

The Trainer's Friend, Inc.

---

6790 E. Cedar Ave., Suite 201  
Denver, Colorado 80224  
U.S.A.

Telephone: (303) 393-8716

E-mail [trainers@trainersfriend.com](mailto:trainers@trainersfriend.com)  
Internet [www.trainersfriend.com](http://www.trainersfriend.com)

# **File RePackager**

## **Installation and Customization Guide**

Version 1.1

**The following terms that may appear in these course materials are trademarks or registered trademarks:**

**Trademarks of the International Business Machines Corporation:**

**AS/400, IBM, Language Environment, MVS, TSO, WebSphere, z/OS, z/VM, z/Architecture, zSeries, z9, z10**

**Trademarks of Microsoft Corp.: Microsoft, Windows, Windows XP, Windows Vista**

**Trademark of American National Standards Institute: ANSI**

**Registered Trademarks of Institute of Electrical and Electronic Engineers: IEEE, POSIX**

**Registered Trademark of The Open Group: UNIX**

**Trademark of Sun Microsystems, Inc.: Java**

**Registered Trademark of Linus Torvalds: LINUX**

**Registered Trademark of Unicode, Inc.: Unicode**

**Trademarks held on behalf of World Wide Web Consortium: W3C, XHTML, XSL, WebFonts**

**Trademark of The Trainer's Friend, Inc.: File Repackager**

# Table of Contents

File RePackager from The Trainer's Friend .....	3
Obtaining the initial file .....	4
Unloading the Libraries .....	5
Running the Installation Verification Procedure .....	6
Notes .....	8
Messages and Codes .....	9
Abend Codes .....	10
Return Codes .....	11
Messages .....	12
Appendix: Code Page Support .....	25
CCSIDs .....	26
Codepages in File RePackager .....	30
Base-36 id's .....	31
Codepages supported in File RePackager .....	33
CodePage Tables used in File RePackager .....	35
Changing the Default Code Page .....	36
Modifying Existing Code Page Tables .....	37
Adding Support For a New Code Page .....	38

This page intentionally left almost blank

# Installing File RePackager

## File RePackager from The Trainer's Friend

- ◆ Obtaining the initial file
- ◆ Unloading the Libraries
- ◆ Running the Installation Verification Procedure
- ◆ Notes

## Obtaining the Initial File

**When File RePackager is ordered, we build a version just for your site and provide you with a single starter file. We can email it to you as an attachment or put it on a private page on our website for you to download (it's less than 1.5MB).**

**This file is named TTFU (for The Trainer's Friend Utilities). Upload TTFU to your z/OS system (binary mode). Presumably it will be under some high level qualfier (HLQ) so the file name on your z/OS system will be <hlq>.TTFU.**

## Unloading the Libraries

From ISPF option 6, or TSO READY, issue a receive command:

```
receive  inda(<hlq>.TTFU)
```

when you get these messages:

```
INMR901I Dataset xxxxx.xxxxx.xxxx from xxxxx on xxxx
INMR906A Enter restore parameters or 'DELETE' or 'END' +
```

reply by pressing 'Enter' to accept the default data set name ('TTFU.LIBRARY', prefixed by your TSO id as the hlq (only works if your TSO PROFILE indicates PREFIX(hlq))); or key in

```
dsn("fully_qualified_desired_data_set_name")
```

Issue this "receive" command against a member in the LIBRARY data set, to restore a load module library:

```
receive inda('<hlq>.TTFU.LIBRARY(LOAD)')
```

Issue this "receive" command against a member in the LIBRARY data set, to restore the library of source code members:

```
receive inda('<hlq>.TTFU.LIBRARY(ASM)')
```

At this point you have the three libraries needed to run and maintain the File RePackager:

```
<hlq>.TTFU.LIBRARY
<hlq>.TTFU.ASM
<hlq>.TTFU.LOAD
```

# Running the Installation Verification Procedure

To ensure the product is working in your environment, edit member FRPISV in <hlq>.TTFU.LIBRARY:

- \* On line 23, replace SCOMSTO with the high level qualfier used for the File RePackager data sets
- \* Provide a valid JOB statement at the front of the file
- \* Submit the resulting job and view the job output; you should see:

Near the top of the JES job log you should see something like:

*hh.mm.ss* JOBnnnnn +Base 36 value for 01140 is: 0VO.

At the end of the job, under TTFLOG, you should see:

```
TTFU000I FileRePackager Version v.r.m logging started.

TTFE001I CPU-id: cpuid. License-id: dddddd-ddddd.
TTFE002I This copy of the software is licensed for use only by xxx
TTFE002I         at the location known as: YYYYYYYYYYYYYY.
TTFE003I Use by any other organization or individual is illegal and
                                                expressly forbidden.
TTFE003I Anyone discovering an unauthorized use is requested to
        report it to The Trainer's Friend at 303-393-8716.

TTFE004I Copyright 2010 by The Trainer's Friend, Inc.
TTFE004I Call The Trainer's Friend at 303-393-8716 for support.

TTFU019I Input commands:
TTFU031I :   ofile fill=x'f3'

TTFU051I Run characteristics:
TTFU004I * Input file:  RECFM=FB, LRECL=      80, BLKSIZE= 27,920
TTFU008I * Output file: RECFM=FB, LRECL=     100, BLKSIZE=   100
TTFU064I * No code page translations will be done.
TTFU058I * All input data records will be processed.
TTFU052I End of run characteristics.

TTFU053I Processing Started.
TTFU054I Processing Ended.

TTFU065I Records read from input file:   144
TTFU066I Records written to output file: 144

TTFU900I Final completion code:   0
```

