

COMP 3400 Mainframe Administration¹

Christian Grothoff

christian@grothoff.org

<http://grothoff.org/christian/>

¹These slides are based in part on materials provided by IBM's Academic Initiative.



Today

- Porting C/C++ Applications to z/OS

TAR

- z/OS Tar only supports V7 format archives
- Use “`--format=v7`” when creating TAR for z/OS on GNU/Linux

FTP

- ASCII — converts to EBCDEC
- BINARY — no conversion
- When transferring tar files, use BINARY
- Using the `zos.kctr.marist.edu` FTP server will store files on the MVS side!

EBCDEC

- Shell scripts and C code must be in EBCDEC, but we got ASCII!
- Need to convert:²

```
iconv -f UTF-8 -t IBM-1047\  
  contrib/ascebc > /tmp/ascebc.sh  
chmod +x /tmp/ascebc.sh  
for n in `find * -type f`  
do  
  /tmp/ascebc.sh $n  
done
```

²Use <https://gnunet.org/svn/libmicrohttpd/contrib/ascebc.sh>!

C Compiler

- The z/OS C compiler is c89, the preprocessor is c89 -E
- Some configure scripts fail to detect it. Use these environment variables (for now):

```
export _CCC_CCMODE=1
export _C89_CCMODE=1
export CC=c89
export CPP="c89 -E"
export CXXCPP="c++ -E -+"
```

z/OS offers POSIX, right?

- POSIX does not specify in which header certain prototypes must be.
- Some POSIX functions are only available if requested:

```
#define _XOPEN_SOURCE_EXTENDED 1
#if OS390
#define _OPEN_THREADS
#define _OPEN_SYS_SOCKET_IPV6
#define _OPEN_MSGQ_EXT
#endif
```

- We may also need: `c89 -Wc, 'LANGlvl(EXTENDED)'`

Oops!

- `sscanf` / `sprintf` now expect EBCDEC
- Network protocols (HTTP, SMTP) still speak ASCII!

⇒ Need to convert

Converting

- `size_t __a2e_s(char * string)`
- `size_t __e2a_s(char * string)`
- Insert everywhere!?
- Cost!?

ASCII in z/OS C Code

- Tell C Compiler to use ASCII for all our strings!
- c89 `-Wc, "ASCII"`

⇒ Now C strings are in ASCII, but `sscanf` still wants EBCDEC!

Getting ASCII for standard C functions

Install libascii in \$HOME/libascii. Then write a script xcc with these contents:

```
c89 -Wc, 'LANGLVL(EXTENDED),XPLINK'  
-Wl, 'XPLINK'  
-D__STRING_CODE_SET__="ISO8859-1"  
-Wc, "ASCII"  
-L$HOME/libascii -lascii $@
```

Now C uses ISO8859-1 for its default codepage rather than IBM-1047 for character constants and string literals and calls the ASCII run-time functions!

Questions

