Sirius - OpenSolaris on System z

Prepared for:
General Distribution

By:
Sine Nomine Associates
43596 Blacksmith Square
Ashburn, VA 20147

September 25, 2007

Document History

Date	*Revision/ Reissue*	*Nature of Change*
25 Sep, 2008	Version 01	New Release

Table of Contents

1. OpenSolaris on z/Architecture

This document describes the design decisions made when planning and implementing the port of OpenSolaris to IBM's z/Architecture. It also describes and lists the changes made to the tools used to build the OpenSolaris code, as well as to the code itself.

1.1 Why?

Apart from having great “hacker” appeal and a high “cool” factor there are interesting and perhaps compelling arguments for pursuing the goal of porting OpenSolaris to z/Architecture.

1.1.1 Re-enfranchising StorageTek Customers

When Sun acquired StorageTek they also inherited a large number of IBM mainframe customers. Now while it would be nice for Sun to migrate these users from IBM to Sun the practical reality of the situation is that it would be easier for Sun to extend their service offerings to these customers existing platform. The StorageTek arm of the business would be able to build on the existing customer relationship knowing that the other parts of Sun are able to complement this relationship.

1.1.2 Eliminating the Religious Arguments

While it should be a strictly business decision of what platforms to run on and which operating systems to use the truth is often different. People, being people, have their prejudices and in the IT world this can be seen by those who are anti-Linux [pro-Solaris] at all costs or anti-SPARC [pro-IBM]. By providing OpenSolaris on z/Architecture this argument is eliminated for the anti-Linux person and the anti-SPARC person. The focus can now be on determining the “right tool for the right task”.

While it is probable that there may be problems at the middle-management level where personal fiefdoms may be threatened, upper management could realize benefits from a better integration of their “classic” and “distributed” groups.

1.1.3 Reuse of Existing Skills

For an existing Solaris site (or a mixed site that includes Solaris) moving to or expanding to OpenSolaris on z/Architecture enables the existing skill sets and operational procedures to be reused and not reinvented.

1.1.4 Increasing the Application Portfolio

Despite the enormous strides made by Linux in attracting software developers to bring their product to their platform, there are still a large number of applications and tools that are not available and may never be available.

Similarly, OpenSolaris on z/Architecture could open new markets to IBM software and services in a similar way to that which Linux on System z has done.

1.1.5 New tools and toys

The section title is a little flippant if not alliterative but what I mean is tools like DTrace and file systems like ZFS would be made available to z/Architecture and the skills and practices developed for them would be usable in this environment.

2. Design Decisions

The following sections describe various characteristics of the design decisions made for the z/Architecture port of OpenSolaris.

2.1 Code Name

As the PowerPC community has chosen the code name “Polaris” for their port, it seems appropriate that this son of the southern hemisphere adopts the stellar name “Sirius” for the port to z/Architecture.

2.2 Datamodel

OpenSolaris on z/Architecture machines will be 64-bit only and conform to the _LP64 data model. There will be support for 32-bit applications.

There will be separate address spaces for kernel and user. The kernel will operate in "primary space mode", user will be home and secondary. In the future, we may look at separating the stack and code into different address spaces to prevent stack overwrite attacks/errors.

2.3 Endianess

Machines implementing z/Architecture are big-endian.

2.4 ABI

OpenSolaris on z/Architecture will conform to the same ABI as described for Linux on System z (see “Linux for zSeries – ELF Application Binary Interface Supplement: LNUX-1107-00”).

2.5 Architecture Level Set

OpenSolaris on z/Architecture requires the following:

- Fullword immediate instructions
- Compare-swap-and-purge (CSP/CSPG) instruction
- Long displacement (RY) instructions
- Long relative displacement instructions
- Load Page Table Entry instruction (LPTE)
- Purge DAT instruction
- Cryptographic instructions

Currently, this means a System z z9-BC or z9-EC machine is required.

2.6 Addressability

OpenSolaris on z/Architecture will support full 64-bit addressing using 3 region tables, a segment table and a page table for each address space.

