stored, to be more convenient for a new JNI API being developed to support a Java GUI. Now, the Advanced Compression options are stored as a single entry within the Send Params of a Transfer. The data entered in the Advanced Compression Options field are essentially passed 'as is' to the bTrade, Inc. compression library; the same for the Advanced Decompression Options.
- Fixed a bug with the command-line parser which was incorrectly matching a keyword if that keyword's text was contained within a larger keyword.
- Added logic to update the [EAPATH] section in the easyacc.ini file to correctly reflect the installation directory.
- Minor changes to utility functions to allow for use by Java GUI; this involved the passing mechanism for status messages from the various callbacks to utilize a JNI interface back to a specified Java object.
- Removed the term 'FTP' from screen titles, screen field labels, and log files, to reflect the multi-protocol nature of the product.
- Improved error handling when an error occurs while sending EDI files to an EAFTP-style server; if the de-enveloping step fails, a more intelligible error message describes what was wrong with the EDI file.
- Changed the EDI-INT network style to transmit the EDI data as a file attachment to the bTrade, Inc. EDI-INT server, instead of as in-line data.
- Added facility to show a progress bar when receiving a file when it's size is known.
- Fixed a bug on Setup Dialer screen where, when the screen first comes up a blank entry was displayed instead of the DialUp Networking entry chosen and saved in a previous visit to the screen.

### **2.1.64 Build 1.31 02/01/2000**

- Now including the DialUp Networking 1.3 Upgrade (DUN 1.3) for Windows 95 for Windows 98. See the release notes in msdun13.txt for information. The upgrade is included in a self-extracting/self-installing executable which may be run by the user should an upgrade be needed. EasyAccess users running on Windows 95 who see errors referring to Windows DLL ws2\_32.dll can install the upgrade to fix the error.
- Incorporated fix to the FTP library to handle problems with the ls and dir FTP commands when communicating with IGN FTP server using SSL.
- Incorporated (new) version 4.4.1r of bTrade, Inc.'s compression library which allows carriage-return as segment delimiters in EDI data.
- Added new command-line keywords VALIDATE\_TRANSFERS\_ONLY and SAVE\_ONLY. See the command-line example file easyacc.cmd for a description and example usage.
- Added to the command-line version (only) the ability to cancel a transfer which is in progress. This is done by creating a (possibly empty) file named 'cancel.fil' in the

EasyAccess 'temp' sub-directory. The presence of this file will cause the current transfer to be terminated. This feature has been added primarily for those users who embed the EasyAccess command-line program within their own applications.

- Added support for GISB\_CLIENT and GISB\_SERVER network-styles. These styles allow the EasyAccess client to send and receive EDI files through the bTrade, Inc. GISB Server.

### **2.1.65 Build 1.30 01/13/2000**

- Incorporated fix to the FTP library to handle dropped Command connection during the sending or receiving of large files. The fix allows EasyAccess to work reliably using unreliable network connections.
- Fixed a problem with the 60-day rule for generating the year displayed on the Query Mailbox screen for the Fedex and GEIS Express network styles. The problem caused the wrong year to display when the current month was November or December.
- Fixed a problem, present only in the GUI version, which prevented transfers specified on the command-line from being added to the list of selected stored transfers. The transfers were added to the list of defined transfers, but not to the list of selected (to run) transfers.

### **2.1.66 Build 1.29 12/30/1999**

- Added support for a new network-style called MARK\_III. This style is a variant of the GEIS EDI\*Express style which is based on the GEIS MARK III TCP/IP FTP server (whereas the EDI\*Express style is based on the GEIS MARK 3000 TCP/IP FTP server). This new style supports the same functionality as the EDI\*Express style, including the sending and receiving of EDI files, receiving GEIS audit reports, querying the contents of your mailbox, and so on. To get around a problem handling EDI data, which has an implicitly defined segment terminator of Carriage Return, this style will, when receiving a file, tell the MARK III server to replace the segment terminator with a tilde (~) character.

