OpenSolaris for IBM System z Technical Update

David Boyes
Neale Ferguson
http://www.sinenomine.net
Agenda

- Why do this?
- Timeline
- Design Decisions
- Porting Process
- Progress Made
- Planned Future Work
- Q & A
OpenSolaris vs Solaris

- **What’s the Difference?**
  - “You can think of OpenSolaris as Solaris.NEXT.” – Sun Marketing
  - OpenSolaris provides the building block technology for what will become the next release of commercial Solaris (in fact, `uname –a` IDs as Solaris 11)
  - Solaris is the core technology available in OpenSolaris PLUS a bunch of add-ons from other parts of Sun and third-parties.
  - OpenSolaris is not (yet) part of the commercial support regimen from Sun (or IBM) support. Plenty of other 3rd party options, though…
Why?

- New workload for IBM System z
  - With the success of the Linux initiative, “mainframe” is less of a dirty word
  - System z capacity increasing to level some previous argument about CPU-intensive workloads
  - Opens up new avenues for “Solaris shops” to push effective virtualization
Why?

- Demonstration of z/VM being “best of breed”
  - Sun domains para-virtualization strategy not working out to be particularly scalable
  - Recent cost structure changes to z/VM pricing leverage better per-virtual machine ROI
  - “Just one more” comment

“Why not? It’s just another virtual machine. We welcome any workload into the System Z family, we’re not picky.”
Why?

- Continue the Server Consolidation push
  - Makes Solaris workloads accessible for consolidation
  - Targets human workload as well as computational workload for better ROI
What’s in it for Users?

- Integrated consolidation strategy
  - Permits concentration to fewer platforms and management tooling
  - Simplified D/R
  - Reuse of:
    - Skill set
    - Procedures
What’s in it for Users?

- Elimination of “religious” arguments:
  - Anti-Linux
  - Anti-Sun
  - Anti-Open Source
What’s in it for Users?

- New tools for improved productivity
  - Availability of new application suites
  - Availability of desirable technology advances
    - Dtrace
    - System management enhancements
    - Printing system enhancements
z/Architecture Overview
z/Architecture Overview

- Descendent of S/360
- Upwardly compatible architecture
- EBCDIC character set
- Big endian
- 16 GPR, 16 FPR, 16 CR, 16 AR
- 5 tier paging schema
- Multiple address spaces
z/Architecture Overview

- 128 bit PSW
  - Program Counter
  - Interrupt Masks
  - Condition Code
  - Addressing mode
- 64 bit control registers
  - Modify the behavior of the hardware
- Access registers
  - Allow concurrent multiple address spaces
- 16 IEEE/HFP/DFP registers
- Prefix page
  - Different Pages 0 & 1 for each CPU
z/Architecture Overview

- 64 bit general registers
  - Can be operated upon as 64 or 32 bit entities

```c
#include <stdio.h>
int main(int argc, char **argv)
{
  union { long x; int y[2]; } longvar;

  longvar.x = -1;
  printf("%08X %08X %ld\n",longvar.y[0],longvar.y[1],longvar.x);
  __asm__ __volatile__ ("slr %0,%0" : "+d" (longvar.x) : "cc");
  printf("%08X %08X %ld\n",longvar.y[0],longvar.y[1],longvar.x);
  __asm__ __volatile__ ("slgr %0,%0" : "+d" (longvar.x) : "cc");
  printf("%08X %08X %ld\n",longvar.y[0],longvar.y[1],longvar.x);
}

FFFFFFFF FFFFFFFF -1
FFFFFFFF 00000000 -4294967296
00000000 00000000 0
```
### z/Architecture Overview

- **PSW locations**

<table>
<thead>
<tr>
<th>Location</th>
<th>Type</th>
<th>Address</th>
<th>Flags</th>
<th>Tag</th>
<th>Offset</th>
</tr>
</thead>
<tbody>
<tr>
<td>EXT 1004</td>
<td>130</td>
<td>07060001</td>
<td>80000000</td>
<td>00000000</td>
<td>00015F1A</td>
</tr>
<tr>
<td></td>
<td>1B0</td>
<td>04000001</td>
<td>80000000</td>
<td>00000000</td>
<td>00014D32</td>
</tr>
<tr>
<td>SVC 008E</td>
<td>140</td>
<td>0701C001</td>
<td>80000000</td>
<td>00000200</td>
<td>002618A6</td>
</tr>
<tr>
<td></td>
<td>1C0</td>
<td>04000001</td>
<td>80000000</td>
<td>00000000</td>
<td>0001406C</td>
</tr>
<tr>
<td>PRG 0004</td>
<td>150</td>
<td>07004001</td>
<td>80000000</td>
<td>00000000</td>
<td>00087C7A</td>
</tr>
<tr>
<td></td>
<td>1D0</td>
<td>04000001</td>
<td>80000000</td>
<td>00000000</td>
<td>00014AD6</td>
</tr>
<tr>
<td>MCH 0000</td>
<td>160</td>
<td>00000000</td>
<td>00000000</td>
<td>00000000</td>
<td>00015F1A</td>
</tr>
<tr>
<td></td>
<td>1E0</td>
<td>04000001</td>
<td>80000000</td>
<td>00000000</td>
<td>00014DEA</td>
</tr>
<tr>
<td>I/O 0004</td>
<td>170</td>
<td>07060001</td>
<td>80000000</td>
<td>00000000</td>
<td>00015F1A</td>
</tr>
<tr>
<td></td>
<td>1F0</td>
<td>04000001</td>
<td>80000000</td>
<td>00000000</td>
<td>00014C3A</td>
</tr>
</tbody>
</table>
z/Architecture Overview

