MakeAFP

HOST2TXT Transform
User's Guide

Version 4.0

This edition applies to HOST2TXT Transform.

MakeAFP welcomes your comments and suggestions. You can send your comments and suggestions to:

support@makeafp.com

When you send information to MakeAFP, you grant MakeAFP a non-exclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

© MakeAFP 2026. All Rights Reserved

Contents

Preface **1**

About this User's Guide 1

Conventions Used in This User's Guide 1

Chapter 1. Overview **3**

Functions at a Glance 3

Chapter 2. Installing HOST2TXT Transform for Windows **4**

HOST2TXT for Windows Prerequisites 4

Installing HOST2TXT on Windows 4

Applying License of HOST2TXT Transform Retail Version for Windows 4

Chapter 3. Installing HOST2TXT Transform for Linux **5**

Installing HOST2TXT on Linux 5

Applying License of HOST2TXT Transform Retail Version for Linux 5

Uninstalling HOST2TXT on Linux 5

Chapter 4. HOST2TXT Command **6**

HOST2TXT Command Syntax 6

HOST2TXT Command Examples 9

Appendix A. EBCDIC Codepage Summary **10**

Appendix B. ASCII Codepage Summary **12**

Appendix C. Mixed SBCS/DBCS and Unicode Codepages Summary **14**

Preface

This User's Guide describes the functions, commands, and parameters associated with HOST2TXT Transform.

About this User's Guide

This User's Guide provides information about using HOST2TXT Transform, it helps you:

- Plan for transforming data from Adobe PDF (Portable Document Format) to AFP (Advanced Function Presentation) format.
- Install and configure HOST2TXT Transform.
- How to use HOST2TXT command.

The information in this user's guide is for users who install and use HOST2TXT Transform.

Conventions Used in This User's Guide

This User's Guide uses consistent conventions for the following:

- Highlighting
- Syntax notation

Highlighting

This User's Guide uses the following highlighting conventions:

- **Bold** Identifies commands, keywords, and other items, whose names or values are predefined by HOST2TXT Transform or must be entered as-is.

- ***Italic*** Identifies parameters whose actual names or values you supply.

Syntax Notation

This User's Guide uses the following syntax notation:

- Italics within a command represent variables for which you must supply a value. For instance:

  -i ***inPath***

means that you need to replace the variable ***inPath*** with a value that represents any valid file system directory.

- Do not enter the following symbols as part of the command:

Vertical bar	|
Braces	{ }
Brackets	**[ ]**
Underscore	_
Ellipsis	…

The above symbols have the following meanings:

- A vertical bar, |, between values, indicates that you can only enter one of the values with the command.

- Braces, { }, around values indicate a required value, you must select one of the mutually exclusive values.

- Brackets, [ ], around parameters indicate that they are optional.

- An underscore, _, indicates the default value, which HOST2TXT uses if you do not specify the parameter with a non-default value.

- An ellipsis, …, following command or set of commands indicates the command or set of commands can be repeated.

Chapter 1. Overview

HOST2TXT provides consistent, efficient, and high-performance transformation for the transactional high volume IBM mainframe line data to the text to be used for open system.

HOST2TXT can process the high volume line data from IBM systems, such as from IBM z/OS systems, Content Manager OnDemand, as well as via IBM FTP, NFS, PSF Download, etc.

Functions at a Glance

HOST2TXT Transform provides the following advanced features, functions, and capabilities:

- Able to convert high-volume IBM mainframe line data in super-performance.
- Able to handle the line data carriage control in ANSI and Machine codes.
- Easy to specify the input data and output data encodings.
- High-performance conversion from IBM mainframe line data encoding in EBCDIC, SBCS-HOST/DBCS-HOST, to the text data encoding in ASCII, SBCS/DBCS for PC/Unix, and Unicode UTF-8.
- Able to process the different input data formats, such as IBM variable record length format, record delimited format, and fixed record length.
- Able to split output text data by the IBM generic index values.
- Able to split and output text files into the dynamic destination directories based on the IBM generic index values.

Chapter 2. Installing HOST2TXT Transform for Windows

This chapter provides information on the server requirements and installation of the HOST2TXT Transform retail version for Windows.