### **2.1.67 Build 1.28 12/03/1999**

- Added support for a new SMTP/POP3-based network style (Windows platforms only). The new style allows for the sending and receiving of files using Internet Mail (i.e. E-mail). The transmitted files may be compressed and/or encrypted. Although this new EasyAccess network-style is called EDI-INT, the Build 1.28 release does not support S/MIME encryption or automatic generation of Message Disposition Notification (MDN) messages. This is an 'early release' of functionality upon which a fully RFC-/draft-compliant EDI-INT network-style will be developed. See the EDI-INT specification in Internet Draft draft-ietf-ediint-as1-11.txt 'MIME-based Secure EDI' (September 1999) and the RFCs and Internet Drafts referenced therein.
- Added command-line support for Audit and Query Mailbox functionality to both the GUI and command-line versions of EasyAccess. The AUDIT keywords allow the user to retrieve a server audit log, for the IGN, WALMART, GEIS EDI\*Express, and EAFTP network styles. The AUDIT keywords are:

**RECEIVE\_AUDIT\_LOGS** - Instructs EasyAccess to create and execute a transfer to receive an audit report from the current server of files sent and received.

**AUDIT\_START\_DATE=** - Used in conjunction with RECEIVE\_AUDIT\_LOGS to specify the interval of time the audit report should cover.

**AUDIT\_END\_DATE=** - Used in conjunction with RECEIVE\_AUDIT\_LOGS to specify the

interval of time the audit report should cover.

**AUDIT\_FILE=** - Specifies the qualified file name of the file when retrieving the audit logs. If not present, the audit logs are written to the default file, audit.log, in the EasyAccess 'temp' directory.

The QUERY keywords are:

**QUERY\_LIST** - Instructs EasyAccess to create and execute a transfer to receive a list of available files from the current server.

**QUERY\_FILE=** - Specifies the qualified file name of the file to receive the server file list. If not present, the file list is written to the default file, list.fil, in the EasyAccess 'temp' directory.

- Added support for a network-style called 'GENERIC\_DOS'. This network style will correctly send and receive files to DOS-based and Windows-based FTP servers. It differs from the 'GENERIC' and 'GENERIC\_SSL' network-styles only in the way it processes the file-lists which result from the dir and ls commands (DOS-based servers produce a DOS-style dir listing, instead of the more common UNIX-style ls listing).
- Added Query Mailbox function to the 'GENERIC', 'GENERIC\_DOS', and 'GENERIC\_SSL' FTP network styles.
- Added auto-reconnect capability to all FTP network-styles. This will cause a session to be re-established if it times out. A time out may occur on the Query Mailbox screen if the user does not take any action for some period of time, typically 5 to 15 minutes (depends on the server), or if a file compression takes longer than the timeout period. An auto-reconnect occurs transparently, upon detection of timeout.
- Added support for Proxy Servers which first require a logon to the Proxy Server, and then a separate logon to the (mapped) target FTP server. Currently, the setup of the Proxy Server specifications within EasyAccess must be done using command-line keywords, as follows:

**PROXY\_TYPE=** - Specifies a Proxy Server is to be used to connect to the target FTP server. Allowed values are currently limited to: PROXY\_TYPE=1 which specifies that Proxy Server requires a login UserId and Password. A connection is first established to the IP address specified by the IP keyword (Primary IP address or domain name); then the Proxy Server login takes place using the userId and password specified by keywords PROXY\_USERID and PROXY\_PASSWD; and then the target FTP server login occurs using the userId and password specified by the FTPUSERID and FTPPASSWD keywords.

**PROXY\_USERID=** For PROXY\_TYPE=1, specifies the login userId for the Proxy Server.

**PROXY\_PASSWD=** For PROXY\_TYPE=1, specifies the login password for the Proxy Server.

### **2.1.68 Build 1.27 10/20/1999**

- Added pre- and post-processing options to the SEND and RECEIVE portions of any Stored Transfer. The user may execute any external program(s) or operating system command(s) by specifying a command-line (program name and path plus program arguments). Each transfer can be configured in this way to run an external program prior to or after sending or receiving files.
- Added Auto-Retry capability and associated Setup Delivery Options screen. This allows for the automatic retry/restart of any set of Stored Transfers which fails to execute to completion for some reason (dropped network connection, power outage, etc.). Via the Setup Delivery Options screen, the user may specify whether to enable auto-retry, the number of retries to attempt, and the period of time between retries.

