

**z/OS Version 1 Release 2**

---

# **UNIX System Services**

---



## **LFS Enhancements**

---



- ▲ Enhanced ASCII support
- ▲ Support for new TCP/IP Resolver
- ▲ LFS Soft Shutdown

## Enhanced ASCII Support

---



- ▲ Automatic codeset conversion (Autoconversion)
  - File tagging
- ▲ C/C++ Compiler options
- ▲ Language Environment Run Time Library (RTL) enhancements

## ASCII Support

---

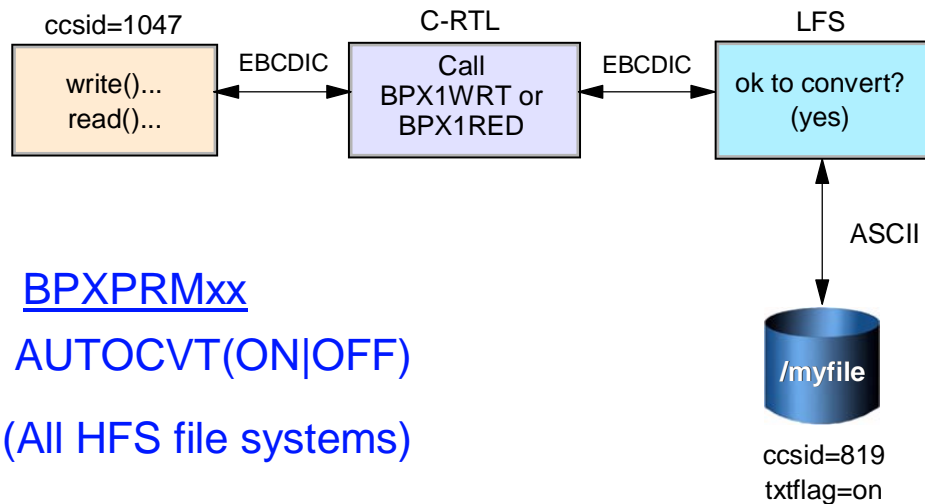


- ▲ Helps with porting of Unix applications to USS
  - Build ASCII-based applications by producing object code with ASCII string literals and character constants and a flag that identifies applications as ASCII or EBCDIC
  - Use Unicode-based wide characters (wchar\_t) in ASCII-based applications
  - Transparently call native ASCII run time library functions from ASCII-based applications
  - Process user-defined ASCII multi-byte code pages with user-supplied code set related methods
  - Create ASCII-based locale objects which allow processing of ASCII data natively at run time

## Automatic Conversion of Files



- ▲ ASCII<->EBCDIC conversion for read() & write()
- ▲ Programs and files can have a CCSID: File tagging



[BPXPRMxx](#)

[AUTOCVT\(ON|OFF\)](#)

(All HFS file systems)

## File Tagging



- ▲ A way to identify the code set of text data in files
  - File tag is metadata for the file
    - `txtflag` (text data) - codeset (16 byte code page)
- ▲ Used during automatic codeset conversion
  - Independent of autoconversion
- ▲ Tagging files
  - `BPXPRMxx ... ROOT - MOUNT` statements
    - `TAG (NOTEXT | TEXT,CCSID)`
  - `chtag` shell command
  - Changes to existing shell commands

## Tag Parameter Examples



**TAG(TEXT,819)** Identifies text file containing ASCII (ISO-8859-1) data.

**TAG(TEXT,1047)** Identifies text file containing EBCDIC ((IBM-1047) data.

**TAG(NOTEXT,65536)** Tags file as containing binary or unknown data.

**TAG(NOTEXT,0)** Is the equivalent of not specifying the TAG parameter.

**TAG(NOTEXT,273)** Tags file with the German code set (IBM-273), but the file is ineligible for automatic conversion.

## Enhanced ASCII Mounts



### BPXPRMxx member

```
MOUNT FILESYSTEM('USER.FILES')  
    MOUNTPoint('/usr/lib/asciifiles')  
    TYPE(HFS)  
    TAG(TEXT,819)
```