- 64 bit addressing
  - 24 bit support
  - 31 bit support
  - Up to 3 levels of “Region Tables” to give:
    - 42, 53, 64 bit addressing
    - Use `samxx` instruction to switch addressing modes

- New term:
  - >16MB = “above-the-line”
  - >2GB = “above-the-bar”
z/Architecture Overview

- 32 bit Access Registers
- CCWs still only use 31 bit address fields
  - IDAL used for “above-the-bar”

```
06440008 07000000
```

```
“ABCDEFGH”
```

```
0000060000000000000000
```

```
06440008 07000000
```

```
16EB
```

```
2GB
```

```
0000060000000000000000
```

```
0
```

© 2008 SNA
Address Spaces

- Kernel runs in Primary Space mode
- User programs run in Home Space mode
- Copy to/from user just a MVC(L/E) in Access Register mode with AR set for kernel/user address spaces
- Compare this to some of the other elaborate schemes used
Copy on Write

- Uses hardware facility developed for AIX/370
  - Protection exception
  - Records page address
  - Compares against vm map
  - Gives copy or makes new page available
- Linux for S/390 has AIX to thank!
z/Architecture Overview

- Initially, batch oriented O/S
  - PCP, TOS, DOS & OS/360
  - OS/MFT OS/MVT
  - OS/VS1, OS/VS2
  - MVS, DOS/VS
  - OS/390, VSE/ESA
z/Architecture Overview

- Time-share & Hypervisors
  - CP-40, CP-67, VM/370, VM/SP, VM/ESA, z/VM
  - Virtualize a real machine
    - Padded cell for multiple users
    - “Guest” operating systems
    - Even run VM under VM
  - Hardware optimization
    - “Start Interpretive Execution” – SIE
    - Provide a virtual machine descriptor and run directly on hardware
z/VM

- Hypervisor
  - Virtualizes according to the z/Architecture Principle of Operations
  - Shares & overcommits resources
    - Memory
    - I/O
    - CPUs
- Virtualizes resources
  - VSWITCH
  - VDISK
- Hypervisors services
  - Geometry independent disk I/O
  - Idealized network device
  - Configuration services & querying
The Porting Project
Timeline

- 2006
  - Download OpenSolaris code
  - Spare time review of code
  - Build tools: gcc/binutils
  - Sun donates Sunblade
  - Get kernel build happening
Timeline

- **2007**
  - Present progress at System z conference in Munich
  - Call with IBM execs
  - Meeting with interested parties in Somers
  - Meeting with Sun CTO and developers
  - Joint Sun/IBM announcement
  - Analyst conference call
  - Demo at Gartner Data Center Conference
  - Formal project begins Oct 2007
Timeline

- 2008
  - January delivery of working kernel, disk driver, libraries and userland commands
  - March delivery of network driver
  - April delivery of “fully functioning” system
    - SMF
    - gdb
    - gcc testsuite
    - perl
  - Extensive testing by dedicated IBM resource
  - Release of “build 95” image
  - Project on Opensolaris.org/os – source code repository
  - SOL-390 mailing list
Timeline

- 2009
  - Begin port of openJDK
    - Target Linux on z first
    - Interpreter working
    - LLVM base of JIT under development
  - Release “build 100”
  - Keep code in sync with public repository
  - Code cleanup
Development Team

- Neale Ferguson
  - Kernel and Integration

- Leland Lucius
  - I/O Subsystem and CCW Layer
  - Disk Driver
  - Network Driver

- Max Cohen
  - GCC and C/C++ Libraries
  - Dynamic loader
  - Libraries

- Adam Thornton
  - Device Drivers and Release Mgmt

- David Boyes
  - Documentation and Vendor Pacification

- Mary K. Holicky
  - Project Management
Code Base

- Current drop based on “Build 100” release
  - All the commands
  - Additional services
  - Additional GNU packages

- Using the “mercurial” tool to keep current

- Development now on current release (123) and staying current
Design Decisions...