## Running the Installation Verification Procedure, continued

\* There will also be a SYSOUT file:

00001	Thornapple	Sebastian	A020290	Top banana	01030	01	333...33
00002	Crabapple	Thorny	A020390	Second banana	01030	02	333...33
00003	Pineapple	Timothy	0020490	Head honcho	01030	02	333...33
.							
.							
.							
01034	Clause	Anthony	M041996	Redder reader	08008	10	333...33
00580	Lipschitz	Sydney	B030795	Iron seeder	08008	10	333...33

Finally, call the phone number 303-393-8716 and give them the CPU-id and License-id from the job listing: this will confirm installation and activate your support (you can leave a voice message if you get a recording).

## Notes

**File RePackager is now ready to go. In <hlq>.TTFU.LIBRARY(\$README) is an inventory of the files available.**

**In this book we have provided a copy of the possible abend codes, user return codes, and messages users are likely to see.**

**At the end we provide detailed information about modifying the original codepage support.**

# Messages and Codes

Abend Codes

Return Codes

Messages

## Abend Codes

We have worked hard to prevent File RePackager from abending ("blowing up"), but some things are out of our control. So here is our general advice:

**System Abends** (Sxxx in job listing): These abends are likely due to errors in your JCL or utility control statements coding, so check your JCL and utility control statements carefully and look at the utility log file; if you can't resolve the problem, contact The Trainer's Friend by phone or email to talk about solving the issue.

Phone: 303-393-8716

Email: [steve@trainersfriend.com](mailto:steve@trainersfriend.com)

**User Abends** (Uddd in job listing): all user abends are generated by I/O errors against the various files:

U230 - I/O error on commands file

U231 - I/O error on log file

U232 - I/O error on data input file

U233 - I/O error on data output file

In the above cases, it's possible they were caused by non-hardware faults, so try to read from the files using some other program (IEBGENER or similar IBM utility) and, again, check your JCL for erroneous or conflicting parameters. Otherwise, talk to the people in your shop who are responsible for hardware error monitoring.

## Return Codes

At the end of every run, the utility emits a single integer, a return code, that provides a high level piece of information on how the run went:

<u>RC</u>	<u>Means</u>
0	No errors were detected, run was successful
4	A warning diagnostic was issued, but the utility was run using default or deduced values
8	An error was encountered that prevents running the actual copy / transformation; nonetheless, error checking continued on the input commands
12	An error occurred that prevented further processing
16	Unable to open the log file; the application terminates
17	Error closing the log file
1000	Unable to open the commands file
1001	Unable to open the input file
2001	Unable to open an HFS input file
1002	Unable to open the output file
1010	Unable to set FILEDATA=RECORD for output file
1020	Error closing the commands file
1021	Error closing the input file
1022	Error closing the output file

Note that open errors are usually due to incorrectly coded DD statements.



**TTFE007S Unable to obtain CPU id.**

The callable service CSRSI failed. This can happen on an older system; the system is not supported by the File RePackager.

**TTFE008W This license expires on *yyyy/mm/dd*; call The Trainer's Friend to update / renew.**

This warning message starts appearing about 30 days before the product license expires.

**TTFE009S This license expired on *yyyy/mm/dd*; call The Trainer's Friend to update / renew.**

The license has expired. The product will not run until you update, renew, or upgrade your copy.

**TTFU012S LRECL of commands file less than 80.**

The TTFIN file must be a sequential file (including member of PDS or PDSE, or a z/OS UNIX file) with records 80 bytes or longer.

**TTFU014I \* Input file: RECFM=xxx, LRECL= *nn,nnn*, BLKSIZE= *nn,nnn*  
or z/OS UNIX file.**

The input data file has been successfully opened.

**TTFU015S Input file RECFM VBS is not supported.**

Spanned records are not supported for input

**TTFU016I \* Output file: RECFM=FB, LRECL= *nn,nnn*, BLKSIZE= *nn,nnn*  
or z/OS UNIX file.**

The output data file has been successfully opened.

**TTFU017S Output file RECFM VBS is not supported.**

Spanned records are not supported for output

**TTFU019I Input commands:**

Marks the beginning of command processing

**TTFU020E \* Unable to load file: TTFUCPS**

This file is necessary for processing. Ensure it is in the STEPLIB for the step

**TTFU021E \* Unable to load file: TTFCDFLT**

This file is necessary for processing. Ensure it is in the STEPLIB for the step

**TTFU023E \* Unable to load file: Tffttt**

This is the codepage translation file; the 'fff' is the base 36 id for the from file and the ttt represents the base 36 id for the to file. This file is necessary for processing. Ensure it is in the STEPLIB for the step

**TTFU024E \* Unable to load file: TTFPARSE.**

This is the main parsing subroutine. This file is necessary for processing. Ensure it is in the STEPLIB for the step

**TTFU028S Insufficient storage; increase REGION size.**

or, from the Parsing routine:

**TTFP028S Insufficient storage; increase REGION size.**

An attempt to dynamically obtain storage failed.