### Shell command

```
> /usr/sbin/mount -f user.files -t hfs -c text,819 /usr/lib/asciifiles
```

### REXX syscall

```
/* rexx */  
m.=''  
m.mnte_fsname='USER.FILES'  
m.mnte_path='/usr/lib/asciifiles'  
m.mnte_type='HFS'  
m.mnte_filetag='033380000'x  
address syscall 'mount m.'
```

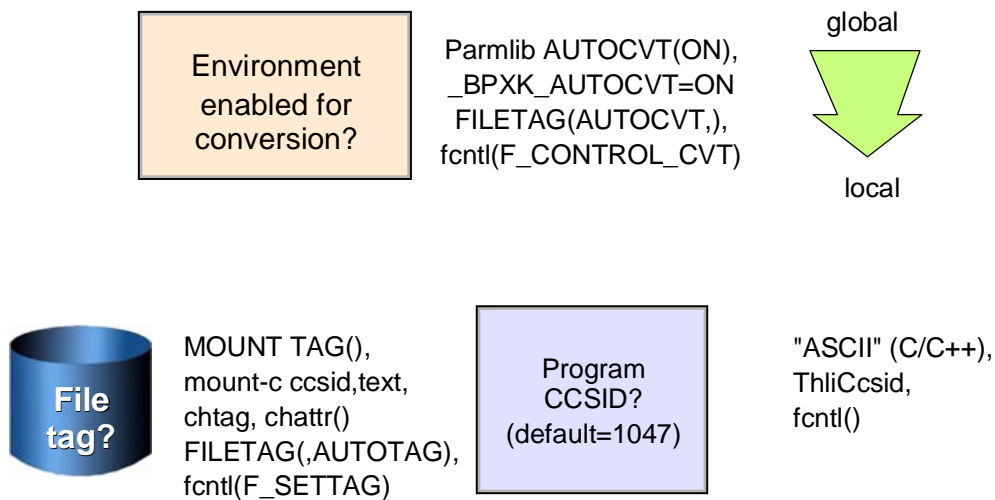
## Checks for Autoconversion



- ▲ Is the environment enabled for conversion
- ▲ Is there a file tag that indicates that the file is a candidate for autoconversion
- ▲ Is the program CCSID different from the CCSID in the file tag

Note: All checks must be true

## User Options for Autoconversion



## Using Autoconversion in Programs

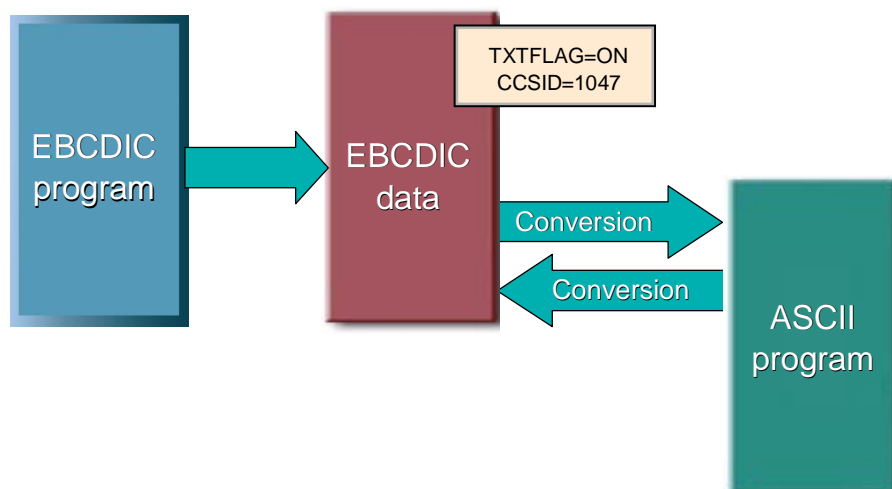


- ▲ For entire programs written in C/C++, use the ASCII compiler to change it to 819 (ISO8859-1 ASCII)
- ▲ For C/C++ threads, use the F\_CONTROL\_CVT subcommand of fcntl()
- ▲ For Assembler programs and threads, use the F\_CONTROL\_CVT subcommand of the BPX1FCT callable service. F\_CONTROL\_CVT sets the CCSID of the program associated with each opened file. (That is, the program CCSID can be different depending on which file is chosen.)
- ▲ For threads, set field Thliccsid using the mapping macro BPXYTHLI