- SNA Codename “Sirius”
- _LP64 datamodel
  - 32-bit compatibility layer for kernel and some Sun utilities

- Architecture Level Set - - IBM System z9 Required
  - Fullword immediate instructions
  - Compare-swap-and-purge (CSP/CSPG) instruction
  - Long displacement (RY) instructions
  - Long relative displacement instructions
  - Load Page Table Entry instruction (LPTEA) ***
  - Purge DAT instruction (IDTE)
  - Will be using cryptographic instructions
...Design Decisions...

- ABI is identical to Linux for IBM System z

- Assumes presence of z/VM
  - 5.3 base
  - DIAG interfaces:
    - Block I/O
    - Network I/O (VM64466/VM64471)
    - PFAULT
    - I/O discovery (DIAG 210)
    - Memory discovery (DIAG 260)
  - VMDUMP
  - SALIPL
  - Co-operative Memory Management (later)
...Design Decisions

- I/O Layer similar to Linux CCW layer

- Separate address spaces for kernel and user processes
  - Allows for split code and data in separate address spaces to prevent buffer overwrite attacks
  - Provision for putting stack into another address space to prevent buffer overrun attacks

- Full 64-bit (16EB) address space
  - 3 levels of region table
  - Linux is 53 bit with most recent patch levels
  - Large page support is already in openSolaris base but not enabled for System z
Current Build Environment

- Initially done in cross-build environment on SPARC64
  - SPARC is “big endian”
  - “ON Build” tools: part of OpenSolaris
    - Ported a couple of tools for s390x support
- Lots of pre/coreqs – e.g. libxml2, mozilla bits and pieces
- Added uts/s390x and uts/zSeries also s390/s390x in lib etc.
- Need to add $[(I|L)]$(ROOT)/usr/include
- Switched to native build of tools/apps in 3rd drop
  - OpenSolaris build requires dmake so can’t yet self-host*
  - GNU tools with new target of “ibm-s390x-solaris2”
    - GCC 4.2.3 with patches (important!) for kernel
    - GCC 4.3.3 for native package builds
    - Binutils very current (2.18.50 or later)
    - gdb-6.1.7
Major Development Areas

- PROM emulation routines
- Virtual memory support: “HAT layer”
- I/O support
  - Device detection and initialization
  - Adapter layer similar to Linux CCW device interface
  - DIAG 250 disk driver
  - Network driver
- Machine check handling/error management
- External interrupt handling
- Thread switching
- Syscall handling
  - Including 32-bit compatibility layer
Integration with z/VM

- Hypervisor interfaces
  - I/O detection via DIAG 210, DIAG 24
  - Storage detection via DIAG 260
  - Network I/O via DIAG 2A8
  - System call to execute CP commands (privileged)
  - Disk I/O via DIAG 250
  - Signal shutdown handling
  - For ECKD devices – CMS formatted & Reserved disk
  - For FBA devices (inc EDEV SCSI) – raw device
  - DCSS for RAMDISK

- Use of SALIPL – instead of porting grub
- VMDUMP when kernel “panics”
DIAG250 v SSCH

103  ./drivers/s390/block/dasd_3370_erp.c
  2315  ./drivers/s390/block/dasd.c
  2770  ./drivers/s390/block/dasd_3990_erp.c
     60  ./drivers/s390/block/dasd_9336_erp.c
     21  ./drivers/s390/block/dasd_9343_erp.c
   128  ./drivers/s390/block/dasd_cmb.c
 1191  ./drivers/s390/block/dasd_devmap.c
   645  ./drivers/s390/block/dasd_diag.c
 2021  ./drivers/s390/block/dasd_eckd.c
   695  ./drivers/s390/block/dasd_eer.c
   253  ./drivers/s390/block/dasd_erp.c
   606  ./drivers/s390/block/dasd_fba.c
   184  ./drivers/s390/block/dasd_genhd.c
   547  ./drivers/s390/block/dasd_ioctl.c
   322  ./drivers/s390/block/dasd_proc.c
11861 total

  1771 diag250_h1.c
  1305 diag250_ll.c
  3076 total
DIAG2A8 v QDIO/QETH

929 drivers/s390/cio/qdio.c
634 drivers/s390/net/qeth_eddp.c
8895 drivers/s390/net/qeth_main.c
168 drivers/s390/net/qeth_mpc.c
319 drivers/s390/net/qeth_proc.c
1842 drivers/s390/net/qeth_sys.c
12787 total

1485 osa.c
1485 total
Architectural Differences

- On SPARC and Intel – dirty/ref bits are part of the TTE/PTE
  - I/O done via DMA which doesn’t update either of those bits
  - Use of the “kpm” segment (remaps real storage) so a move dirties the transient virtual address but not the remapped target
- System z – dirty/ref bits are associated with real page
  - I/O will update those bits
  - Page may be marked for “flushing” when it’s not needed or not wanted
- Mechanism to detect when a page is “up to date”
Dirty Page Explained

vnode Mapping

kpm Mapping

Real Storage
HAT Layer

- An abstraction layer
  - Common API for doing address translation and page table maintenance
  - Architecture-specific routines to implement many of these APIs
  - About 11K lines of code for System z
  - Concepts:
    - Address space that describes a process’ memory
    - Segments – possibly non-contiguous areas of an address space in use
    - hat table that manages memory mapping for a process
    - htable entries that manage page tables
    - page_t structures that manage pages
I/O