**TTFU031I** : *\_\_ content \_\_*

Display utility command record image; blank lines and comments are not displayed

**TTFU034W > Col. 1 must be blank or asterisk.**

Neither commands nor continuations may begin in column 1

**TTFU036W > Command name too long; ignored.**

Command names are 15 characters or less in length

**TTFU038W > No command found.**

Should not see this, due to earlier checks. (See TTFU067W)

**TTFU039W > No operand found.**

After a command was found, the operand string was blank

**TTFU041E Reached unexpected point: xxxxx;**

**contact Trainers Friend: 303-393-8716**

(All on one line.) Possible bug in utility. Contact us.

**TTFU042E > Operand string too long. Ignored.**

More than 32756 bytes of operand string. Rest of process is canceled.

**TTFU050E Errors found, application terminated.**

**TTFU050E 'E' level diagnostics encountered. Syntax checking completed.**

Syntax checking found 'E' level errors (check the log), completed syntax checking, but did not run the utility.

**TTFU051I Run characteristics:**

At this point, the input commands have been gathered and parsed. Now, the utility reports on its understanding of the run request. This could be useful if you find unexpected results.

**TTFU052I End of run characteristics.**

This line delimits the end of the analysis phase.

**TTFU053I Processing Started.**

This line marks the beginning of the run phase.

**TTFU054I Processing Ended.**

This line indicates the run has completed.

**TTFU055S Processing Canceled.**

This message is issued instead of TTFU053I and TTFU054I when the analysis phase detects errors that indicate the run cannot be successful. Read the previous log messages.

**TTFU058I \* All input data records will be processed.**

**TTFU059I \* Skip first *nnn* input data records.**

**TTFU060I \* Stop after *nnn* data records written.**

Based on OPTIONS STARTAFTER and STOPAFTER paramters

**TTFU062E ==> Missing info for translate table ==> Tfffft**

A codepage translation requires an entry in the table that summarizes all the translate tables. Someone may have added a codepage translation table without updating TRANINFO. See the chapter on codepage support in the Installation and Customization Guide.

**TTFU063I \* Character strings will be translated from code  
page *nnnnn* to code page *nnnnn***

or

**TTFU064I \* No code page translations will be done**

Information about translation to be done in this run

**TTFU065I Records read from input file: *nnnnnnnnnnnn***

**TTFU066I Records written to output file: *nnnnnnnnnnnn***

Summary information.

**TTFU067W > Unknown command; ignored.**

The first word on a utility command record was not recognized.

**TTFU070W >Multiple OPTIONS commands present.**

**First encountered command will be used.**

**TTFU071W >Multiple IFILE commands present.**

**First encountered command will be used.**

**TTFU072W >Multiple OFILE commands present.**

**First encountered command will be used.**

Only one of each of these commands is allowed in any run.

**TTFU076W Invalid character x'*xx*' found at location *nn* in record number *m***

**TTFU077W Replaced by substitution character.**

A character was found in the input data that cannot be translated into the code page of the output; the substitution character has replaced the untranslatable character in the output

Check that any cp= parameter you have specified is correct for the input file you are working with

**TTFU078E Run stopped.**

An error was encountered (such as an untranslatable character in message TTFU076W) and the OPTIONS command specified the run should be stopped when this error is found.

**TTFU083S Severe error from parse routine.**

Generated when return code from TTFPARSE is 12.

**TTFU086W Utility run skipped due to options value.**

Generated when RUN=NO coded on OPTIONS command

**TTFU094I Output record larger than file LRECL.**

**TTFU095W Based on OPTIONS TRUNCATE value, a truncated record will be written.**

**TTFU093I Bad record is input record number: *n***

**TTFU094I Output record larger than file LRECL.**

**TTFU096W Based on OPTIONS TRUNCATE value, the record will not be written out.**

**TTFU093I Bad record is input record number: *n***

**TTFU094I Output record larger than file LRECL.**

**TTFU097E Based on OPTIONS TRUNCATE value, the run will be stopped.**

**TTFU093I Bad record is input record number: *n***

There is not enough room in the output record, even after truncating trailing blanks, to hold the generated data; the response depends on the setting of TRUNCATE= on the OPTIONS statement (default is to stop the run).

**TTFU098S Output LRECL too small. Re-run with larger LRECL.**

Increase the LRECL value for TTFOUT and re-run the job.

**TTFU099E Input HFS record larger than buffer. Run stopped.**

Reading from input record, and record size is larger than 128KiB. This is not supported

**TTFU100W Output HFS record larger than prefix size allows. The record will not be written out.**

The output file is a z/OS UNIX file with record prefixes; the size of the current record is larger than the prefix size can hold; for 1 byte prefix, output record size > 256; for a 2 byte prefix, output record size > 65536; for a 4 byte prefix the output record size is larger than the output buffer. The record is skipped and processing continues.