HOST2TXT for Windows Prerequisites

Here are the prerequisites to run HOST2TXT Transform for Windows:

1. Windows 7 or above, 64-bit.
2. Windows Server 2008 or above, 64-bit.

Installing HOST2TXT on Windows

To install HOST2TXT Transform on a Windows server or workstation:

1. Log on to the Windows system as an administrator.
2. Run HOST2TXT Transform setup package you received.
3. Follow the instructions on the installation screens to install the package, the destination folder is **c:\HOST2TXT**.

Applying License of HOST2TXT Transform Retail Version for Windows

Installing Soft-license Key

1. Open Windows Command Prompt, run command HOST2TXT.exe**.**
2. On HOST2TXT Transform Licensing GUI, copy the Serial Number, and then paste into an email to be sent to MakeAFP Support at support@makeafp.com to request a software license key for your system.
3. Once a HOST2TXT Transform License key is received, you need to run HOST2TXT to apply the license key.

Chapter 3. Installing HOST2TXT Transform for Linux

This chapter provides installation information of the HOST2TXT Transform retail version for Linux.

Installing HOST2TXT on Linux

To install HOST2TXT Transform on a Linux server:

If install on the HOST2TXT default installation path **/usr/share/host2txt**, type the following command:

```
sudo sh host2txt_install.bin
```

If install on a user-defined specific path, type the following command:

```
sudo sh host2txt_install.bin your_host2txt_path_name
```

Applying License of HOST2TXT Transform Retail Version for Linux

HOST2TXT Transform for Linux provides the soft-license key.

Installing Soft-license Key

1. Type command **host2txt**, copy the Serial Number generated, and then paste into an email to be sent to MakeAFP Support at support@makeafp.com to request a software license key for your system.
2. Once your HOST2TXT Transform License key file **host2txt.lic** is received, copy it into your HOST2TXT installation path, default path is:

   ```
   /usr/share/host2txt
   ```

Uninstalling HOST2TXT on Linux

To uninstall HOST2TXT Transform on a Linux server:

If installed HOST2TXT on the default installation path `/usr/share/HOST2TXT`, type the following command:

```
sudo sh host2txt_install.bin --uninstall
```

If installed HOST2TXT on a user define specific path, type the following command:

```
sudo sh host2txt_install.bin your_host2txt_path_name --uninstall
```

Chapter 4. HOST2TXT Command

HOST2TXT Transform provides an easy-to-use high-performance command-line program, to enable you quickly transform high-volume IBM mainframe line data to the text data for the open systems.

HOST2TXT Command Syntax

HOST2TXT command is designed to be easy to use, only some of the basic flag parameters are needed to control the HOST2TXT Transform.

Command syntax

host2txt [*hostFile*] [*txtFile*] [*options*]

Parameters

hostFile `
Optional, specify the fully qualified filename of the input IBM mainframe line data file, wildcard characters(* and ?) are allowed for the multiple input files, alternatively, you may transform all input files from a specific path specified by the flag parameter -i *inPath.*

txtFile
Optional, specify the filename of output text file, if it is not specified then its filename forms by the input file base name with filename extension *.txt, and output to the current path.

-i ***inPath***
Optional, specify the input path from which to process all IBM mainframe line data files, default is the current path if it is specified.

-ln ***n***
Optional, specify the output text line delimiter, valid values are:

```
0          Default value, on Windows systems: \r\n; On Linux systems: \n
1          \n for Linux/Unix systems
2          \r\n for Windows/DOS systems
```

-o ***outPath***
Optional, specify the output path of the text files, default is the current path if it is specified.

-O ***dynamicOutPath***
Optional, specify the convention to construct the dynamic output directories for the splitting of text data.

