

Working with Mainframe Reports

Background	1
About the Author	1
Convert Reports from EBCDIC to ASCII	1
Tailoring the EBCDIC to ASCII Conversion.....	2
Mainframe CMS variable length records	2
Convert EBCDIC 132 column records/lines to PC Format	2
Convert EBCDIC Fixed Length Records to PC Format	2
Working with Packed Decimal (Comp-3) Fields, Packed Numeric and Zoned Decimal Fields	3
Doing it by yourself	3
Notes on Negative Packed Numerics and Line Feeds.....	4
Removing/replacing lower ASCII characters	5
More White Papers and Documentation like This.....	5

Background

Mainframe report files (e.g. IBM AS/400, DB2 etc) are usually stored in EBCDIC format, which differs from the ASCII format used by Windows and Unix machines. TextPipe can be used to convert these files to ASCII format.

About the Author

Simon Carter is the CEO of www.DataMystic.com and lead architect of TextPipe Pro, a data conversion, extraction and mining application. Since 1990 he's been helping customers deal with huge quantities of data in a range of data formats.

Convert Reports from EBCDIC to ASCII

This is used for a direct character mapping without regard for packed decimal fields. If you have packed decimal fields, you need to use the Mainframe Copybook filter.

TextPipe provides two conversion filters under *Filters\Convert\Mainframe\EBCDIC to ASCII* and *ASCII to EBCDIC*.

To convert a mainframe file to ASCII format:

1. Open TextPipe
2. On the Filters tab, in the All Available Filters box, double-click on *Filters\Convert\Mainframe\EBCDIC to ASCII*
3. The new filter is added
4. Check the *Output filter* at the base of the filter tree to determine where the output file will be placed, and whether you want to backup the original file or not
5. Drag and drop the file to be converted onto TextPipe's window
6. Click the *Go* button at the base of the window.

7. The file is now converted.

If your file contains packed or zoned decimal fields, please see further below for how to handle them.

Tailoring the EBCDIC to ASCII Conversion

If you need to tailor the EBCDIC to ASCII conversion table, you can modify the mapping used by TextPipe. Follow these steps to generate a new mapping that you can modify:

1. Open TextPipe
2. On the Filters tab, in the All Available Filters box, double-click on *Filters\Maps\Convert EBCDIC to ASCII*
3. You can now tailor the mapping as you wish.
4. Drag and drop the file to be converted onto TextPipe's window
5. Click the *Go* button at the base of the window.
6. The file is now converted.

Mainframe CMS variable length records

If your mainframe file consists of lines of different length, use the *Filters\Convert\End of line characters* filter.

Use the *Filters\Convert\End of line characters* filter to convert mainframe CMS format variable records to records delimited by a line feed. This filter also converts from EBCDIC to ASCII in the same step.

Convert EBCDIC 132 column records/lines to PC Format

Often mainframe reports consists of 132 character lines, which actually have an extra character at the start of each line which determines double printing, bold and other special functions, typically with a '*' or '0' or '1' character.

There are several steps to fixing these files for PC use:

1. Open TextPipe
2. On the Filters tab, in the All Available Filters box, double-click *Filters\Convert\Mainframe\EBCDIC to ASCII*
3. On the Filters tab, in the All Available Filters box, double-click *Filters\Convert\End of line characters* filter, with a fixed length of 133 and a DOS (PC) output line feed
4. On the Filters tab, in the All Available Filters box, double-click *Filters\Remove\Column* to remove the extra character from Column 1 of each line
5. Drag and drop the file to be converted onto TextPipe's window
6. Click the *Go* button at the base of the window.
7. The file is now converted.

We provide a filter for this purpose; please see **mainframe\mainframe binary files to PC text files.fl**

Convert EBCDIC Fixed Length Records to PC Format

1. Open TextPipe
2. On the Filters tab, in the All Available Filters box, double-click *Filters\Convert\Mainframe\EBCDIC to ASCII*

3. On the Filters tab, in the All Available Filters box, double-click *Filters\Convert\End of line characters* filter; specify the Fixed record length input size, and click the DOS (PC) output line feed
4. Drag and drop the file to be converted onto TextPipe's window
5. Click the *Go* button at the base of the window.
6. The file is now converted.

Working with Packed Decimal (Comp-3) Fields, Packed Numeric and Zoned Decimal Fields

Often mainframe reports contain zoned decimal and packed decimal fields which are either unreadable or incorrect when converted directly to ASCII.

Using TextPipe's **Mainframe Copybook Wizard**, you can paste in a copybook with your field names, lengths and types, and TextPipe will build the filter list you need in order to convert it to ASCII. The Wizard is found under *Filters\Wizards\Mainframe Copybook Wizard*.