2.7 Hypervisor

OpenSolaris on z/Architecture will assume the presence of the z/VM hypervisor. It will use many of the features of the hypervisor to perform tasks such as:

- I/O to disk devices using the DIAG x'250' facility. This will remove the need for the device driver to support extensive recovery, retry, and logging capabilities, as these will be performed by z/VM.
- Retrieving configuration information such as I/O devices present and memory size and configuration.
- Co-operative memory management to allow the hypervisor to communicate with the kernel so as to better manage system memory.
- Page fault handshaking that will enable the OpenSolaris kernel to dispatch another process when a memory fault is encountered and the page in question has been paged out by z/VM.

Relying on z/VM's APIs shields the Solaris implementation from hairy issues that would delay or complicate implementation as well as provide opportunities for better performance.. Synthetic I/O instead of intricacies of mainframe devices, a standardized message passing protocol for interacting with TCP/IP stack, and error handling for hardware failures come from this choice. It's called paravirtualization now (in Xen land), but has been present in various ways in VM for quite a while.

2.8 DDI Implementation

The I/O subsystem of System z is unlike that of the SPARC, Intel, or AMD platforms. It is probable that a common I/O layer will need to be written that will allow the subchannels used under z/Architecture to be mapped to structures in OpenSolaris that will meet the requirements of the DDI layer.

A similar type of mechanism was put in place for the port of Linux to z/Architecture and so the concepts used there should be transferable to OpenSolaris (although the code will not).

2.9 Conventions

Each prefix page will contain an address to the current CPU structure. The current thread for that CPU is anchored off the CPU structure. There are no special registers set to hold or locate a kernel data structure.

Within the code and in the naming of files or directories ‘s390’ will refer to 32-bit code and ‘s390x’ will refer to 64-bit code.

2.10 Toolset

The gnu compiler, assembler and linker will be used to build the various components of OpenSolaris. The levels of the tools are as follows:

- GCC 4.3.1
- Binutils 2.17.50

Code destined to run on System z will be cross-compiled (cross-assembled, and cross-linked) on a Sparc 64-bit platform. This platform provides the necessary “big-endian” environment to match the System z target.

Tools such as ctfconvert etc. will be built to run on the Sparc platform.

3. An Introduction to z/Architecture

The z/Architecture evolved from the S/360 (System/360) of the 1960s. IBM, and Thomas Watson, Jr., in particular, risked the family jewels in undertaking the development of the S/360. It was the largest private venture in American history, with $5 billion spent on five new plants and 60,000 additional employees. S/360 was first to employ instruction microprogramming to facilitate derivative designs and create the concept of a family architecture. The family originally consisted of six computers that could each use the same software and peripherals.

Since that time there have been radical changes and enhancements, but a programmer from that era would recognize many of the facilities of S/390. S/360, S/370, and ESA/390, and z/Architecture are upwardly compatible. S/360 was originally designed to allow programs written for earlier IBM hardware to be migrated to the "new" S/360. The ability to move between different models of machines with different performance and price points but conform to this architecture is what has differentiated and separated the S/360 from the rest of the computing world. This upward compatibility is what required that operating systems running on the S/360 to use the IBM EBCDIC character set rather than the standard ASCII system. The architecture itself does not dictate the use of any specific character set, which is why porting Linux was not problematic in this area.

3.1 Basic Operation

z/Architectures uses 64-bits to address 16EB of memory. z/Architecture defines up to a five-tier paging scheme ("regions 0 1 & 2", "segments" and "pages"). Not all five tiers are required so a system may support 31-bits, 42-bits, 53-bits or 64-bits of addressability.

In addition, z/Architecture allows for multiple address spaces and multiple translation lookaside buffers (TLB), access lookaside buffers (ALB) for mapping each separate address space to the physical memory. Such buffers speed up the address translation process by avoiding decoding an address using the (up to) 5 levels of translation tables.

The latest instantiation of z/Architecture as a processor complex is known as System z.

3.2 SMP Support

The z/Architecture is implemented on processors that range from a card that slips into your laptop to a 54-way SMP configuration not much larger than a refrigerator that sits in a corner of the machine room.