The following keywords are supported for auto-split text data into the dynamic directories:

`(yy), (yyyy)`	by the date value of current year
`(mm)`	by the date value of current month
`(dd)`	by the date value of current day
`(HH)`	by the time value of current hour
`(MM)`	by the time value of current minute
`(SS)`	by the time value of current second
`(idx`***n***`,`*column*`,`*length*`)`	by the part of generic index value string capturing from a generic index. ***n*** indicates the order number of index attribute names specified by the command argument flag **-s**; *column* specifies the byte offset from the beginning of the index value string, a value of "1" refers to its first byte; *length* specifies the number of bytes to be captured.
`(idx`***n***`,`*fieldNo*`,`*fieldCol*`,`*fieldLen*`,"`*delimiters*`")`	by the part of a generic index value string captured from an AFP index. ***n*** indicates the order number of index attribute names specified by the command argument flag -s; *fieldNo* specifies the order number of the data fields to be used to construct the index value string; *fieldCol* specifies the byte offset from the beginning of the data field, a value of "1" refers to its first byte; *fieldLen* specifies the maximum number of bytes to be captured; *delimiters* specifies the characters to be used to indicate the end of the data fields.

Example:

Assuming today is May 30, 2023, with e:`\visa\(yyyy)\(mm)\(dd)` dynamic output directory definition, text data will be split into the following directory:

```
e:\visa\2023\05\30\
```

Assuming command-line flag parameter `-s INVOICE` is specified, INVOICE value string forms by the convention *x-mm-dd-yy-xxxx-accountNo, and* dynamic output directory definition is defined as:

```
f:\visa\(idx1,1,1)\(idx1,9,2)\(idx1,3,2)\(idx1,6,2)\(idx1,12,4)
```

Text data with INVOICE index value A-05-16-12-0301-2340561670 will be split into the directory:

```
f:\visa\A\12\05\16\0301\
```

Assuming command-line flag parameter `-s INVOICE` is specified, INVOICE value string forms by the convention *x-mm-dd-yy-xxxx-accountNo, and* dynamic output directory definition is defined as:

```
f:\visa\(idx1,1,1)\(idx1,9,2)\(idx1,3,2)\(idx1,6,2)\(idx1,12,4)
```

Document with INVOICE index value A-05-16-12-0301-2340561670 will be split into the directory:

```
f:\visa\A\12\05\16\0301\
```

Assuming command-line flag parameter `-s "Account No"` is specified, its index value string forms by the convention *xxxx-yymmdd-accountNo, xxxx* field maximum length may be up to 4 bytes, and dynamic output directory definition is defined as:

```
f:\debitCard\(idx1,1,1,4,"-")\(idx1,2,1,6,"-")\
```

Document with "Account No" index value A12-120516-2340561670 will be split into the directory:

```
f:\debitCard\A12\120516\
```

-cc
Optional, indicate that the input line data contains the carriage control codes.

-v
Optional, specify that the input line data is in IBM variable record length format, it is the default value if other input data format is not specified.

-d
Optional, specify that the input line data record is delimited by IBM mainframe host line-feed hex code x'25', PC carriage-retune & line-feed hexcode x'0d0a', or Unix line-feed hexcode x'0a'.

-l *nnnn*
Specify that the input line data is in fixed record length *nnnn* bytes.

-list
List the encoding codepage names.

-f *fromCP*
Specify the codepage ID of input line data encoding.

-r
Optional, replace the existing output text file, the default is to add the filename extension with a sequential number to retain the existing text file.

-rc "*char*"
Optional, replace invalid characters from the generic index attribute values with a "*char*" character for the splitting of text output files, the default "*char*" value is blank.

-t *toCP*
Optional, specify the codepage ID of the output text data encoding, the default is UTF-8 if it is not specified.

-s *indexes*
Optional, split line data pages by the generic indexes exported from IBM CMOD, the output text file base name will be formed by the index values in the order of index values attribute (GROUP_FIELD_NAME) names exported from IBM CMOD. The default text filename delimiter is dot (.) to separate filename fields, the custom delimiter can be specified in the syntax indexName[delimiter], and invalid characters with filename can be replaced in the syntax {chars, newChar}, for example:

```
"indexName1{'/,_}[_];indexName2{ ,_}[_];indexName3{%*,-}[_]"
```

-sd
Optional, scan and read all sub-folders.

-si
Optional, read IBM line data stream from the standard-input.

-so

Optional, write text data stream to the standard-output.