**TTFP201W > Operand name too long.**

An operand name longer than 15 characters was encountered; this can occur if an operand does not have an = sign after it.

**TTFP202E > Operand namespace exhausted.**

More than 24 operand names in a single statement.

**TTFP203W > Unrecognized operand name: *nnnnnn*; ignored.**

Operand name misspelled or unsupported.

**TTFP204E > Operand value too long.**

Value longer than allowed maximum for the operand or total size of all values larger than 1200 bytes.

**TTFP206E > Unsupported CodePage requested: *nnnnn***

Check the list of supported code pages

**TTFP207W > Encountered name with no value.**

An operand was found on a command such as **name=**, or **name=** with a space or no value following the equals sign

**TTFP209E > PATH not found, so FILEDATA, RECLLEN, RECDELIM, RECPREFIX not allowed.**

The named parameters all relate to z/OS UNIX files

**TTFP210W > Invalid RECPREFIX value (valid: 1,2,4); ignored.**

Fix the value in RECPREFIX

**TTFP211E > Invalid RECLLEN value: not positive integer.**

Fix the value in RECLLEN

**TTFP212W > Invalid FILEDATA value (valid: T,B,R)**

Fix the value in FILEDATA

**TTFP213E >Invalid RECDELIM value (valid:CR,LF,CRLF,LFCR,NL).  
Copy will not be run.**

Fix the value in RECDELIM

**TTFP214W > Invalid value for V=. Only Y is valid; ignored.**

Fix the value in V

**TTFP215W > Multiple RECDELIM parameters; first used.**

Include the value you really want

**TTFP216W > Multiple RECLLEN parameters; first used.**

Choose the value you want to use.

**TTFP218W > Multiple RECPREFIX parameters; first used.**

Specify the value you mean to use.

**TTFP219W > RECPREFIX and RECDELIM are mutually exclusive; first used.**

Specify the correct value

**TTFP221W > Multiple PATH parameters; first used.**

Only one PATH parameter per xFILE command.

**TTFP222W > Multiple FILEDATA parameters; first used.**

Specify the value you mean.

**TTFP225E > FILEDATA=R conflicts with RECPREFIX.**

FILEDATA=R implies a [hidden] leading prefix

**TTFP226E > FILEDATA=B requires at least one of RECLLEN, RECPREFIX,  
RECDELIM.**

For binary files, the utility needs some way to determine the record boundaries

**TTFP227E > FILEDATA=T conflicts with RECPREFIX.**

Text files use delimiters to determine record boundaries.

**TTFP228E > FILEDATA=T requires RECDELIM.**

Text files use delimiters to determine record boundaries.

**TTFP229E > PATH requires at least one of RECLEN, RECPREFIX,  
RECDELIM.**

This occurs if none of the above parameters are specified; the utility has no way of knowing how to determine record boundaries.

**TTFP230W > PATH speciifed with no FILEDATA; FILEDATA set to {B|T}**

Utility chooses FILEDATA if it is not specified and PATH is. T is chosen if RECDELIM is also specified, otherwise B is used.

**TTFP231W > Invalid value for FILL character; space used.**

Only N (for nulls), B (for blanks) or x'xx' (hex representation of a single character) are allowed.

**TTFP232W > Multiple FILL parameters; first used.**

Specify the value you mean to use.

**TTFP235W > RUN specified incorrectly. RUN=NO assumed.**

Specify RUN=NO or omit.

**TTFP240W > Multiple STARTAFTER parameters; first used.**

Specify the value you mean.

**TTFP241W >Multiple STOPAFTER parameters; first used.**

Specify the value you mean.

**TTFP242E > Invalid STARTAFTER value: not positive integer.**

Fix the value in STARTAFTER

**TTFP243E > Invalid STOPAFTER value: not positive integer.**

Fix the value in STOPAFTER

**TTFP244W > > Invalid TRUNCATE option; TRUNCATE=P assumed.**

Run should be OK, but change the value to eliminate the message

**TTFU900I Final completion code: *nn***

Looking for 0 or at most 4.

This page intentionally left almost blank.