In addition, emulators like Hercules and Flex will allow your PC to run any z/Architecture operating system and application.

3.3 Processor Partitioning

Processor partitioning divides a single machine into multiple virtual systems or *images* that each appear to the operating system running in it as a complete and isolated processor. Partitioning allows you to share all processing resources selectively. The number of partitions you can create depends on the manufacturer and the machine type.

Partitioning can also be achieved using the hypervisor z/VM, which I'll discuss in greater detail in the following section, provides a processor with virtual machines where the limit is measured in the range of hundreds to tens of thousands.

3.4 I/O Subsystem

One of the distinguishing features of z/Architecture is its channel subsystem. The architecture defines a unified means of accessing its I/O subsystem. It does this by defining a channel subsystem that is, in effect, a collection of sophisticated independent outboard processing systems that take complete responsibility for performing I/O operations from the CPU. An operating system only has to issue a single instruction to get an I/O operation initiated. The channel subsystem and the I/O devices will perform all the support actions, such as memory access, path selection, and connection, handling conditions such as RPS miss, caching, and error recovery.

I/O commands are directed to devices using Channel Command Words (CCW). These words can be chained together to perform complex I/O operations. It is up to the subchannel and the underlying device to process these commands and return a status to the central processor.

3.5 Processor Speed

Computers are often rated for speed in terms of MIPS, sometimes (correctly) referred to as Meaningless Indicators of Processor Speed. This is especially true of z/Architecture. Any true estimate of MIPS must include the work performed by the channel subsystem. Each component of the subsystem may have considerable processing power that is equivalent to a standalone server.

Similarly, the instruction mix is also an important factor in processor speed. For example, decimal instructions are generally slower than integer. Also there are many complex instructions implemented in millicode (code between microcode and macrocode) like the cryptographic or compression instructions. These take many cycles to complete.

These factors need to be kept in mind when you see comparisons of CPU performance.

3.6 Virtual Machines

Virtual machines have found renewed interest in things like VMWare and Java Virtual Machines. With z/VM, a virtual machine can run anything that could be run on the bare iron, including a copy of z/VM itself (and a copy running in that copy and so on). Virtual machines provide a "padded-cell environment" that isolates one user from another while also allowing them access to both the real resources of the machine and the virtual resources of the VM operating system. You can, for example, define multiple virtual CPUs when more or fewer real ones exist, or virtual disks that may or may not correspond to real hardware.

So, why virtual machines? R.P. Goldberg in the Proceedings of ACM SIGARCH-SIGOPS Workshop on Virtual Computer Systems, March 1973, pp. ii-iii describes the rationale:

> *The development of interest in virtual computer systems can be traced to a number of causes. First, there has been a gradual understanding by the technical community of certain limitations inherent in conventional time-shared multi-programming operating systems. While these systems have proved valuable and quite flexible for most ordinary programming activities, they have been totally inadequate for system programming tasks. Virtual machine systems have been developed to extend the benefits of modern operating system environments to system programmers. This has greatly expedited operating system debugging and has also simplified the transporting of system software. Because of the complexity of evolving systems, this is destined to be an even more significant benefit in the future.*
>
> *As a second point, a number of independent researchers have begun to propose architectures that are designed to directly support virtual machines, i.e. virtualizable architectures. These architectures trace their origins to an accumulated body of experience with earlier virtual machines, plus a set of principles taken from other areas of operating system analysis. They also depend upon a number of technical developments, such as the availability of low-cost associative memories and very large control stores, which now make proposals of innovative architectures feasible.*
>
> *A third reason for the widespread current interest in virtual machines stems from its proposed use in attacking some important new problems and applications such as software reliability and system privacy/security. A final point is that IBM has recently announced the availability of VM/370 as a fully*

supported software product on System/370. With this action, IBM has officially endorsed the virtual machine concept and transformed what had been regarded as an academic curiosity into a major commercial product.

z/VM is a hypervisor, that is, it provides an interface definition to the entities running on it that is the same as the interface definition provided by the real hardware. What this means is that the logical entities we call virtual machines are idealized simulations of a computer. The Control Program (CP) component of z/VM operates the real machine hardware and multiplexes the physical resources of the computing system to the virtual machines.