-ss "***char***"
Optional, set the delimiter used to append the sequence number suffix to the text output filename in the case if the same split text output filename exists, the default delimiter is the underscore character (_).

HOST2TXT Command Examples

The following example shows how to use the HOST2TXT command to transform a line data file exported from IBM Content Manager OnDemand, which is encoded in EBCDIC encoding codepage 500 with carriage-control code, output text will be encoded in UTF-8 with filename extension *.txt.

```
host2txt REPORT.0.TWMD1ORPTGRP.ATWODBMD1.out -cc -f 500
```

Appendix A. EBCDIC Codepage Summary

Codepage ID	Description	Encoding
37	USA/Canada - CECP	EBCDIC
38	US-ASCII Character Set	EBCDIC
259	Symbols, Set 7	EBCDIC
260	Canadian French - 116	EBCDIC
273	Germany F.R./Austria- CECP	EBCDIC
274	Belgium - CECP	EBCDIC
275	Brazil - CECP	EBCDIC
276	Canada (French) - 94	EBCDIC
277	Denmark/Norway - CECP	EBCDIC
278	Finlandd/Sweden- CECP	EBCDIC
280	ITALY- CECP	EBCDIC
281	Japan (Latin) - CECP	EBCDIC
282	Portugal - CECP	EBCDIC
284	Spain/Latin America - CECP	EBCDIC
285	UNITED KINGDOM - CECP	EBCDIC
286	Austria/Germany F.R., Alt (3270)	EBCDIC
287	Denmark/Norway, Alternate (3270)	EBCDIC
288	Finland/Sweden, Alternate (3270)	EBCDIC
289	Spain, Alternate (3270)	EBCDIC
290	Japan (Katakana)	EBCDIC
293	APL (USA)	EBCDIC
297	France - CECP	EBCDIC
310	Graphic Escape APL/TN	EBCDIC
361	International Set 5	EBCDIC
363	Symbols, Set 8	EBCDIC
382	Austria, Germany, Japan	EBCDIC
383	Belgium	EBCDIC
384	Brazil	EBCDIC
385	Canada (French)	EBCDIC
386	Denmark/Norway	EBCDIC
387	Sweden/Finland	EBCDIC
388	France, Japan	EBCDIC
389	ITALY, Japan (Italian)	EBCDIC
390	Japan (Latin)	EBCDIC
391	Portugal	EBCDIC
392	Spain/Philippines	EBCDIC
393	Latin America (Spanish)	EBCDIC
394	U.K., Austral., IRE., H.K., N.Z.	EBCDIC
395	United States, Canada (English)	EBCDIC
420	Arabic Bilingual	EBCDIC
423	Greece - 183	EBCDIC
424	Israel (Hebrew)	EBCDIC
500	International #5	EBCDIC
803	Hebrew Character Set A	EBCDIC