# Appendix: Code Page Support

This Appendix provides information on modifying or expanding codepage support

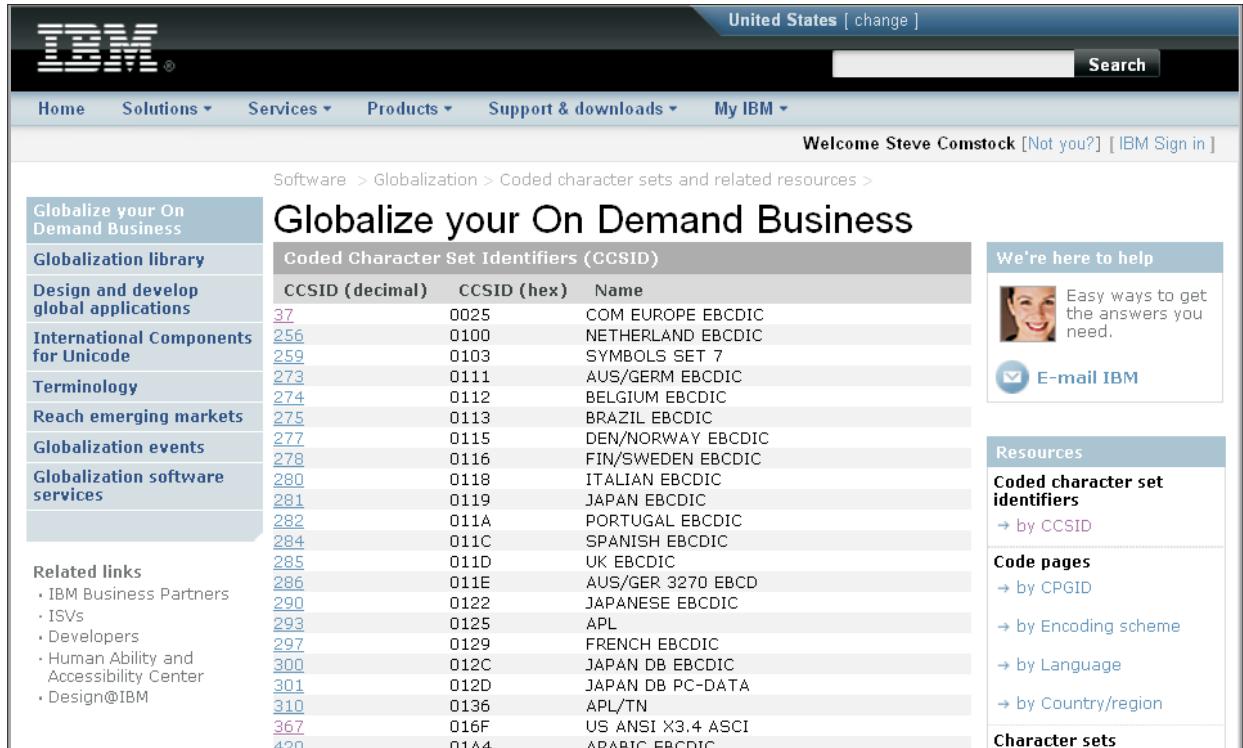
- CCSIDs**
- Codepages in File RePackager**
- Base-36 id's**
- Codepages supported in File RePackager**
- CodePage Tables used in File RePackager**
- Changing the Default Code Page**
- Modifying Existing Code Page Tables**
- Adding Support For a New Code Page**

# CCSIDs

Character set names use terms like "CCSID" - Coded Character Set ID (an integer), "CP" - Code Page, plus various aliases. It can all be quite confusing.

Since File RePackager is developed to run on z/OS, we decided to use this official IBM site when deciding what code pages to support:

[http://www-01.ibm.com/software/globalization/ccsid/ccsid\\_registered.jsp](http://www-01.ibm.com/software/globalization/ccsid/ccsid_registered.jsp):



The screenshot shows the IBM website interface. At the top, there is the IBM logo and a navigation bar with links for Home, Solutions, Services, Products, Support & downloads, and My IBM. A search bar is also present. Below the navigation bar, there is a welcome message for Steve Comstock and a breadcrumb trail: Software > Globalization > Coded character sets and related resources >. The main content area is titled 'Globalize your On Demand Business' and features a table of 'Coded Character Set Identifiers (CCSID)'. The table has three columns: CCSID (decimal), CCSID (hex), and Name. The CCSID (decimal) column contains blue links. To the left of the table is a sidebar with various links under 'Globalize your On Demand Business'. To the right, there are sections for 'We're here to help' and 'Resources'.

CCSID (decimal)	CCSID (hex)	Name
<a href="#">37</a>	0025	COM EUROPE EBCDIC
<a href="#">256</a>	0100	NETHERLAND EBCDIC
<a href="#">259</a>	0103	SYMBOLS SET 7
<a href="#">273</a>	0111	AUS/GERM EBCDIC
<a href="#">274</a>	0112	BELGIUM EBCDIC
<a href="#">275</a>	0113	BRAZIL EBCDIC
<a href="#">277</a>	0115	DEN/NORWAY EBCDIC
<a href="#">278</a>	0116	FIN/SWEDEN EBCDIC
<a href="#">280</a>	0118	ITALIAN EBCDIC
<a href="#">281</a>	0119	JAPAN EBCDIC
<a href="#">282</a>	011A	PORTUGAL EBCDIC
<a href="#">284</a>	011C	SPANISH EBCDIC
<a href="#">285</a>	011D	UK EBCDIC
<a href="#">286</a>	011E	AUS/GER 3270 EBCD
<a href="#">290</a>	0122	JAPANESE EBCDIC
<a href="#">293</a>	0125	APL
<a href="#">297</a>	0129	FRENCH EBCDIC
<a href="#">300</a>	012C	JAPAN DB EBCDIC
<a href="#">301</a>	012D	JAPAN DB PC-DATA
<a href="#">310</a>	0136	APL/TN
<a href="#">367</a>	016F	US ANSI X3.4 ASCII
<a href="#">420</a>	01A4	ARABIC EBCDIC

The "CCSID (decimal)" column values are also links that take you to more information about that code page.( Note that CCSIDs are five-digit numbers; here leading zeros are omitted). Now, for example, scroll down and follow the link for code page 819 and you will see a page that includes this information ...