The z/architecture allows VM to do this because it separates its instruction set into privileged (aka Supervisor State) and non-privileged (aka Problem State) groups. In the Supervisor State, all instructions are valid. In the Problem State, only those instructions are valid that provide meaningful information to the problem program and that cannot affect system integrity; such instructions are called unprivileged instructions. The instructions that are never valid in the Problem State are called privileged instructions. When a CPU in the Problem State attempts to execute a privileged instruction, a privileged-operation exception is recognized. A CPU executes another group of instructions, called semi-privileged instructions, in the Problem State only if specific authority tests are met; otherwise, a privileged-operation exception or a special-operation exception is recognized.

An operating system uses privileged operations to schedule resources between competing applications that are running under it. CP will dispatch a virtual machine running its operating system in non-privileged mode and then trap any privileged operations performed by the virtual machine. When it traps these operations it can

- Determine whether it is a valid thing for the virtual machine to have done
- Determine whether the resource the virtual machine is trying to use is accessible to that virtual machine
- Map any I/O operations to a virtual device or a real or emulated device
- Allow the virtual machine to continue processing from the point of the trap

Similarly, when interrupts occur on the real machine, CP will determine whether the interrupt needs to be reflected to a particular virtual machine such as when an I/O operation that had been initiated by a Linux virtual machine has just completed. Much of the workload for intercepting and simulating instructions and interrupts for a virtual machine has been lifted from CP by the inclusion of hardware assist functions built in to the

processor complexes. These hardware assists provide significant performance boosts for the virtual machine.

VM has always been the bastard child of IBM. It is extremely efficient, which means that you do not need as much hardware to run it. This does not please those who sell hardware. Prior to the success it has had with Linux every so often IBM attempted to kill VM it off, but it has proven resilient:

> *"Throughout 1967 and very early 1968, IBM's Systems Development Division, the guys who brought you TSS/360 and OS/360, continued its effort to have CP-67 killed, sometimes with the help of some IBM Research staff. Substantial amounts of Norm Rasmussen's, John Harmon's and my time was spent participating in technical audits which attempted to prove we were leading IBM's customers down the wrong path and that for their (the customers'!) good, all work on CP-67 should be stopped and IBM's support of existing installations withdrawn."* [R. U. Bayles quoted in Varian 97].

VM has a lot to offer guest operating systems. Think of it as a highly-intelligent BIOS that relieves them of distractions such as dynamic sparing and hardware recovery, as well as supporting the concurrent operation of hundreds or thousands of virtual machines

z/VM is a great platform for running guest operating systems such as OpenSolaris:

1. Resources can be shared among multiple images running on the same VM system. These resources include: CPU cycles, memory, storage devices, and network adapters.
2. Server hardware consolidation. Running tens or hundreds of systems on a single System z server offers customers savings in space and personnel required to manage real hardware.
3. Virtualization: The virtual machine environment is flexible and adaptable. New guests can be added to a z/VM system quickly and easily without requiring dedicated resources. This is useful for replicating servers in addition to giving users a flexible test environment.
4. Running guests on z/VM means these guests can transparently take advantage of VM's support for System z hardware and RAS features.
5. z/VM provides high-performance communication among virtual machines running a variety of operating systems on the same processor. These include virtual switches (layer 3 and layer 2) for internal and external communications and hipersockets for communicating between hardware partitions (LPARs).
6. Data-in-memory performance boosts are offered by VM's exploitation of the z/Architecture (for example, virtual disks).

7. Debugging. z/VM offers a functionally rich debug environment that is particularly valuable for diagnosing problems in the kernel and device drivers.
8. Control and automation: VM's long-standing support for scheduling, automation, performance monitoring and reporting, and virtual machine management is available for guest virtual machines, as well. An entire System Management API is available to control the creation and operation of virtual machines.
9. Horizontal growth: An effective way to grow your workload capacity is to add more guests to a z/VM system.