- Fixed a problem in the IGN network style where a Receive transfer would fail if there were no files found on the IGN server. The correct behavior is to simply ignore this condition and to proceed to the next transfer, if specified.
- Added Trading Partner Address-Book. Available only for the EAFTP network-style currently. The Address Book allows you to enter all your trading partners and specify for each one their Primary Network and Mailbox as well as a Backup Network and Mailbox. The Backup Network and Mailbox are used automatically by EasyAccess if a failure occurs while executing a transfer. Specifically, EasyAccess will switch to the Backup Network and Mailbox after one-half of the specified retries have failed on the Primary Network. The Trading Partner Address Book allows you to send and receive to/from ANY mailbox on any of your defined Networks in a very convenient way.
- Added support for adding/editing Trading Partner Address Book entries from the command-line via the following new command-line keywords:

**TPBOOK=(.....)** - Specifies an entry in the Trading Partner Address Book is to be added or edited.

**NAME=** - Specifies the Trading Partner name to be added/modified.

**NETWORK1=** - Specifies the Primary Network to be used when sending to or receiving from this Trading Partner.

**MAILBOX1=** - Specifies the Primary mailbox, or user ID, or login name for the Trading Partner on the Primary Network.

**NETWORK2=** - Specifies (optional) Backup Network to be used when sending to or receiving from this Trading Partner. The Backup Network is used only if a transfer fails using the Primary Network and Mailbox (if Auto-Retry is enabled). The Backup Network will be used for the last half of the specified retries. That is, if Auto-Retry is enabled and MAX\_RETRY is 2, then when a transfer fails, the first retry will use the Primary Network and the second retry will use the Backup Network.

**MAILBOX2=** - Specifies (optional) Backup Mailbox, or user ID, or login name for the Trading Partner on the Backup Network.

Here is an example command-file entry to create/modify a Trading Partner:

```
TPBOOK=( NAME=MyPartner NETWORK1=EAFTP-INTERNET
MAILBOX1=MyPartnersMailbox
NETWORK2='IGN-I/E SSL'
MAILBOX2=CMAP.MyPartnersIGNAccount)
```

\* Note that the Trading Partner Address Book is used only by the EAFTP network style at this time.

- Added support for controlling Auto-Retry from the command-line via the following new command-line keywords:

**RETRY=** - Enables/disables AutoRetry. RETRY=Y enables; RETRY=N disables.

**MAX\_RETRY=** - Specifies the number of automatic retries. MAX\_RETRY=2 - Specifies that when a transfer fails, EasyAccess should retry the transfer twice. If the currently selected network-style uses the Trading Partner Address Book, then in this example the Backup Network and Mailbox would be used for the second retry (if the first retry failed).

**RETRY\_DELAY=** - Specifies the time, in seconds, between retry attempts.

RETRY\_DELAY=15 specifies a delay of 15 seconds.

- Added support for controlling pre- and post-processing for the Send and Receive steps from the command-line, using the following keywords: That is, these keywords let you tell EasyAccess to execute a program before or after you have it send or receive files.

Note that you must use these keywords inside a transfer definition.

#### Keyword Usage guideline

**SEND\_VERIFY=** Y or N. - Tells EasyAccess whether it should check for the existence of the file(s) you are telling it to Send at the time the transfer is being created. The default is to check that the files exist, to catch typing errors and so on. However, if you are executing a Send Pre-processing program which will create the files to be sent, then you will want to disable this checking, since the files may not exist until the transfer is executed.

**RECEIVE\_VERIFY=** (DATAGUARD only) Y or N. Tells EasyAccess to check for the existence of the file(s) you are telling it to, unsecure at the time the transfer is being created. The default is to check that the file(s) exist, to catch typing errors and so on. However, if you are executing an Unsecure (Receive) Pre-processing program which will create the files to be unsecured, then you will want to disable this checking, since the files may not exist until the transfer is executed.

**PRE\_SEND=** - Specifies a program or command-file is to be run prior to the Send cycle of a transfer. See below for the syntax and related keywords.

**POST\_SEND=** - Specifies a program or command-file is to be run after to the Send cycle of a transfer. See below for the syntax and related keywords.

**PRE\_RECEIVE=** - Specifies a program or command-file is to be run prior to the Receive cycle of a transfer. See below for the syntax and related keywords.

**POST\_RECEIVE=** - Specifies a program or command-file is to be run after to the Receive cycle of a transfer. See below for the syntax and related keywords.