## CCSIDs, 2

Software > Globalization > Coded character sets and related resource character set identifiers >

### Globalize your On Demand Business

**Globalization library**

**Design and develop global applications**

**International Components for Unicode**

**Terminology**

**Reach emerging markets**

**Globalization events**

**Globalization software services**

**Related links**

- IBM Business Partners
- ISVs
- Developers
- Human Ability and Accessibility Center
- Design@IBM

**CCSID Information Document**  
All CCSID (C) Copyright IBM Corporation 1998

**CCSID** 819 (decimal)  
0333 (hex)

**Name** ISO 8859-1 ASCII

**SC** Interoperable

**FMS** FULL

**Registration Date**

**Description** ISO 8859-1: Latin Alphabet Number 1 Latin-1 c regions

**Notes** Related CCSID with euro is 923

**Encoding Scheme** 4100

**Size** 190

**MCCSID** [819](#)

**ACRI**

**Group Information:**

**Group 1** 1I (Interoperable, Latin 1)

**Number of Code Pages** 1

Character Set	Code Page	CCSID	Control Definition
<a href="#">00697</a>	<a href="#">00819</a>	<a href="#">819</a>	SP SUB NL LF CR EOI 4 5 0 2 2 1

With just a little of this being of interest. For right now, the Code Page column at the bottom, "819" is a further link to a page with additional information ...

# CCSIDs, 3

Software > Globalization > Coded character sets and related resources > Code pages by CPGID >

## Globalize your On Demand Business

**SBCS Code Page Information Document**  
All code pages (C) Copyright IBM Corporation 1987

<b>CPGID</b>	<b>00819</b>
<b>Name</b>	ISO/ANSI Multilingual
<b>Encoding Scheme</b>	4100
<b>Code</b>	ISO 8-Bit
<b>Registration Date</b>	3/1/1987
<b>Last Revision Date</b>	
<b>Languages</b>	Afrikaans, Basque, Catalan, Danish, Dutch, English, Faeroese, Finnish, Flemish, French, German, Icelandic, Irish, Italian, Latin, Latin1, Norwegian, Portuguese, Spanish, Swedish
<b>Countries/Regions</b>	Australia, Austria, Belgium, Brazil, Canada, Denmark, Finland, France, Germany, Iceland, Ireland, Italy, Latin America, Multinational, Netherlands, New Zealand, Norway, Portugal, South Africa, Spain, Sweden, Switzerland, United Kingdom, United States
<b>Maximal Character Set</b>	<a href="#">00697</a>
<b>Other Character Set</b>	
<b>Notes</b>	1. This code page meets the requirements of the external standards: - ANSI X3.134.2 - ISO 8859 Part 1 (February 1987) - ECMA 94 (June 1986)  2. Euro supported code page is 923 (latin 9) The euro currency symbol is added at \xA4.
<b>Code Page</b>	<a href="#">CP00819.pdf</a> <a href="#">CP00819.txt</a>

**Globalize your On Demand Business**

**Globalization library**

**Design and develop global applications**

**International Components for Unicode**

**Terminology**

**Reach emerging markets**

**Globalization events**

**Globalization software services**

**Related links**

- IBM Business Partners
- ISVs
- Developers
- Human Ability and Accessibility Center
- Design@IBM

Again, we are interested in the link at the bottom of the page; this leads us to a PDF page containing the mapping of characters to codepoints. You can view this here, or download it to your workstation to study.

# CCSIDs, 4

The document for codepage 819 looks like this:

		IBM Corporate Specification														C-H 3-3220-050	
		REGISTRY: Graphic Character Sets and Code Pages														© Copyright IBM Corp. 1987. All rights reserved.	
		Code Page CPGID 00819															
HEX DIGITS HEX → DEC ↓	0-	1-	2-	3-	4-	5-	6-	7-	8-	9-	A-	B-	C-	D-	E-	F-	
0		sp	0	@	P	'	p				esp	°	À	Ð	á	ð	
1		!	1	A	Q	a	q				i	±	Á	Ñ	â	ñ	
2		"	2	B	R	b	r				ê	²	Â	Ò	ã	ó	
3		#	3	C	S	c	s				ë	³	Ã	Ó	ä	ô	
4		\$	4	D	T	d	t				ü	´	Ä	Ö	å	ö	
5		%	5	E	U	e	u				ý	µ	Å	Ø	ä	ö	
6		&	6	F	V	f	v				ÿ	¶	Æ	Å	æ	ö	
7		'	7	G	W	g	w				ÿ	·	Ç	×	ç	÷	
8		(	8	H	X	h	x				ˆ	¸	È	Ø	è	ø	
9		)	9	I	Y	i	y				˘	˙	É	Ù	é	ù	
A		+	:	J	Z	j	z				˚	˚	Ê	Ú	ê	ú	
B		+	;	K	[	k	{				«	»	Ë	Û	ë	û	
C		,	<	L	\	l					¬	¼	Ì	Ü	ì	ü	
D		-	=	M	]	m	}				½	½	Í	Ý	í	ý	
E		.	>	N	^	n	~				¾	¾	Î	Þ	î	þ	
F		/	?	O	_	o					¸	¸	Ï	ß	ï	ÿ	

However, if you want some really thorough, useful information about a codepage, visit David Bond's Tachyon Software site, starting with:

<http://www.tachyonsoft.com/cpindex.htm>

this page has a link to each IBM code page and detailed information about each character on the page.