4. System z Architectural Considerations

z/Architecture is big-endian that has:

- Sixteen 64-bit general purpose registers;
- Sixteen floating-point registers that may operate in IEEE or IBM modes.
- Sixteen 64-bit control registers that control machine operation such as for address spaces, interrupt masks, etc.
- Sixteen 32-bit access registers that control the concurrent use of multiple address spaces

4.1 Program Status Word

The heart and soul of z/Architecture is the Program Status Word (PSW). The PSW is the most important register on the machine; it is 128 bits long and serves multiple roles. The PSW has several advantages over a normal program counter in that you can change multiple settings (such as address translation and program counter) in a single instruction. The following table breaks down the PSW into its components:

Bits	Value
0	Reserved (must be 0 otherwise specification exception occurs.
1	Program Event Recording 1 PER enable. PER is used to facilitate debugging e.g. single stepping.
2-4	Reserved (must be 0).
5	Dynamic address translation 1=DAT on.
6	Input/Output interrupt Mask
7	External interrupt Mask used primarily for inter-processor signaling and clock interrupts.
8-11	PSW Key used for complex memory protection mechanism not used under Linux
12	Architecture selection: 1 on S/390; 0 on z/Architecture
13	Machine Check Mask 1=enable machine check interrupts
14	Wait State set this to 1 to stop the processor except for interrupts and give time to other LPARS used in CPU idle in the kernel to increase overall usage of processor resources.
15	Problem state (if set to 1 certain instructions are disabled) all Linux user programs run with this bit 1 (useful info for debugging under VM).
16-17	Address Space Control 00 Primary Space Mode when DAT on. Control Register 1 (CR1) is affiliated with this mode and points to the primary segment table origin etc. 01 Access register mode this mode is used in functions to copy data between kernel and userspace. 10 Secondary space mode and is affiliated with CR7. 11 Home Space Mode. It is affiliated with CR13.
18-19	Condition codes (CC)
20	Fixed point overflow mask if 1=FPU exceptions for this event occur (normally 0)
21	Decimal overflow mask if 1=FPU exceptions for this event occur (normally0)

Bits	Value
22	Exponent underflow mask if 1=FPU exceptions for this event occur (normally 0)
23	Significance Mask if 1=FPU exceptions for this event occur (normally 0)
24-30	Reserved Must be 0.
31-32	Extended Addressing Mode; Basic Addressing mode. Used to set addressing mode 00 – 24 bit 01 – 31 bit 11 – 64 bit
33-63	Reserved. Must be 0.
64-127	Address: In 24-bit mode bits 64-103=0; bits 104-127=Address In 31-bit mode bits 64-96=0; bits 97-127=Address

Table 1: PSW Description and Contents

4.2 Interrupts

Interrupts are implemented by storing the current PSW and loading a new PSW. This is similar in concept to a vector interrupt table. There are five main types of interrupts defined for z/Architecture. Each type has several different classes that qualify the type of interrupt and are stored in a specified low-core area upon interrupt. The interrupt types are:

1. External -- External events such as timers or signals from other CPUs. The event type is stored as the qualifier.
2. Supervisor Call – Program-controlled signal to operating system. A code 0x00-0xff is stored as the qualifier.
3. Program Check -- Invalid instructions, page translation exceptions, access violations. The type of program fault is stored as the qualifier.
4. Machine Check -- Hardware component warning or failure notification. The type of failure is stored as the qualifier.
5. I/O -- Input/output device status changes. The identifier of the device causing the interrupt is stored as the qualifier.

Upon the occurrence of an interrupt the current PSW (i.e. context) is stored at an architecture defined location for that interrupt type and a new PSW is loaded that will cause execution to resume at the specific interrupt handler.