The **PRE\_SEND=**, **POST\_SEND=**, **PRE\_RECEIVE=**, and **POST\_RECEIVE=** keywords all use the following syntax and keywords to specify a program or command-line to be executed and how to tell EasyAccess how to check if the program or command-line executed successfully:

#### Keyword Usage guideline

**CMDLINE=** Specifies the directory and filename of the program, batch file, script file, or operating system command to be executed, along with any arguments to be passed to the program/script/command.

**RETCODE=** Numeric value. Specifies a return code value to be used in determining if the program/script/command executed successfully or not.

**SUCCEEDS\_IF\_GT** - Tells EasyAccess that the program/script/command succeeded if the return code is greater than the value specified by the **RETCODE=** keyword.

**SUCCEEDS\_IF\_LT** - Tells EasyAccess that the program/script/command succeeded if the return code is less than the value specified by the **RETCODE=** keyword.

**SUCCEEDS\_IF\_EQ** - Tells EasyAccess that the program/script/command succeeded if the return code is equal to the value specified by the **RETCODE=** keyword.

**SUCCEEDS\_ALWAYS** - Tells EasyAccess that the program/script/command always succeeds, regardless of it's return code.

**FAILS\_IF\_GT** - Tells EasyAccess that the program/script/command, failed if the return code is greater than the value specified by the **RETCODE=** keyword.

**FAILS\_IF\_LT** - Tells EasyAccess that the program/script/command failed if the return code is less than the value specified by the **RETCODE=** keyword.

**FAILS\_IF\_EQ** - Tells EasyAccess that the program/script/command failed if the return code is equal to the value specified by the **RETCODE=** keyword.

**FAILS\_ALWAYS** - Tells EasyAccess that the program/script/command always fails, regardless of it's return code.

Note that this is only useful in testing, since the pre- or post-processing will fail, causing EasyAccess to consider the transfer to have failed.

The syntax used to specify pre- and/or post-processing is shown in the following example:

```
TRANSFER=( name=...
sendclass=...
...

Unix example:
PRE_SEND=[ CMDLINE='sh -x myScript.ksh 2>err.out'
RETCODE=127
SUCCEEDS_IF_LT
]

Windows example:
POST_SEND=[ CMDLINE='C:\\MyPrograms\\cleanup.exe /log'
RETCODE=0
SUCCEEDS_IF_EQ
]
)
```

This transfer will execute a Unix shell script (myScript.ksh) before the Send step and will execute a Windows program (cleanup.exe) after the Send step.

- Added ability to retain the server-side filename for received files. This is done by specifying a client-side directory to receive the file(s) instead of a client-side directory and filename. That is, when receiving files, if you specify only the directory to receive into, the incoming files will retain the filenames from the server.
- Added single-task command-line keywords for use by browser plug-in version of program.
- Added ability to retain server-side connection while user views Query Mailbox screen. This minimizes the number of server connect attempts when the user selects server files to be downloaded or deleted. Currently, the connection is retained only for the EAFTP network style.
- For OpenVMS platform, added automatic version control for all ini-files and log-files written, so the number of versions on the file system is never more than three.
- For EAFTP network-style, modified Audit Detail screen to display all the file details and increased size of Description field to allow for the display of the ISA record for EDI data.
- Added support for certificate extensions to contain the EDI name and the Email Address of the participant.

### **2.1.69 Build 1.26 09/27/1999**

- Fixed a bug which prevented IEBASE keyword from working properly. The command-line program would complain it could not find the AdHoc transfer for the specified network when invoked using the IEBASE keyword or invoked using the iebase.exe program.

### **2.1.70 Build 1.25 08/09/1999**

- Fixed a bug which caused a Receive transfer to cease execution when it encountered no files to receive (IGN network style only).

### **2.1.71 Build 1.24 06/25/1999**

- Changes to compression utilities only to fix problems with certificate file access error-handling during compression and a problem with the AUTOEXT parameter in decompression where the output-file numbering as not consecutive.