- OpenSolaris defines 15 interrupt levels 1-15
  - The lower the level the less important
  - Levels 11-15 run without any interrupts enabled
  - Clock interrupts on level 10 – still has a 100Hz timer
  - Higher levels can pre-empt lower levels
  - Use of PSW mask, control registers, and ISC to map I/O interrupts to a given class

- I/O discovery
  - STSCH – to detect subchannel
  - DIAG 210 to retrieve device characteristics
  - DIAG 24 to determine console address
  - All before interrupts are enable or I/O subsystem is initialized
I/O

- Created a “CCW” nexus driver to represent the I/O topology
- con3215 driver
  - I/O to z/VM console
  - Nothing fancy!
- DIAG 250
  - Multiple I/O outstanding, interrupts not necessarily in the order of the request
- DIAG 2A8
  - Hides complexity of QDIO
  - Layer 2 based
  - DECNet has been tested!
I/O

- Machine check handler will field device attach/detach events

<table>
<thead>
<tr>
<th>00: CP LINK * 200 F202</th>
</tr>
</thead>
<tbody>
<tr>
<td>00: DASD F202 LINKED R/W</td>
</tr>
<tr>
<td>WARNING: Channel Report:</td>
</tr>
<tr>
<td>Solicited: 0</td>
</tr>
<tr>
<td>Overflow: 0</td>
</tr>
<tr>
<td>Chain: 0</td>
</tr>
<tr>
<td>Source Code: 03</td>
</tr>
<tr>
<td>Ancilliary: 1</td>
</tr>
<tr>
<td>Recovery Code: 04</td>
</tr>
<tr>
<td>Source ID: 0001</td>
</tr>
<tr>
<td>NOTICE: Volume TD1200 discovered at 0f202 with blockize 4096 and offset 634</td>
</tr>
<tr>
<td>WARNING: New device f202 online</td>
</tr>
</tbody>
</table>
Miscellaneous Techno Weenie Items

- Uses TRACG opcode to create a trace table
  - CR12 points to current trace entry
  - Traces interrupts and task switches
  - `pgm_flih` performs wrap function
- Uses IDTE to invalidate region and segment table entries
- Uses SCLP services
  - Write to console before I/O system initialized
  - Registers for SIGNAL SHUTDOWN
- Hercules has implemented a couple of the DIAGs and is aiming to do all that OpenSolaris uses
- Most annoying fix?
  - `char c; c = getopt(...);`
Major Development Areas

- Libraries and loader
  - libc etc. part of OpenSolaris source tree
- gcc
  - New target s390x-ibm-solaris2
  - 4.3.2 for kernel build; 4.3.3 native compiler
  - Added #pragma _init, _fini, and ident support
- binutils 2.17.50 +
  - Added a couple of Sun extensions
  - Updated to 2.19
- gdb 6.1.7
Patches to GNU Tools

- 519 binutils_2.18_s390_20080725.diff
- 2860 gcc_4.2.3_s390_20080725.diff
- 19289 gdb_6.7.1_s390_20080725.diff
- 158 gdb_bfd_only_s390_20080725.diff

- Sign off by fsf
- Patch to config.guess accepted and available
  - Target s390x-ibm-solaris2 is “real”
  - config package will propagate to all other gnu packages
- Patches to gcc have been reviewed
  - Corrected patches will be sent up stream
- binutils under review
- gdb will take longer
Some Statistics

- 52909 source files in OpenSolaris tree
- 2240 files added
  - 1091 makefiles
  - 202 assembler (mostly syscall invocations)
  - 282 C
  - 418 headers
- 192 common files modified
Progress Made So Far

- Completed clean build of kernel, usr/lib, and user commands supplied with OpenSolaris source tree

- Server-oriented device drivers
  - Disk
  - Console
  - Network

- GNU compiler/debugger suite and libraries for C/C++ and other gcc-based languages

- Important open-source utilities (gmake, emacs, perl, python etc)

- Countless open-source servers, libraries and tools (Apache, more secure FTP server, ssh, etc)

- Built Solaris IPS – packages/patches will be made available this way
Build 100 is available for download

- Go to www.sinenomine.net and follow the links
- Snapshot of a working system
- Install via DDR2CMSX
- Full documentation for install and operation
- gcc 4.3.3 available as a separate download
## Downloads