On VM you can display these PSWs for a virtual machine or you can use the hardware management console if running on the bare iron or within an LPAR:

4.3 Prefix Page

This per CPU memory area is too intimately tied to the processor not to mention. It exists between the real addresses 0-8192 (the first two pages) on the processor and is exchanged with a page in absolute storage by the set prefix instruction. There is an instance of this page for each real or virtual processor attached to the virtual machine. The prefix register holds the address of the prefix page. Bytes 0-512 (200 hex) are used by the processor itself for holding such information as exception indications and entry points for exceptions. The prefix page contains important fields like the old and new PSWs.

4.4 Addressing

First, some quick definitions from the z/Architecture Principles of Operations [ESAPOP] document by IBM:

For purposes of addressing main storage, three basic types of addresses are recognized: absolute, real, and virtual. The addresses are distinguished on the basis of the transformations that are applied to the address during a storage access. Address translation converts virtual to real, and prefixing converts real to absolute. In addition to the three basic address types, additional types are defined which are treated as one or another of the three basic types, depending on the instruction and the current mode.

- An absolute address is the address assigned to a main-storage location. An absolute address is used for a storage access without any transformations performed on it.
- A real address identifies a location in real storage. When a real address is used for an access to main storage, it is converted, by means of prefixing, to an absolute address.
- A virtual address identifies a location in virtual storage. When a virtual address is used for an access to main storage, it is translated by means of dynamic address translation to a real address, which is then further converted by prefixing to an absolute address.
- A primary virtual address is a virtual address that is to be translated by means of the primary segment-table designation. Logical addresses are treated as primary virtual addresses when in the primary-space mode. Instruction addresses are treated as primary virtual addresses when in the primary-space mode, secondary-space mode, or access-register mode.
- A secondary virtual address is a virtual address that is to be translated by means of the secondary segment-table designation. Logical

addresses are treated as secondary virtual addresses when in the secondary-space mode.

- An AR (access register) specified virtual address is a virtual address that is to be translated by means of an access-register-specified segment-table designation. Logical addresses are treated as AR-specified addresses when in the access-register mode. Access registers allow you to use 16 address spaces concurrently.
- A home virtual address is a virtual address that is to be translated by means of the home segment-table designation. Logical addresses and instruction addresses are treated as home virtual addresses when in the home-space mode.

4.5 Dynamic Address Translation

The process of translating a virtual address to a real address consists of a lookup using two tables: a segment table and a page table. These tables reside in real or absolute storage.

Dynamic address translation as done in z/Architecture may use up to 5 levels of translation tables or as few as two. A 64-bit virtual address may be broken down into an 11-bit region 0 index (R0), an 11-bit region 1 index (R1), an 11-bit region 2 index (R2), an 11-bit segment index (SX), a 12-bit page index (PX) and a 9-bit byte index (BX).

Depending on the addressing mode a control register points to a table origin (Region 0 for 64-bit addressability; R1 for 53-bits; R2 for 42-bits; and, SX for 31-bits).

Each entry within the R0-3 & SX tables represents a portion of storage and indicates whether there is a valid subordinate table available to map real storage.

There is one more step that z/Architecture uses: transforming the real address into an absolute address by adding the contents of the prefix register. This mechanism allows the implementation of SMP in the architecture by providing each processor with its own prefix page (aka “page 0”). The hardware uses the prefix page as a place to store important per-processor information.

There is also a hardware shortcut available called the Translate Lookaside Buffer (TLB). A lookup of this table uses the indexes to index a table of recently accessed page frames. This search happens in parallel with the normal translation process. If an address is found in the TLB then the normal process is terminated and the page frame address found in the TLB is used to form the real address.

4.6 I/O Layer

In contrast to most other hardware architectures, z/Architecture has defined a unified I/O access method. This relieves the device drivers as they don't have to deal with different bus types, polling versus interrupt processing, shared versus non-shared interrupt processing, DMA versus port I/O (PIO), and other hardware features as much. However, this implies that either every single device driver needs to implement the hardware I/O attachment functionality itself, or that the operating system provides for a unified method to access the hardware, thus providing all the functionality that every single device driver would have to provide itself. In order to build common device support for I/O interfaces, a functional layer was introduced that provides generic I/O access methods to the hardware.