### **2.1.72 Build 1.23 06/10/1999**

- Minor base-code changes to support SCO Unix (Version 3.2 and higher).
- Added license agreement (file license.txt). It can be viewed from the Help menu item (GUI only) or via your editor of choice.
- For IGN network style, fixed problem where the SITE EDIALIASONLY 1 command was not being issued in all required situations, causing ENDEDI to fail for ZZ-qualified headers.
- For IGN network style, added default behavior when key fields are missing from the Common Data Header (CDH). The CType field is used to determine if the file in the mailbox is an EBCDIC or Binary file, and the DFormat field indicates if the file is an EDI file or not. If these fields are missing, then the default behaviour is to treat the file as an EBCDIC file (and hence to download it in ASCII mode unless it is also an EDI file), and to treat the file as non-EDI.
- For IGN network style, added support for embedded spaces in the Class field, fixing a problem with the Query display and download (receive) of files with embedded spaces in the Class field.
- For IGN network style, added COMPRESS site command when sending compressed non-EDI, non-filtered data. This causes the CDH entry created for the file to reflect the fact that the file is compressed using the btrade, Inc. compression utilities.
- Fixed problem for Windows NT version which caused the command-line arguments to be incorrectly parsed if RAS (Remote Application Services) is not installed.
- Small tweak to GUI to cause FTP Transfer screen to keep the most recent text visible within the display window.
- Fixed a problem with use of Passphrase Location when generating a certificate request (for those network styles with Security/Registration on the Main Menu). The problem resulted in a failure to Receive Runtimes due to an 'rimport' failure after the runtimes file was successfully received.
- Added large-file support for the Sun Solaris 2.6 version. Now you can send and receive files of virtually unlimited size.
- Added support for the following new keywords for command-line usage:

**GENKEYS** - Creates a certificate request and, if the Security Network is configured, automatically sends the certificate request to the designated mailbox;

e.g. ea2kw95c GENKEYS  
PASSLOC=

- Allows the specification of a passphrase location for splitting the private key data across files/disks/machines. This feature enhances the security of your private key(s);

e.g. ea2kw95c GENKEYS PASSLOC=a:\\passkey.txt Works with GENKEYS keyword only.  
RECEIVE\_RUNTIMES

- If the Security Network is configured, automatically runs a Receive Transfer to fetch your runtimes and imports them allowing for easy distribution of your X12.58 or SSL certificates/runtimes; if the Security Network is not configured, prompts for the location of the runtimes file which was distributed to you via another media;

e.g. ea2kw95c RECEIVE\_RUNTIMES

Removed the 'Revoke' button from the Security/Registration screen since it's intended functionality was never implemented.

### **2.1.73 Build 1.22 04/13/1999**

- Added support for the IBM Information Exchange Common Data Header (CDH), and support for specifying a Class when sending EDI files to the IBM Information Exchange network-style. The CDH specifies information about the transmission mode used to send the file to the Information Exchange Mailbox, and thereby allows EasyAccess to determine what transmission mode should be used to receive the file. Added limited support for filenames containing spaces in CDH name field.
- Fixed problem with randomization of private/public key generation on the Security/GenKeys screen (affected UNIX platforms only).
- Added support for IBM Global Dialer (Windows platforms only).
- Modified API to the ea2kxxx DLL for ease of use and consistency.
- The example program showing how to use the ea2kxxx DLL has been updated.
- Added support for Approval Code entry when receiving runtimes from multiple KeyManagers.
- Minor base-code changes to support DEC/Compaq OpenVMS (Version 7.1) and DEC/Compaq Unix (aka Tru64 Unix) (Version 4.0D).
- Added support for FTP Login UserIds containing the '@' sign, to facilitate communications to an FTP server via an intermediate Proxy Server. Once the login to the Proxy Server has been accomplished, the FTP UserId is truncated at the '@' sign, and normal login to the FTP server proceeds.
- The Setup Dialer dialog has been modified to allow the selection of no entry. Previously, once an entry had been selected, there was no way to select no entry - some non-blank entry had to be selected.
- A new network instance has been created for IBM Global Network users which support both SSL and compression. The default configuration, called the 'IGN-I/E SSL' configuration, is not configured to perform compression, and sends/receives files in ASCII mode by default. The new configuration, called 'IGN-I/E SSL WITH COMPRESSION' is setup to compress/decompress data and transmit data in Binary mode.

### **2.1.74 Build 1.21 02/18/1999**

- Added a 'purge' of server files older than 3 days for GEIS EDI\*SWITCH.

- Corrected a problem for GEIS EDI\*SWITCH that caused a Receive transfer to receive files that had been previously received.