<table>
<thead>
<tr>
<th>Month</th>
<th>Unique visitors</th>
<th>Number of visits</th>
<th>Pages</th>
<th>Hits</th>
<th>Bandwidth</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jan 2008</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Feb 2008</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Mar 2008</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Apr 2008</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>May 2008</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Jun 2008</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Jul 2008</td>
<td>1</td>
<td>1</td>
<td>18</td>
<td>27</td>
<td>234.61 KB</td>
</tr>
<tr>
<td>Aug 2008</td>
<td>8</td>
<td>18</td>
<td>66</td>
<td>80</td>
<td>17.22 GB</td>
</tr>
<tr>
<td>Sep 2008</td>
<td>170</td>
<td>247</td>
<td>495</td>
<td>510</td>
<td>1.45 GB</td>
</tr>
<tr>
<td>Oct 2008</td>
<td>2391</td>
<td>2886</td>
<td>4413</td>
<td>4426</td>
<td>517.72 GB</td>
</tr>
<tr>
<td>Nov 2008</td>
<td>663</td>
<td>899</td>
<td>1559</td>
<td>1559</td>
<td>237.89 GB</td>
</tr>
<tr>
<td>Dec 2008</td>
<td>340</td>
<td>456</td>
<td>919</td>
<td>919</td>
<td>184.25 GB</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td><strong>3573</strong></td>
<td><strong>4507</strong></td>
<td><strong>7470</strong></td>
<td><strong>7521</strong></td>
<td><strong>958.52 GB</strong></td>
</tr>
</tbody>
</table>
## Build 100 Download Stats

<table>
<thead>
<tr>
<th>Day</th>
<th>Number of visits</th>
<th>Pages</th>
<th>Hits</th>
<th>Bandwidth</th>
</tr>
</thead>
<tbody>
<tr>
<td>01 Apr 2009</td>
<td>16</td>
<td>108</td>
<td>108</td>
<td>180.91 GB</td>
</tr>
<tr>
<td>02 Apr 2009</td>
<td>78</td>
<td>84</td>
<td>84</td>
<td>3.90 GB</td>
</tr>
<tr>
<td>03 Apr 2009</td>
<td>22</td>
<td>31</td>
<td>31</td>
<td>1.31 GB</td>
</tr>
<tr>
<td>04 Apr 2009</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>174.72 KB</td>
</tr>
<tr>
<td>05 Apr 2009</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>06 Apr 2009</td>
<td>3</td>
<td>4</td>
<td>4</td>
<td>12.21 KB</td>
</tr>
<tr>
<td>07 Apr 2009</td>
<td>13</td>
<td>25</td>
<td>25</td>
<td>3.96 GB</td>
</tr>
<tr>
<td>08 Apr 2009</td>
<td>12</td>
<td>17</td>
<td>17</td>
<td>4.18 MB</td>
</tr>
<tr>
<td>09 Apr 2009</td>
<td>13</td>
<td>19</td>
<td>19</td>
<td>2.62 GB</td>
</tr>
<tr>
<td>10 Apr 2009</td>
<td>19</td>
<td>27</td>
<td>27</td>
<td>3.99 GB</td>
</tr>
<tr>
<td>11 Apr 2009</td>
<td>6</td>
<td>10</td>
<td>10</td>
<td>2.62 GB</td>
</tr>
<tr>
<td>12 Apr 2009</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>11.61 KB</td>
</tr>
<tr>
<td>13 Apr 2009</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
Who’s downloading it?

<table>
<thead>
<tr>
<th>Domains/Countries</th>
<th>Pages</th>
<th>Hits</th>
<th>Bandwidth</th>
</tr>
</thead>
<tbody>
<tr>
<td>Commercial</td>
<td>com</td>
<td>1160</td>
<td>1166</td>
</tr>
<tr>
<td>Unknown</td>
<td>ip</td>
<td>1081</td>
<td>1088</td>
</tr>
<tr>
<td>Network</td>
<td>net</td>
<td>930</td>
<td>930</td>
</tr>
<tr>
<td>USA Educational</td>
<td>edu</td>
<td>120</td>
<td>120</td>
</tr>
<tr>
<td>Netherlands</td>
<td>nl</td>
<td>116</td>
<td>116</td>
</tr>
<tr>
<td>Germany</td>
<td>de</td>
<td>115</td>
<td>115</td>
</tr>
<tr>
<td>Russian Federation</td>
<td>ru</td>
<td>88</td>
<td>88</td>
</tr>
<tr>
<td>Canada</td>
<td>ca</td>
<td>74</td>
<td>74</td>
</tr>
<tr>
<td>Non-Profit Organizations</td>
<td>org</td>
<td>62</td>
<td>62</td>
</tr>
<tr>
<td>Brazil</td>
<td>br</td>
<td>59</td>
<td>59</td>
</tr>
<tr>
<td>Belgium</td>
<td>be</td>
<td>59</td>
<td>59</td>
</tr>
<tr>
<td>United Kingdom</td>
<td>uk</td>
<td>51</td>
<td>51</td>
</tr>
<tr>
<td>Australia</td>
<td>au</td>
<td>45</td>
<td>45</td>
</tr>
<tr>
<td>France</td>
<td>fr</td>
<td>39</td>
<td>39</td>
</tr>
<tr>
<td>Japan</td>
<td>jp</td>
<td>39</td>
<td>39</td>
</tr>
<tr>
<td>USA Government</td>
<td>gov</td>
<td>36</td>
<td>36</td>
</tr>
<tr>
<td>Italy</td>
<td>it</td>
<td>34</td>
<td>34</td>
</tr>
<tr>
<td>Poland</td>
<td>pl</td>
<td>23</td>
<td>23</td>
</tr>
<tr>
<td>Sweden</td>
<td>se</td>
<td>20</td>
<td>20</td>
</tr>
<tr>
<td>Switzerland</td>
<td>ch</td>
<td>16</td>
<td>16</td>
</tr>
<tr>
<td>Norway</td>
<td>no</td>
<td>16</td>
<td>16</td>
</tr>
</tbody>
</table>
What’s Happening Now