The Mainframe Copybook Wizard does a great job of handling most mainframe files. There are two specific cases :

Convert only Packed Fields

Occasionally the target application can handle EBCDIC data, but needs packed or zoned decimals to be expanded. If this is the case, first use the Mainframe Copybook Filter to generate the TextPipe filter for your data structure. Then,

1. Remove all restrictions that apply only to Alpha fields (ie leave all packed or zoned filters)
2. Remove the single Convert EBCDIC to ASCII filter at the base of the filter list.
3. Add a new *Filters\Convert\Mainframe\Convert EBCDIC to ASCII* filter as the last filter inside the restriction for EVERY packed/zoned decimal field.

Handling VSAM Multiple-Record Types

TextPipe can also cope with multiple record types.

Let's use an example. Let's say your record structure is always 100 bytes long, and the 3rd character specifies the record type, 'A', 'B' or 'C'.

Using a *Filters\Replace\Search/replace* filter with a perl pattern of

```
.{2}A.{97}
```

..will match 100 characters where the 3rd character is 'A'. The replacement text is always set to

```
$0
```

Inside this Search/Replace filter, we use the Mainframe Copybook filter to generate the expansion for the 'A' type record.

Similarly for record types 'B' and 'C'.

If each record type is a different length, or the record type indicator is in a different position – no matter. Just alter the search pattern accordingly. See below:

```
Restrict to blocks of length 100
|
|--EasyPattern [[ 2 chars, EBCDIC('A'), 97 chars ]] with [$0]
| | [X] Match case
| |--Comment: Use the Mainframe Copybook Wizard to generate filters for the A record
| |--Merge output to file c:\A_Records.ebc
```

```

|   +--Remove all
|
|--EasyPattern [[ 2 chars, EBCDIC('B'), 97 chars ]] with [$0]
|   | [X] Match case
|   |--Comment: Use the Mainframe Copybook Wizard to generate filters for the B record
|   |--Merge output to file c:\B_Records.ebc
|   +--Remove all
|
+--EasyPattern [[ 2 chars, EBCDIC('C'), 97 chars ]] with [$0]
|   | [X] Match case
|   |--Comment: Use the Mainframe Copybook Wizard to generate filters for the C record
|   |--Merge output to file c:\C_Records.ebc
|   +--Remove all

```

This example can be found in **mainframe\split mainframe record types.fl**

Important Note: The record type identifier **MUST** be specified in EBCDIC, as at this stage in the filter it has not yet been converted. Thus to look for an 'A' record you would have to specify the hex character code '\xA0'. You can use a new filter with a *Convert\Mainframe\ASCII to EBCDIC* filter followed by a *Convert\Hex Dump* filter to determine the correct EBCDIC character code to use.

If you have variable record lengths, we use a pattern to capture the variable length record block first - see the example below.

Record Type	Length
A	50 bytes
B	100 bytes
C	200 bytes

```

EasyPattern [[ 2 chars, (EBCDIC('A'), 47 chars) or (EBCDIC('B'), 97 chars) or
(EBCDIC('C'), 197 chars) ]] with [$0]
|   | [X] Match case
|--EasyPattern [[ 2 chars, EBCDIC('A'), 47 chars ]] with [$0]
|   | [X] Match case
|   |--Comment: Use the Mainframe Copybook Wizard to generate filters for the A record
|   |--Merge output to file c:\A_Records.ebc
|   +--Remove all
|
|--EasyPattern [[ 2 chars, EBCDIC('B'), 97 chars ]] with [$0]
|   | [X] Match case
|   |--Comment: Use the Mainframe Copybook Wizard to generate filters for the B record
|   |--Merge output to file c:\B_Records.ebc
|   +--Remove all
|
+--EasyPattern [[ 2 chars, EBCDIC('C'), 197 chars ]] with [$0]
|   | [X] Match case
|   |--Comment: Use the Mainframe Copybook Wizard to generate filters for the C record
|   |--Merge output to file c:\C_Records.ebc
|   +--Remove all

```

This example can be found in **mainframe\split mainframe record types-variable length.fl**

Notes on Negative Packed Numerics and Line Feeds

If your data includes negative packed numerics, you need to be particularly careful NOT to use any filters than interpret columns prior to the data being converted from EBCDIC to ASCII. This is because the EBCDIC negative information looks like a DOS line feed, which throws out any line or column-based filters. Filters to avoid include Insert Columns, Restrict to Columns etc.

Removing/replacing lower ASCII characters

In DB2 table unloads, it is common for VARCHAR fields to be padded with lower ASCII characters (from 0 to 31 decimal). To remove these characters, you can use a Remove Binary filter (suitably restricted), or to replace them, use a perl pattern search/replace like this:

```
[\x00-\x1f]
```

...and replace with a single space. You could also use a map to replace an arbitrary set of characters with a space. This filter should be placed AFTER the Convert EBCDIC->ASCII filter.

More White Papers and Documentation like This

Available from:

www.datamystic.com/docs