### **2.1.75 Build 1.20 01/20/1999**

- Added support for the GEIS EDI\*Switch network style.
- Added support for DATA-GUARD network style which performs secure compression/decompression on the client machine (no file transfer to a remote server).
- Fixed problem with IBM Information Exchange network style which was causing the ReceiveEDI option on any transfer to receive all files in the user's mailbox instead of only the EDI files in the mailbox.
- Fixed problem with MCI EDI\*Net logon (use of ACCT keyword).
- Fixed problem with MCI EDI\*Net Edit Transfer screen (because MCI EDI\*Net requires a session to either send only, receive only, or receive and then send, the EasyAccess 2000 application will now support either send or receive within in a single transfer).
- Added Compression Options and Decompression Options screens accessible from Edit Transfer screen. These new screens let the user view and modify the current settings for file compression/decompression on a per transfer basis.
- Added license agreement to the Help item on the menu bar.
- Implemented a '60-day rule' for determining year displayed in the Query Mailbox function for the FEDEXNET and GEIS EDI\*EXPRESS network styles, since these networks do not provide this data. The 60-day rule assumes the correct year is the prior year if the given month is more than two months ahead of the current month.
- Now dynamically loading the Remote Access Services (RAS) dll which provides for remote dial capability within EasyAccess 2000. This allows clients without RAS to run the application.
- Corrected problem with EAFTP network style so that the correct transmission mode (ASCII or Binary) is used when receiving files from the server.
- X12.58 security only: Now using the alias.tbl (alias table) located in the runtime sub-directory (instead of the security sub-directory).
- X12.58 security only: All secure-compressed files will be assumed to be filtered and will be transmitted in ASCII mode.

### **2.1.76 Build 1.19 11/18/1998**

- Fixed problem with decompression option Auto-Extent for EDI files.
- Fixed problem with ReceiveEDI for Sterling Commerce network style.

### **2.1.77 Build 1.18 11/15/1998**

- Added transfer-restart capability for all networks. This feature allows a failed transfer or a group of transfers to be restarted at the point of failure, and includes file-level restart for the EAFTP network style. Please see the EasyAccess documentation for details.
- Now using version 4.4 of btrade, Inc. compression/decompression software.

- Added support for MCI EDI\*Net network style.
- Added support for SendEDI using the EAFTP network style.
- Fixed a number of minor problems for the Sterling Commerce network style.
- Fixed a number of minor problems for the GEIS EDI\*Switch network style.
- Corrected Query Archive Mailbox screen display so that the file downloaded is not removed from the displayed list after the download is completed.

### **2.1.78 Build 1.17 10/09/1998**

- Added 'Change Password' capability for FedexNet.
- Fixed problem with FTP Setup screen which caused program crash when the screen was closed by clicking the 'X' in the top right of the title bar.
- Changed default behaviour of the Command-Line version of EasyAccess so when no transfers are specified on the command-line or in a command-file it will automatically execute any transfers set up in the GUI version with the 'Save' option on the Run Stored Transfers screen (as per the EasyAccess documentation).

### **2.1.79 Build 1.16 10/05/1998**

- Fixed a problem in Status/View Audit Logs, which caused a problem in the AIX and HPUX versions (only) of EasyAccess.
- Fixed a problem in Query Mailbox for the EAFTP network on all Motif platforms of EasyAccess which caused the full path-name of the files in the user's mailbox to display, rather than just the filename. This was inconvenient, since the field width on the Query Mailbox screen is frequently too small to allow the full file-spec to fit.

### **2.1.80 Build 1.15 09/30/1998**

- Added DELETE\_AFTER\_SEND keyword support for Send Transfers (all networks) to support the deletion of source files after they have been successfully sent to the server. This keyword appears in the DEFAULT\_SENDFPARMS section in the easyacc.ini file for each network style, and in the 'SP' section for each transfer in the exfer.ini file. The default value for each network style is currently 'N' for this new keyword. The value for this keyword may be specified during transfer-creation using the command-line version of EasyAccess by adding DELETE\_AFTER\_SEND=Y to the transfer definition (i.e. the part inside the parentheses) or may be specified manually by editing the easyacc.ini or exfer.ini file(s). The value for this keyword is currently not accessible from the GUI version of EasyAccess.
- Added Query Mailbox Archive functionality for EAFTP and GEIS\_SWITCH network styles. This lets users view and re-download files which have been previously downloaded from the server. This functionality has been added to the FEDEX network style but is currently not enabled, pending review by Federal Express.
- Modified the logic used for EAFTP Receive Transfers so that, if an error occurs during the download or decompression, the file is not marked on the server as having been downloaded.
- Cleaned up the Audit Reports for the EAFTP network; the column headings and detail field labels were incorrect in a couple of places, and the date date-filter did not work correctly.