- Official POC
- Several unofficial POCs
  - Federal
  - Latin America
- Staying current
  - Updating each week against public repository
  - Pushing to “Betelgeuse” repository at opensolaris.org
- Dtrace
  - 1st pass at code (userland and kernelland) complete
  - Initial testing underway
Dtrace – A first effort

> lockstat -A w
  4:32pm  up 2 day(s), 16:43, 2 users, load average: 0.31, 0.28, 0.17
User  tty   login@ idle  JCPU  PCPU what
root  console Sat12am 1 10 1 -sh
root  pts/1    4:20pm  2 1:32 18 w

R/W reader hold: 173 events in 5.260 seconds (33 events/sec)

<table>
<thead>
<tr>
<th>Count</th>
<th>indiv</th>
<th>cuml</th>
<th>rcnt</th>
<th>nsec</th>
<th>Lock</th>
<th>Caller</th>
</tr>
</thead>
<tbody>
<tr>
<td>171</td>
<td>99%</td>
<td>99%</td>
<td>0.00</td>
<td>306081</td>
<td>0x3000b58e080</td>
<td>(usermode)</td>
</tr>
<tr>
<td>2</td>
<td>1%</td>
<td>100%</td>
<td>0.00</td>
<td>237500</td>
<td>0x3000b58e000</td>
<td>(usermode)</td>
</tr>
</tbody>
</table>

> dtrace -n 'syscall:::entry { @num[probefunc] = count(); }'
dtrace: description 'syscall:::entry ' matched 226 probes
^C

kill 1
llseek 1
lstat 1
lwp_continue 1

> dtrace -n 'syscall::open*:entry { printf("%s %s",execname,copyinstr(arg0)); }'
dtrace: description 'syscall::open*:entry ' matched 2 probes

<table>
<thead>
<tr>
<th>CPU</th>
<th>ID</th>
<th>FUNCTION:NAME</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>20651</td>
<td>open:entry nsqd /etc/svc/volatile/repository_door</td>
</tr>
<tr>
<td>0</td>
<td>20651</td>
<td>open:entry nsqd /etc/svc/volatile/repository_door</td>
</tr>
</tbody>
</table>
What’s Next

- Merge with “tickless” project
- Additional packages (e.g. Apache)
- JDK – the 800lb gorilla in the room
  - zeroASM JDK now part of OpenJDK
  - LLVM port for JIT underway
- Linux interoperability
  - ABI same on both (ex. 1 issue)
  - Run time loader recognizes objects built for glibc
  - Hello world program works!
Remaining Development Areas

- mdb (and finish testing/fixing Dtrace)
- Port of Solaris linker to s390x – “complete”
  - Is the default linker for gcc 4.3.3
  - Work required for TLS support with dynamic loader
- Additional applications and device drivers
  - Tape
  - Crypto acceleration hardware
- Java
- Linux compatibility layer
- Optimizations – stop wasting storage, use –O3 for kernel build
- Bug fixes as people report them
Recent New GNU & Other Packages

- apr-1.3.3
- apr-util-1.3.4
- binutils-2.19
- bison-2.4
- coreutils-6.12
- cvs-1.11.23
- dejagnu-1.4.4
- diffutils-2.8.4
- expect-5.43
- flex-2.5.35
- gzip-1.3.12
- m4-1.4.12
- perl-5.10
- python-2.4

- pyOpenSSL-0.8
- sed-4.1.5
- setuptools-0.6c9
- subversion-1.5.5
- tcl8.5.6
- texinfo-4.13a
- gmp-4.2.1
- mpfr-2.3.0
- git
- mercurial
A Quick Look…

```bash
# prtconf
System Configuration: IBM Corporation s390x
Memory size: 512 Megabytes
System Peripherals (Software Nodes):

s390x
  scsi vhci, instance #0 (driver not attached)
  ramdisk, instance #0
  pseudo, instance #0
  options, instance #0
  ccw, instance #0
  cns1, instance #0
  osa, instance #0
  osa, instance #1 (driver not attached)
  osa, instance #2 (driver not attached)
  dasd (driver not attached)
  dasd (driver not attached)
  dasd (driver not attached)
  dasd (driver not attached)
  dasd, instance #6
  dasd, instance #7 (driver not attached)
  dasd, instance #8 (driver not attached)
  dasd, instance #9 (driver not attached)
  dasd, instance #10
  dasd (driver not attached)
  diag250, instance #0 (driver not attached)
  cpus, instance #0
```
## A Quick Look…