829	Math Symbols	EBCDIC
836	Peoples Republic of China (PRC)	EBCDIC
838	Thai - EBCDIC	EBCDIC
870	Latin2 Multilingual	EBCDIC
871	Iceland - CECP	EBCDIC
875	Greece	EBCDIC
880	Cyrillic Multilingual	EBCDIC
889	Thailand	EBCDIC
892	OCR - A	EBCDIC
893	OCR - B	EBCDIC
905	Latin3 Multilingual	EBCDIC
918	Urdu - EBCDIC	EBCDIC
924	Latin 9 EBCDIC	EBCDIC
1002	DCF REL 2 Compatibility	EBCDIC
1003	U.S. Text Subset	EBCDIC
1025	Cyrillic Multilingual	EBCDIC
1026	Latin5	EBCDIC
1027	Japanese (Latin) Extended	EBCDIC
1028	Hebrew Publishing	EBCDIC
1032	MICR, E13-B Combined	EBCDIC
1033	MICR, CMC-7 Combined	EBCDIC
1039	GML List Symbols	EBCDIC
1068	Text With Numeric Spacing	EBCDIC
1069	Latin4 EBCDIC	EBCDIC
1087	Symbols, Adobe	EBCDIC
1091	Symbol Set 7, Modified	EBCDIC
1093	IBM LOGO	EBCDIC
1110	Latin2 Multilingual	EBCDIC
1112	Baltic - Multilingual, EBCDIC	EBCDIC
1122	Estonia, EBCDIC	EBCDIC
1123	Cyrillic, Ukraine EBCDIC	EBCDIC
1130	Vietnamese EBCDIC	EBCDIC
1132	Lao EBCDIC	EBCDIC
1140	USA, Canada ECECP	EBCDIC
1141	Austria, Germany ECECP	EBCDIC
1142	Denmark, Norway ECECP	EBCDIC
1143	Finland, Sweden ECECP	EBCDIC
1144	Italy ECECP	EBCDIC
1145	Spain, Latin America ECECP	EBCDIC
1146	UK ECECP	EBCDIC
1147	France ECECP	EBCDIC
1148	International ECECP	EBCDIC
1149	Iceland ECECP	EBCDIC
1153	Latin2 Multilingual with Euro	EBCDIC
1154	Cyrillic Multilingual with euro	EBCDIC
1155	Turkey with euro	EBCDIC
1156	Baltic - Multi with euro	EBCDIC
1157	Estonia with euro	EBCDIC
1158	Cyrillic, Ukraine with euro	EBCDIC
1160	Thailand EBCDIC with Euro	EBCDIC
1164	Vietnamese EBCDIC with euro	EBCDIC
1166	Cyrillic, Multi with euro	EBCDIC
1300	GENERIC BAR CODE/OCR-B	EBCDIC

Appendix B. ASCII Codepage Summary

Codepage ID	Description	Encoding
367	ASCII	ASCII
437	Personal Computer	ASCII
813	Greece - ISO/ASCII 8-Bit	ASCII
819	Latin1 ISO/ANSI 8-BIT	ASCII
848	PC, Cyrillic, Ukraine with Euro	ASCII
849	PC, Cyrillic, Belo Russian Euro	ASCII
850	PC Multilingual	ASCII
851	Greek - Personal Computer	ASCII
852	Latin2 Multilingual PC	ASCII
853	Latin3 Personal Computer	ASCII
855	Cyrillic - Personal Computer	ASCII
856	Hebrew - Personal Computer	ASCII
857	Latin5 PC	ASCII
858	PC - Multilingual with Euro	ASCII
860	Portugal - Personal Computer	ASCII
861	Iceland - Personal Computer	ASCII
862	Hebrew - Personal Computer	ASCII
863	Canadian French - PC	ASCII
864	Arabic - Personal Computer	ASCII
865	Nordic - Personal Computer	ASCII
866	Cyrillic #2 - Personal Computer	ASCII
867	Israel - Personal Computer	ASCII
869	Greece - Personal Computer	ASCII
872	Cyrillic PC with Euro	ASCII
874	Thai - Personal Computer	ASCII
876	OCR-A ASCII	ASCII
877	OCR-B ASCII	ASCII
895	Japan Latin EUC	ASCII
896	Japan Katakana EUC	ASCII
897	Japan PC #1	ASCII
899	Symbols, Set 7 ASCII	ASCII
901	PC, Baltic - Multilingual with Euro	ASCII
902	8-bit Estonia with Euro	ASCII
903	Peoples Republic of China - PC	ASCII
904	Republic of China (ROC) - PC	ASCII
910	APL ASCII	ASCII
912	Latin2 ISO/ANSI 8-BIT	ASCII
913	Latin 3, ISO/ASCII	ASCII
914	Latin4 ISO/ANSI 8-BIT	ASCII
915	Cyrillic ISO/ASCII 8-Bit	ASCII
916	Hebrew ISO/ASCII 8-Bit	ASCII
920	Latin5 ISO/ANSI 8-BIT	ASCII
921	PC, Baltic - Multilingual	ASCII
922	Estonia PC	ASCII