## Programs Accesssing Data



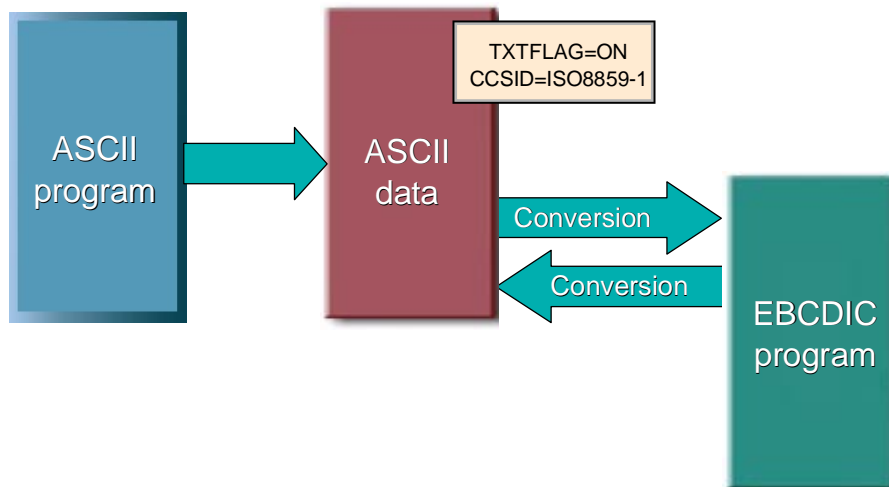
### EBCDIC data accessed by ASCII program



## Programs Accessing Data



### ASCII data accessed by EBCDIC program



## TCP/IP Resolver Support



### ▲ New TCP/IP Resolver Address Space

- Common search order for IP address / hostname resolution
- New Parmlib statement (BPXPRMxx):  
RESOLVER\_PROC(*procname*) or  
RESOLVER\_PROC(NONE) or  
RESOLVER\_PROC(DEFAULT)
- LFS starts *procname* when an INET domain is defined in BPXPRMxx
- TCP/IP supplies the default procedure

## D OMVS,OPTIONS



```
BPXO043I 10.57.12 DISPLAY OMVS 932
OMVS      000F ACTIVE          OMVS=(2A)
CURRENT UNIX CONFIGURATION SETTINGS:
MAXPROCSYS      =          300      MAXPROCUSER      =          10125
MAXFILEPROC     =          65535     MAXFILESIZE     =      NOLIMIT
MAXCPUTIME      = 2147483647         MAXUIDS        =           50
MAXPTYS         =           256
MAXMMAPAREA     =          4096      MAXASSIZE       = 2147483647
MAXTHREADS      =         100000     MAXTHREADTASKS =          32768
MAXCORESIZE     =         4194304    MAXSHAREPAGES  =         32768000
IPCMSGQBYTES    =         262144     IPCMSGQMNUM    =          10000
IPCMSGNIDS      =          20000     IPCSEMNIDS     =          20000
IPCSEMNOPS      =          32767     IPCSEMNSEMS    =           25
IPCSHMPAGES     =          25600     IPCSHMNIDS     =          20000
IPCSHMNSEGS     =           1000     IPCSHMSPAGES   =         2621440
SUPERUSER       = BPXROOT           FORKCOPY        = COPY
STEPLIBLIST     =
USERIDALIASTABLE=
PRIORITYPG VALUES: NONE
PRIORITYGOAL VALUES: NONE
MAXQUEUEDSIGS   =         100000     SHRLIBRGNSIZE  =         67108864
SHRLIBMAXPAGES  =           4096     VERSION        =      Z02RA1
SYSCALL COUNTS  = NO                 TTYGROUP       =      TTY
SYSPLEX         = YES                 BRLM SERVER    =      SC63
LIMMSG          = NONE                 AUTOCVT        = OFF
RESOLVER PROC   = DEFAULT
```

## LFS Soft Shutdown



### ▲ F BPXOINIT,SHUTDOWN=FILESYS

- Unmounts filesystems prior to re-IPL
- Cached buffers synched to disk

### ▲ Non-Sysplex:

- Unmount Force all filesystems

### ▲ Sysplex:

- Unmount automounted filesystems & filesystems mounted on them
- Move Automove(Yes) filesystems to other systems
- Unmount Force remaining filesystems