IBM

# IBM Education Assistant (IEA) for z/OS V2R3

## UNIX - 2038/2042 support

# Agenda

- Trademarks
- Session Objectives
- Overview
- Usage & Invocation
- Interactions & Dependencies
- Migration & Coexistence Considerations
- Installation
- Session Summary
- Appendix

# Trademarks

- See url http://www.ibm.com/legal/copytrade.shtml for a list of trademarks.

- Additional Trademarks:  None

# Session Objectives

- External z/OS UNIX syscall changes supporting 2038.

- General discussion of internal UNIX changes supporting 2042.

# Overview

- Problem Statement / Need Addressed

  1. The signed 4 byte POSIX seconds since epoch (SSE) will overflow in January 2038.

  2. The unsigned 8 byte z/Architecture time-of-day clock provided via STCK will overflow in September 2042.

- Solution

  z/OS UNIX is updating the remaining external system calls to support dates beyond 2038/2042.

  z/OS UNIX is updating internal flows to handle 2042 related overflows.

- Benefit / Value

  Provide early support for future use of z/OS UNIX.

# Usage & Invocation

Assembler Callable Service system call updates for 2038 support:

- BPX1QCT(message control)

- BPX1SCT(semaphore control)

- BPX1SHM(shared message control)

Sub-commands IPC_SET and IPC_STATUS only allow 4 byte POSIX times.

AMODE 64 version BPX4QCT, BPX4SCT, BPX4SHM allow for 8 byte POSIX times.

*New support:*  IPC_SET_EX and IPC_STATUS_EX allow for 8 byte extended POSIX times with AMODE 31.

e.g.

CALL BPX1QCT,(MSG_ID,

          =A(IPC_STATUS_EX),

          =A(BUFFER),          ← output includes 64 bit times (see BPXYMSG)

        RV,RC,RSN)

(continued...)

- BPX1QRC(message receive)

  Flag MSG_INFO only provides 4 byte message sent time.

  *New support:*  New flag MSG_INFO_EX allow 8 byte message sent time, just like BPX4QRC.

- BPX1GTH (get thread data)

  *New support:*  Output mapped by BPXYPGTH includes 8 byte start time, i.e., time the process was dubbed.

- BPX1GPS (get process data)

  *New support:*  Output mapped by BPXYPGPS also includes 8 byte start times

(continued...)

- z/OS UNIX REXX support

  REXX syscalls that support time related file attributes are now 2038 ready.

  e.g.,

  … (set and display a file change time) ...

  *"v_setattr vntoken" st_ctime (4102444800)*

  *"v_getattr vntoken" st.*

  *"gmtime" st.st_ctime tm.*

  *say 'gmtime st_ctime =' st.st_ctime ' = ' ,*

     *right(tm.tm_year,4,0) '-' right(tm.tm_mon,2,0) '-' right(tm.tm_mday,2,0),*

     *right(tm.tm_hour,2,0)':'right(tm.tm_min,2,0) 'GMT'*


  *…*   produces  ...

  gmtime st_ctime = 4102444800  =  2100-01-01 00:00 GMT

(continued...)

- z/OS UNIX 2042 support

Most changes for 2042 are internal.   Since not all z/OS supervisor functions are 2042 ready yet, some z/OS UNIX<->z/OS Supervisor functions have not changed (yet).

Some external changes that might be needed for 2042 have not changed. For example, z/OS UNIX SMF external time values are based on the STCK instruction.  z/OS UNIX has not changed these.  However, it is possible customer programs reading SMF data can interpret "wrapped" times (e.g., 00000000 00000001x) as post-2042 times.  z/OS already does this in some cases.  For example, via IPCS:

>ip listtod 0000000000000001

**09/17/2042 23:53:47.370496 STCK   X'00000000 00000001'**

So the time of day is not 1900, but 2042.  z/OS UNIX does this internally as well in some cases.  This is an area that is still under discussion.

# Interactions & Dependencies

- Software Dependencies

    - None

- Hardware Dependencies

    - None

- Exploiters

    - z/OS C/C++ Run Time Library Support

# Migration & Coexistence Considerations

None

# Installation

None

# Session Summary

- z/OS UNIX is 2038 ready and (almost) 2042 ready.

# Appendix

- *z/OS UNIX System Services Planning (GA32-0884)*

- *z/OS UNIX System Services File System Interface Reference (SA23-2285)*

- *z/OS UNIX System Assembler Callable Services Reference (SA23-2281)*