## Codepages in File RePackager

**This is all just background, however, unless you want to modify any of the codepage tables we supply, or if you want to add your own codepage tables. Before you can do this successfully, however, you need to know about base-36 id's ...**

## Base-36 id's

We needed to build our own translate tables between all supported code pages (exception: translation between the various UTF code pages is done by hardware instructions with no tables involved). Each translate table is stored as a member in the run time library for the utility.

Originally we thought we could name these members and include the from and two CCSIDs in the member name. But the problem is CCSIDs are five digits, so this would require member names longer than the eight character limitation for member names. Even if you allow omitting leading zeros, some CCSIDs have four and five significant digits, and we wanted an approach that could support all CCSIDs we might ever want to include.

We came up with an algorithm to convert each CCSID to a base 36 number; this produces 3 characters from the set 0-9, A-Z. We call the result the "Base-36 id" of a CCSID. This works out well, since member names have a limit of 8 characters and the characters must only be uppercase alpha or numeric digits (special cases: @, #, \$ are allowed), and the first character must be alphabetic (or @, #, or \$).

By generating member names, we can avoid having to have a table of tables: that is, a table that provides a look up member name for each combination of from-CCSID and to-CCSID. For translation tables, the names are;

**T****sss****ttt**

where '**sss**' is the base-36 id of the source file's CCSID, and '**ttt**' is the base-36 id of the output file's CCSID. This works for all CCSIDs up to 46656.

## Base-36 id's, 2

CCSIDs that are higher than 46656 are:

49589	C1B5	UK PC-DISPLAY
49652	C1F4	BELGIUM EBCDIC
50016	C360	ARABIC PC-DISPLAY
53668	D1A4	Arabic EBCDIC - special
53685	D1B5	USA MS-PC-DATA
53748	D1F4	INTL EBCDIC
61696	F100	GLOBAL SB EBCDIC
61697	F101	GLOBAL SB PC-DATA
61698	F102	GLOBAL PC-DISPLAY
61699	F103	GLBL ISO-8 ASCII
61700	F104	GLBL ISO-7 ASCII
61710	F10E	GLOBAL USE ASCII
61711	F10F	GLOBAL USE EBCDIC
61712	F110	GLOBAL USE EBCDIC
62251	F32B	Arabic/Latin EBCDIC for OS/390 OE
65520	FFF0	Unicode, empty plane

**Note:** these can manually be mapped to artificial base-36 id's in our table of supported codepages (id's 000-036 are not mapped to by our algorithm for any ibm code page, for example, so if we support any of the above code pages, simply put in entries with the CCSID and the artificial base-36 id)

## Codepages Supported in File RePackager

<u>CCSID</u>	<u>base-36 id</u>	<u>Name(s) and Notes</u>
00037	011	<b>EBCDIC USA; supports these countries / regions: Canada, Australia, New Zealand, Netherlands, Brazil, Portugal, South Africa, Taiwan, Multinational; supports these languages: Afrikaans, Basque, Catalan, Danish, Dutch, English, Faeroese, Finnish, Flemish, French, German, Icelandic, Irish, Italian, Latin, Latin1, Norwegian, Portuguese, Spanish, Swedish</b>
00367	0A7	<b>US ASCII-7, also US ANSI X3.4 ASCII; supports countries: Australia, New Zealand, United States; supports English.</b>
00819	0MR	<b>ISO-8859-1, also ISO Latin-1 Western European ASCII supports these countries / regions: Australia, Austria, Belgium, Brazil, Canada, Denmark, Finland, France, Germany, Iceland, Ireland, Italy, Latin America, Multinational, Netherlands, New Zealand, Norway, Portugal, South Africa, Spain, Sweden, Switzerland, United Kingdom, United States; supports these languages: Afrikaans, Basque, Catalan, Danish, Dutch, English, Faeroese, Finnish, Flemish, French, German, Icelandic, Irish, Italian, Latin, Latin1, Norwegian, Portuguese, Spanish, Swedish</b>
00923	0PN	<b>ISO-8859-15, also ISO Latin-9; the doc says this is 819 + the Euro; but there are some other differences at code points x'A4', x'A6', x'A8', x'B4, x'B8', x'BC', x'BD', and x'BE'</b>
00924	0PO	<b>Latin 9 EBCDIC; codepage 1047 + the Euro, but there are some other code point differences.</b>

## Codepages Supported in File RePackager, 2

<u>CCSID</u>	<u>base-36 id</u>	<u>Name(s) and Notes</u>
01047	0T3	EBCDIC Latin 1/Open Systems; supports countries / regions: Australia, Austria, Belgium, Brazil, Canada, Denmark, Finland, France, Germany, Iceland, Ireland, Italy, Japan, Latin America, Multinational, Netherlands, New Zealand, Norway, Portugal, South Africa, Spain, Sweden, Switzerland, United Kingdom, United States; supports these languages: Afrikaans, Basque, Catalan, Danish, Dutch, English, Faeroese, Finnish, Flemish, French, German, Icelandic, Irish, Italian, Latin, Latin1, Norwegian, Portuguese, Spanish, Swedish
01140	0VO	00037 + Euro support; this is the default codepage as supplied with the utility.