<table>
<thead>
<tr>
<th># svcs</th>
<th>STIME</th>
<th>FMRI</th>
</tr>
</thead>
<tbody>
<tr>
<td>legacy_run</td>
<td>17:13:24</td>
<td>lrc:/etc/rc2_d/S20sysetup</td>
</tr>
<tr>
<td>legacy_run</td>
<td>17:13:24</td>
<td>lrc:/etc/rc2_d/S4011c2</td>
</tr>
<tr>
<td>legacy_run</td>
<td>17:13:24</td>
<td>lrc:/etc/rc2_d/S42ncakmod</td>
</tr>
<tr>
<td>legacy_run</td>
<td>17:13:24</td>
<td>lrc:/etc/rc2_d/S47pppd</td>
</tr>
<tr>
<td>legacy_run</td>
<td>17:13:24</td>
<td>lrc:/etc/rc2_d/S70uucp</td>
</tr>
<tr>
<td>legacy_run</td>
<td>17:13:24</td>
<td>lrc:/etc/rc2_d/S73cachefs_daemon</td>
</tr>
<tr>
<td>legacy_run</td>
<td>17:13:24</td>
<td>lrc:/etc/rc2_d/S81dodatadadm_udaplt</td>
</tr>
<tr>
<td>legacy_run</td>
<td>17:13:24</td>
<td>lrc:/etc/rc2_d/S89PRESERVE</td>
</tr>
<tr>
<td>legacy_run</td>
<td>17:13:25</td>
<td>lrc:/etc/rc2_d/S94ncalogd</td>
</tr>
<tr>
<td>legacy_run</td>
<td>17:13:25</td>
<td>lrc:/etc/rc2_d/S98deallocate</td>
</tr>
<tr>
<td>legacy_run</td>
<td>17:13:32</td>
<td>lrc:/etc/rc3_d/S16boot_server</td>
</tr>
<tr>
<td>online</td>
<td>17:12:52</td>
<td>svc:/system/svc/restart:default</td>
</tr>
<tr>
<td>online</td>
<td>17:12:53</td>
<td>svc:/network/pfil:default</td>
</tr>
<tr>
<td>online</td>
<td>17:12:53</td>
<td>svc:/network/tnc1:default</td>
</tr>
<tr>
<td>online</td>
<td>17:12:54</td>
<td>svc:/network/datalink-management:default</td>
</tr>
<tr>
<td>online</td>
<td>17:12:55</td>
<td>svc:/network/loopback:default</td>
</tr>
<tr>
<td>online</td>
<td>17:13:02</td>
<td>svc:/network/physical:default</td>
</tr>
<tr>
<td>online</td>
<td>17:13:02</td>
<td>svc:/system/identity:node</td>
</tr>
<tr>
<td>online</td>
<td>17:13:02</td>
<td>svc:/system/metainit:default</td>
</tr>
<tr>
<td>online</td>
<td>17:13:02</td>
<td>svc:/system/filesystem/root:default</td>
</tr>
<tr>
<td>online</td>
<td>17:13:03</td>
<td>svc:/system/scheduler:default</td>
</tr>
<tr>
<td>online</td>
<td>17:13:03</td>
<td>svc:/system/boot-archive:default</td>
</tr>
</tbody>
</table>
...A Quick Look...

```
# /usr/sbin/fmadm config
 MODULE                  VERSION STATUS  DESCRIPTION
 cpumem-retire            1.1     active  CPU/Memory Retire Agent
 disk-transport           1.0     active  Disk Transport Agent
 fabric-xlate             1.0     active  Fabric Ereport Translater
 fmd-self-diagnosis       1.0     active  Fault Manager Self-Diagnosis
 io-retire                2.0     active  I/O Retire Agent
 sysevent-transport       1.0     active  SysEvent Transport Agent
 syslog-msgs              1.0     active  Syslog Messaging Agent
 zfs-diagnosis            1.0     active  ZFS Diagnosis Engine
 zfs-retire               1.0     active  ZFS Retire Agent
```
# ls -l /devices/ccw

```
total 18
drwxr-xr-x  2 root  sys  512 Dec 31 1969 cns1@0x0009
crw-------  1 root  root  309,  0 Feb 15 2008 cns1@0x0009:con3215
drwxr-xr-x  2 root  sys  512 Dec 31 1969 dasd@0x0200
brwxrwxrwx  1 root  root  305,  6 Dec 31 1969 dasd@0x0200:dasd
crw-------  1 root  sys  305,  6 Mar 4 18:40 dasd@0x0200:dasd,raw
drwxr-xr-x  2 root  sys  512 Dec 31 1969 dasd@0x0201
brwxrwxrwx  1 root  root  305,  7 Dec 31 1969 dasd@0x0201:dasd
crw-------  1 root  sys  305,  7 Mar 4 18:40 dasd@0x0201:dasd,raw
drwxr-xr-x  2 root  sys  512 Dec 31 1969 dasd@0x0202
brwxrwxrwx  1 root  root  305,  8 Dec 31 1969 dasd@0x0202:dasd
crw-------  1 root  sys  305,  8 Mar 4 18:40 dasd@0x0202:dasd,raw
drwxr-xr-x  2 root  sys  512 Mar  4 18:40 dasd@0x0203
brw-------- 1 root  sys  305,  9 Mar 4 18:40 dasd@0x0203:dasd
crw-------  1 root  sys  305,  9 Mar 4 18:40 dasd@0x0203:dasd,raw
drwxr-xr-x  2 root  sys  512 Mar  4 18:40 dasd@0x0203:dasd,raw
brw-------- 1 root  sys  305, 10 Mar 4 18:40 dasd@0x0300:dasd
```