- Added support for non-RFC compliant form of the STOU FTP command. The RFC-compliant form of the STOU command is also supported. Some servers do not follow the RFC spec for the Store-Unique command and require a filename (like the STOR command). In addition, some firewalls will not allow the non-compliant form of the STOU command to pass. A new keyword, SUNIQUE, was created for each network section to specify how (or if) the STOU command is to be issued for each server. Allowed values are:  
  
**SUNIQUE=0** - Tells EasyAccess to use the STOR command and not use the STOU command - this applies to the majority of FTP servers;  
**SUNIQUE=1** - Tells EasyAccess to use the STOU command with a filename, like the STOR command - this is the non-compliant form.  
**SUNIQUE=2** - Tells EasyAccess to use the STOU command without a filename in the manner specified by RFC 959 for FTP. FEDEXNET users should have SUNIQUE=2 in the FEDEXNET and FEDEXNET X12.58 sections of their easyacc.ini file.
- Changed the manner in which EasyAccess logs into the server to minimize the number of logins which occur when multiple transfers are executed in one program run.
- Fixed a problem with the IEBASE option for the command-line version of EasyAccess: the first time the IEBASE option was used the MULTIFILE flag was not being set correctly to tell EasyAccess to use the bexfer.ini file instead of the exfer.ini file for the definitions of the transfers to be run.

### **2.1.81 Build 1.14 09/24/1998**

- Added network style 'STERLING-COMMERCE' to support Sterling's COMMERCE:Network FTP server and added a default STERLING-COMMERCE network instance to the base EasyAccess product configuration.
- Added support for MVS command-line client.
- Fixed problem in the command-line version when a Receive transfer was defined on the command-line or in a command-file and the RECEIVE= keyword specified a filename but no path information.
- Changed all text with 'IGN' to 'IBM Global Network' per request from IBM.
- Cleaned up logging so that all logs are written to the INIPATH directory tree when EasyAccess is invoked with the INIPATH= command-line argument.
- Fixed a problem when importing certificates from a local file; the Cancel button now works!
- Added the FTP QUIT command's response to the AdHoc Transfer and Stored Transfer screens.
- Modified the construction of security transfers (Send Cert Req and Receive Runtimes) to accommodate the mailbox-based approach used by the EAFTP network style.
- Corrected a problem with the use of UserID and Class as filters for Receive Transfers for the EAFTP network style.
- Corrected display of Received Date field for EAFTP network Audit Reports.
- Modified product packaging for UNIX platforms to use the bTrade, Inc. self-extracting executable approach instead of a tar-file.

**2.1.82 Build 1.13 09/21/1998**

- Fixed a bug which prevented wild-carded files from being properly processed in a Send transfer.

**2.1.83 Build 1.12 09/12/1998**

- Improved the Help for parsePFX utility (IBM Global Network only)
- Changed FTP Setup, Dial Setup, AdHoc Transfer and Run Stored Transfer screens so that non-editable fields now display with a grey background.

**2.1.84 Build 1.11 09/07/1998**

- Further improvements to parsePFX utility error handling (IBM Global Network only).
- Now handling case of import of certificates from local file system

**2.1.85 Build 1.10 09/02/1998**

- Corrected problems with IEBASE functionality for command-line version.
- Improved error reporting for parsePFX utility program (IBM Global Network only).
- Improved handling of key-files and security text tables to handle the addition of symmetric key functionality in a later release.
- Modified data presentation in Dial Setup window so that non-editable fields appear greyed-out.