**Note:** telling the File RePackager to convert between codepages is only viable if all the data in the source records is character; any data stored in packed decimal, binary integer, or internal floating point formats will not be treated correctly.

**Special note:** if numeric data was created on the mainframe as COBOL data with an S in the picture and no usage clause, e.g.: PIC S9(5)V99, without COMP, COMP-1, COMP-2, COMP-3, BINARY, or PACKED-DECIMAL the last digit will be represented as an alphabetic character or brace ( { or } ) instead of a numeric digit. This will cause an error in translation also.

## CodePage Tables used in File RePackager

□ The following members are found in the supplied TTFU.ASM library; if changed, they should be assembled and bound (linked) into TTFU.LOAD:

- ◆ TTFCDFLT - a 10 character CSECT that identifies the default codepage for files
  
- ◆ TTFUCPS - a table listing each supported code page by CCSID along with it's base-36 id and codepage family (EBCDIC, ASCII, UTF-8, UTF-16, or UTF-32)
  
- ◆ Tsssttt - translate tables as described on page 31
  
- ◆ TRANINFO - a table about the translation tables available; each entry contains the base-36 id of a source codepage, the base-36 id of a target code page, and two bytes of flags containing information about that particular translation.

The source for all these tables, is supplied in the TTFU.ASM file shipped with the File RePackager.

## Changing the Default Code Page

You can change the default code page used for the utility by modifying the source member TTFCDFLT:

```
000001 TTFCDFLT CSECT
000002 *last_change 2010.03.20-1
000003 TTFCDFLT AMODE ANY
000004 TTFCDFLT RMODE ANY
000005 * The installation default code page
000006           DC C'011400VOE '
000007           DC c'Ver1 of TTFCDFLT'
000008 *
000009 *
000010           END TTFCDFLT
```

◆ You should probably make a copy of the original first

◆ Line 6 contains 10 characters as follows:

✗ Five character CCSID of default code page

✗ Three character base-36 file id of the code page

✗ One character code page type: this must be E, since the default code page must be some EBCDIC code page

✗ A space (reserved for possible future use)

Assemble and bind this new version into the library where you run the utility from, and your done

## Modifying Existing Code Page Tables

You may wish to change the way a particular code page is supported. In particular, you can change:

- ◆ **Mapping from one code page to another - Tf36t36**

- ✗ Simply change the source in TTFU.ASM then Assemble and bind into TTFU.LOAD

**Note: As a safety feature, make copies of original members; perhaps set up JCL to use a library with your modifications in the STEPLIB concatenation ahead of TTFU.LOAD. This way you can back out any unwanted result by deleting the module in the modified load library.**

## Adding Support For a New Code Page

To add a new code page, here are the recommended steps

- ◆ Go to <http://www.tachyonsoft.com/cpindex.htm> and click on the link for the codepage you are adding
- ◆ Using the mapping you find at this link, construct a model that you can use to build translate tables; we recommend a naming convention of `MDLccssid`; look at the existing MDL files in `TTFU.ASM` for sample
- ◆ Run the `GETB36` utility to find the base 36 id for your new codepage (see the last step in member `FRPIVP`)
- ◆ Build a set of translate tables: from each supported code page to your new code page, and from your new code page to each already supported code page

- ✗ Member names are of the form `Tf36t36` as discussed on page 31; go to ISPF 3.4 and bring up the `TTFU` libraries; next to `TTFU.ASM` enter `E`; this gets you a member list; on the command line type:

```
==> s Tf36t36;copy mdlccsid;c mdlccsid Tf35t36 all
```

- ✗ This will get you in edit of your new translate table, copy the model for the `from-ccsid`, and change the name to be correct
- ✗ Next, for each code point in the table, code the target value for your translation; for code points you consider invalid, use `x'01'`

## Adding Support For a New Code Page, 2

- ◆ **After you have your translate tables built, Assemble and bind each of them into TTFU.LOAD**
- ◆ **Next, you need to update member TRANINFO; use the existing member in TTFU.ASM as a starting point and as a model: follow this style: add an entry for each translate table; comment in the same style as shown also**

**X Assemble and bind TRANINFO into TTFU.LOAD**

- ◆ **Finally, you need to update TTFUCPS to add your new codepage in the list of supported codepages; again. follow the existing code as a model; place your new entry before the end of table flag**

**X Assemble and bind TTFUCPS into TTFU.LOAD**

- ◆ **At this point you should have all the pieces in place. Now, of course, do lots of testing!**

This page intentionally left almost blank.