crw-------- 1 root  sys  305, 10 Mar 4 18:40 dasd@0x0300:dasd,raw
drwxr-xr-x  2 root  sys  512 Dec 31 1969 osa@0x0bc0
crw-------- 1 root  sys  306,  1 Mar 4 18:40 osa@0x0bc0:osa0
drwxr-xr-x  2 root  sys  512 Dec 31 1969 osa@0x0bc1
crw-------- 1 root  sys  306,  2 Mar 4 18:40 osa@0x0bc1:osa1
drwxr-xr-x  2 root  sys  512 Dec 31 1969 osa@0x0bc2
crw-------- 1 root  sys  306,  3 Mar 4 18:40 osa@0x0bc2:osa2
...A Quick Look

```bash
# ps -e

<table>
<thead>
<tr>
<th>PID</th>
<th>TTY</th>
<th>TIME</th>
<th>CMD</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>?</td>
<td>0:00</td>
<td>sched</td>
</tr>
<tr>
<td>1</td>
<td>?</td>
<td>0:00</td>
<td>init</td>
</tr>
<tr>
<td>2</td>
<td>?</td>
<td>0:00</td>
<td>pageout</td>
</tr>
<tr>
<td>3</td>
<td>?</td>
<td>0:01</td>
<td>fsflush</td>
</tr>
<tr>
<td>100229</td>
<td>?</td>
<td>0:00</td>
<td>utmpd</td>
</tr>
<tr>
<td>100004</td>
<td>?</td>
<td>0:01</td>
<td>svc.star</td>
</tr>
<tr>
<td>100006</td>
<td>?</td>
<td>0:04</td>
<td>svc.conf</td>
</tr>
<tr>
<td>100314</td>
<td>?</td>
<td>0:00</td>
<td>fmd</td>
</tr>
<tr>
<td>100214</td>
<td>?</td>
<td>0:00</td>
<td>sac</td>
</tr>
<tr>
<td>100098</td>
<td>?</td>
<td>0:00</td>
<td>sysevent</td>
</tr>
<tr>
<td>100015</td>
<td>?</td>
<td>0:00</td>
<td>dlmgmd</td>
</tr>
<tr>
<td>100103</td>
<td>?</td>
<td>0:00</td>
<td>devfsadm</td>
</tr>
<tr>
<td>100272</td>
<td>?</td>
<td>0:00</td>
<td>inetd</td>
</tr>
<tr>
<td>100278</td>
<td>?</td>
<td>0:00</td>
<td>automoun</td>
</tr>
<tr>
<td>100112</td>
<td>?</td>
<td>0:00</td>
<td>picld</td>
</tr>
<tr>
<td>100472</td>
<td>pts/1</td>
<td>0:00</td>
<td>ksh</td>
</tr>
<tr>
<td>100099</td>
<td>?</td>
<td>0:00</td>
<td>nscd</td>
</tr>
<tr>
<td>100281</td>
<td>?</td>
<td>0:00</td>
<td>cron</td>
</tr>
<tr>
<td>100300</td>
<td>?</td>
<td>0:00</td>
<td>sshd</td>
</tr>
<tr>
<td>100209</td>
<td>?</td>
<td>0:00</td>
<td>rpcbind</td>
</tr>
<tr>
<td>100319</td>
<td>?</td>
<td>0:00</td>
<td>sendmail</td>
</tr>
</tbody>
</table>
```

Summary

- Putting OpenSolaris on IBM System z opens up a lot of interesting options for exploiting virtualization and existing Solaris knowledge in a really reliable environment.

- The porting process has been long, but is proving to be of great interest to IBM, Sun and others

- Build 100 image available for download

- Hercules 3.07 will be able to run OpenSolaris for z standalone but without network

- Questions?
Contact Info

David Boyes
Neale Ferguson
Sine Nomine Associates
info@sinenomine.net
www.sinenomine.net