923	Latin 9	ASCII
1004	IBM PC Desktop Publishing	ASCII
1006	Urdu ISO - PC	ASCII
1008	Arabic ISO/ASCII 8-Bit	ASCII
1029	Arabic Extended ISO/ACSII 8-Bit	ASCII
1038	Symbols, Adobe ASCII	ASCII
1041	Japanese Extended - PC	ASCII
1042	Simplified Chinese Extended - PC	ASCII
1043	Traditional Chinese Extended PC	ASCII
1046	Arabic Extended ISO/ASCII 8-Bit	ASCII
1088	Korean KS - PC	ASCII
1089	Korean ISO - PC	ASCII
1092	Symbol Set 7, Modified - PC	ASCII
1111	Latin2 ISO/ANSI 8-BIT	ASCII
1115	Chinese GB	ASCII
1116	Estonia - PC	ASCII
1117	Latvia - PC	ASCII
1118	Lithuania - PC	ASCII
1119	Lithuania, Russian - PC	ASCII
1124	Cyrillic, Ukraine ISO-8	ASCII
1125	PC, Cyrillic Ukrainian	ASCII
1129	Vietnamese ISO-8	ASCII
1131	PC, Cyrillic, Belo Russian	ASCII
1133	Lao ISO-8	ASCII
1139	Japan Alphanumeric Katakana	ASCII
1161	Thailand PC with Euro	ASCII
1162	Windows Thailand	ASCII
1163	Vietnamese ISO-8 with Euro	ASCII
1250	Latin 2 – Windows	ASCII
1251	Cyrillic – Windows	ASCII
1252	Latin 1 – Windows	ASCII
1253	Greece – Windows	ASCII
1254	Turkey – Windows	ASCII
1255	Israel – Windows	ASCII
1257	Latin 4 – Windows	ASCII
1258	Vietnamese – Windows	ASCII
1275	Apple Latin 1	ASCII
1276	Adobe PS Standard	ASCII
1277	Adobe PS ISO Latin 1	ASCII
1280	Apple Greece	ASCII
1281	Apple Turkey	ASCII
1282	Apple Central Europe	ASCII
1283	Apple Cyrillic	ASCII
5346	Latin 2 - Windows	ASCII
5347	Cyrillic - Windows	ASCII
5348	Codepage for 5348	ASCII
5349	Greece - Windows	ASCII
5350	Turkey - Windows	ASCII
5351	Israel - Windows	ASCII
5352	Codepage for 5352	ASCII
5353	Latin 4 - Windows	ASCII
5354	Vietnamese 4 - Windows	ASCII

Appendix C. Mixed SBCS/DBCS and Unicode Codepages Summary

Codepage ID	Description	Encoding
937	Traditional Chinese EBCDIC/DBCS-HOST	EBCDIC/DBCS (CP37/CP835)
939	Japanese EBCDIC/DBCS-HOST	EBCDIC/DBCS (CP1027/CP300)
943	Japanese ASCII/SJIS for PC/Unix	ASCII/SJIS (CP897/CP941)
950	Traditional Chinese ASCII/BIG5 for PC/Unix	ASCII/BIG5 (CP1114/CP947)
1363	Korean ASCII/KSC for PC/Unix	ASCII/KSC (CP1126/CP1362)
1364	Korean EBCDIC/DBCS-HOST	EBCDIC/DBCS (CP833/CP834)
1386	Simplified Chinese ASCII/GBK for PC/Unix	ASCII/GBK (CP1114/CP1385)
1388	Simplified Chinese EBCDIC/DBCS-HOST (GBK)	EBCDIC/DBCS (CP836/CP837)
13767	Simplified Chinese EBCDIC/DBCS-HOST (GB18030)	EBCDIC/DBCS (CP836/CP837K)
1208, 1209, 5304, 5305. 13496,13497, 17592, 17593, 65001, UTF8 UTF-8	Unicode UTF-8	Unicode UTF-8
1201, 13488, 13489, 17584, 17585, 21680, 21681, 25776, 25777, 29872, 29873, 61955, 61956, UTF16BE, UTF-16BE	Unicode UTF16 BigEndian	UTF16 Big Endian
1200, 1202, 1203, 13490, 13491, 17586, 17587, 21682, 21683, 25778, 25779, 29874, 29875, UTF16LE, UTF16-LE	Unicode UTF16 LittleEndian	UTF16 Little Endian

MakeAFP