Though the hardware platform knows about a huge variety of different peripheral attachments like disk devices (also known as DASD), tapes, and communication controllers, they can all be accessed by a well-defined access method and they present I/O completion in a unified way: I/O interruptions. Every single device is uniquely identified to the system by a so-called subchannel, where the z/Architecture allows for 65,536 devices to be attached per channel set and there may be multiple channel sets. [Lung, 1999]

In order not to introduce a new I/O concept to the common Linux code, Linux on System z preserves the IRQ concept and semantically maps the subchannels to Linux as IRQs.

4.7 Floating Point

z/Architecture supports short, double, and long double IEEE floating point registers. A separate floating point control register (FPC) is used to hold status and other flags.

4.8 Control Registers

Control Registers are used to modify the operation of the z/Architecture processor. They are only accessible in "supervisor state". The operation of each register is described in the Principle of Operations manual.

4.9 Address Spaces

Just as with the Sparc architecture it is possible (and desirable) to put the kernel in a separate 64-bit address space. By operating in multiple address space mode the translation tables can be accessed concurrently to facilitate easy (controlled) exchange of data from user to kernel space. Furthermore, it is possible to place executable code in its own address space, which enables the implementation of "execute only" storage which the basic page protection mechanism (storage keys) does not.

5. Debugging under z/VM

There is a vast set of tools available to the kernel hacker under z/VM. The most useful of these is the `TRACE` command. With this command VM will trap various events or instructions within a virtual machine. You are able to instruction step and watch for changes in the contents of storage or registers and restrict to one virtual CPU out of many. The following are samples of what's available, with examples of each in use. The samples are taken from the debugging of a 32-bit Linux system.

Trace Linux `syscalls` and show me the registers and the contents addressed by register 1 when the trap is sprung. These are implemented by the Supervisor Call (SVC) instruction. On interrupt, the user address space is designated as the secondary address space:

```
TRACE SVC RUN CMD D seco0.10;BASE1 G
```

Trace all supervisor calls

Don't stop

When trap springs, issue a command to display 16 bytes of storage in the secondary address space pointed to by register 1 as well as the other general registers.

```
 -> 400AC134" SVC    0AAE      -> 0001909A'    CC 0
```

In the S/390 implementation the code X'AE' equates to the `rt_sigaction` system call.

```
-> 40043E2E" SVC    0AAE      -> 0001909A'    CC 0

V7FFFF450  00000001 00000000 00000000 00000000 06 L0565F450
```

The storage addressed by R1 is displayed as well as the contents of the registers

```
GPR  0 =  7FFFF508  7FFFF450  00000015  7FFFF3C0
GPR  4 =  7FFFF450  00000008  00424FC0  7FFFF8F8
GPR  8 =  00401B9C  7FFFF600  7FFFFE88  7FFFF570
GPR 12 =  C0114AE0  C0043D84  C0043DBC  7FFFF340
 -> 400AC174" SVC    0A39      -> 0001909A'    CC 0
V400AC174  0A39A748 F0011524 A7B40003 07FE0D10 06 L07D96174
GPR  0 =  7FFFF508  400AC174  00000190  00000190
GPR  4 =  7FFFF450  00000008  00424FC0  7FFFF8F8
GPR  8 =  0040233C  7FFFF8F8  7FFFFE88  7FFFF690
GPR 12 =  C0114AE0  80424FC8  80425240  7FFFF690
 -> 40043FD6" SVC    0AAF      -> 0001909A'    CC 0
V7FFFF588  00390000 00000000 00000000 00000000 06 L0565F588
GPR  0 =  7FFFF520  7FFFF588  00000000  7FFFF588
GPR  4 =  7FFFF608  00000008  00424FC0  7FFFF8F8
```

```
   GPR  8 =  00000000  4003847C  7FFFFE88  7FFFF520
   GPR 12 =  C0114AE0  C0043F68  C0043FA2  7FFFF498
```