**2.1.86 Build 1.9 08/24/1998**

- Fixed a variety of problems involving the creation and editing of transfers by beefing up the validation of transfer data at the time the transfer is created/edited.
- Major revision of command-line argument processing to include support for a new keyword set and command-file processing (see doc).
- Added support for self-flushing log files, so that nothing is lost in the log in the event of a power failure, crash, etc. This feature is enabled by prefixing the log-level with a dash in the IDENTIFY section in the easyacc.ini file. Since it much less efficient, this mode of logging is not recommended for general use.
- Modified the Retrieve Audit Screen to show the format of the date fields and to allow user to tab to End Date field.
- Modified data presentation on several screens so that non-editable fields appear grayed-out.
- Added 'Are You Sure' message boxes to the delete functionality for transfers (Edit Transfer) and server files (Query Mailbox).
- Corrected problem with the Shift and Control keys (Windows version) for selecting multiple transfers on the Run Stored Transfer screen.
- Added 'Add All' and 'Remove All' buttons on the Run Stored Transfer screen.

- Fixed 'autoextent' logic so that the first file created retains the user-specified file extension. Only subsequent files will have their file-extensions modified with '.001', '.002', and so on.
- (IBM Global Network only:) Fixed a problem which prevented both Sent and Received audit logs from being generated together.
- The EasyAccess Client User's Guide and Reference documentation has been largely revised; users are encouraged to download the new doc from the bTrade, Inc. web site at <http://www.bTrade.com/>

#### **2.1.87 Build 1.8 07/27/1998**

- Added support for GEIS EDI\*Switch FTP server (network style GEIS\_SWITCH in the easyacc.ini file)

#### **2.1.88 Build 1.7 07/21/1998**

- For IBM Global Network: fixed problem retrieving Audit Logs when a date range is specified.
- Added vertical scroll-bar to Setup Dialer entry-selection list

#### **2.1.89 Build 1.6 07/17/1998**

- First GA release of Version 6.4
- Consolidated network attributes in pre-defined ini-file entries, including default APRF/Class lists, and default compression and decompression parameters, so that each network is now completely pre-configured, including SSL and X12.58 variants, where applicable
- Consolidated Stored Transfers into network-specific ini-file entries so that the transfers are network-specific. This allows for the pre-configuration of the Security- and Maintenance-related transfers for each network.
- Various minor bug fixes found during our QA testing.

#### **2.1.90 Build 1.5 07/06/1998**

- First release of Version 6.4, including the following features:
  - Dial and Auto-Dial using your DialUp Networking (Windows 95/NT).
  - Use of the native Windows Help facility (Windows 3.1/95/NT).
  - Execution of FTP transfers via background threads (improves response time)
- Support for multiple instances of a chosen network allowing for selection from the drop-down list of networks. This allows for the configuration of several network instances with different IP addresses or the use of an SSL server for some transfers and a non-SSL server for others.
- Release of a full-functioned command-line version of EasyAccess which is tailored to be run from 'batch' files or from within other programs.
- Meaningful exit codes; especially relevant for the command-line version.

- Integration of the 6.0, 6.1, and 6.2 versions into a single product to simplify the use of EasyAccess in multiple environments.
- Integration into the EasyAccess program of the ParsePFX utility for importing IBM's Information Exchange certificates (IBM Global Network only).

#### **2.1.91 Build 1.4 05/22/1998**

- Now compress source files into the 'temp' directory instead of the 'outgoing' directory to prevent the overwriting of source files in the 'outgoing' directory.

#### **2.1.92 Build 1.3 04/15/1998**

- Changed default receive mode to ASCII when COMPRESS=Y and either SECURE=Y or FILTER=Y to match the corresponding default send mode.

#### **2.1.93 Build 1.2 03/23/1998**

- Fixed bug in Run Stored Transfers action (only ReceiveEDI affected)
- Fixed Session Status, Status Summary and Status Detail to show the 'End FTP Session Log' message at the actual session end.
- Added FTP-layer logging to Session Detail

#### **2.1.94 Build 1.1 03/18/1998**

- Added EDI file display on Query screen.
- Added Binary/Ascii download mode check-box on Query screen
- Added intelligent default send/receive modes (Binary/Ascii)
- Changed EA62 COMPRESS default to N (is Y for 60/61)
- Changed EA61 SECURE default to Y (is N for 60/62)
- Added Auto-install for keys when AUTOUPDATERUNTIME is Y

#### **2.1.95 Build 1.0.0 03/09/1998**

- Added scrollable release notes in About box
- Added site probe command for EDI SEND to IBM Global Network
- Added support for CPFTP digital certificate distribution