Trace timer pops by tracing external interruptions of type 0x1004, which signify the popping of the S/390 clock comparator:

```
TRACE EXTERNAL 1004
*** 0001B2A0'   EXT     1004 -> 000198B2'
*** 0001B2A0'   EXT     1004 -> 000198B2'
*** 0001B2A0'   EXT     1004 -> 000198B2'
```

Trace the first X'40' bytes of instructions and any changes to the contents of registers in the I/O top half processor.

```
D PSW I/O
I/O 000C  38 OLD 070C0000 800703D0   78  NEW 04080000 800197BC
TRACE I R 197BC.40 RUN
TRACE G R 197BC.40 RUN
-> 000197BC' ST    50F00C00 >> 00000C00    CC 0
   000197C0' TM    91010039    00000039    CC 0
   000197C4' BRC   A7840004 -> 000197CC'   CC 0
-> 000197CC' AHI   A7FAFF08    CC 2    G15=056318E8
   000197D0' SRL   88F00003    00000003    CC 2    G15=00AC631D
   000197D4' SLL   89F00003    00000003    CC 2    G15=056318E8
   000197D8' STM   900EF068 >> 05631950    CC 2
   000197DC' ST    5020F0E8 >> 056319D0    CC 2
   000197E0' MVC   D203F0A40C00 >> 0563198C    00000C00    CC 2
   000197E6' STAM  9B0FF0A8 >> 05631990    CC 2
   000197EA' MVC   D207F0600038 >> 05631948    00000038    CC 2
   000197F0' XC    D706F000F000 >> 056318E8    056318E8    CC 0
   000197F6' MVI   9238F0EF >> 056319D7    CC 0
   000197FA' SLR   1F00        CC 2    G00=00000000
```

Trace I/O operations to a network driver (using network devices X'440' and X'441'):

```
TR I/O 440 INT CCW RUN
TR I/O 441 INT CCW RUN
 -> 0002059A' SSCH  B2334000    0034D344    CC 0    SCH 0001    DEV 0441
           CPA 07EFDCE0   PARM 00000000    KEY 0  FPI C0  LPM 80
VDEV 0441 CCW E3600000 00000000 STS 0C
          CCW 0160005C 05820000 STS 00 DATA 005C005A08000000 ... STS 0C
          CCW 03200000 00000000 STS 0C
*** 00078A56'   I/O DEV 0441 -> 000197BC'   SCH 0001   PARM 00000000
*** 0001B2A0'   I/O DEV 0440 -> 000197BC'   SCH 0000   PARM 00000000
 -> 0002059A' SSCH  B2334000    0034D154    CC 0    SCH 0000    DEV 0440
           CPA 07EFDC70   PARM 00000000    KEY 0  FPI C0  LPM 80
```

6. System z I/O Support

Under construction.

6.1 I/O Discovery

The anchor of all the I/O support is the ioDev structure, which describes the I/O device. It is a doubly linked list that may be added to or deleted from as the configuration changes. When a Solaris device node is created its "dip" (`dev_info_t`) is placed in this structure and a pointer to the ioDev structure is placed in the driver private data area of the dip.

```
#include <sys/ios390x.h>

typedef struct _ioDev {
        void    *next;          /* Next device                  */
        void    *prev;          /* Previous device              */
        void    *private;       /* Private data(pointer to dip) */
        uint32_t schid;         /* Subchannel id                */
        uint32_t instance;      /* Reserved                     */
        struct  schib sch;      /* Subchannel information block */
        struct  vrdcblok dev;   /* Device characteristics       */
} ioDev;

typedef struct _devList {
        int     devCount;
        ioDev   *devices;
} devList;

/*-------------------------------------------------------*/
/* ECKD Device Characteristics...                        */
/*-------------------------------------------------------*/
struct eckdchar {
        short   vrdcprim;       /* No. primary cyls     */
        short   vrdctrkc;       /* Tracks per cylinder  */
        int     vrdcsect : 8;   /* Number of sectors    */
        int     vrdctotr : 24;  /* Total track length   */
        short   vrdcha;         /* Length for R0 & HA   */
        char    vrdcmode;       /* Capacity calc mode   */
        char    vrdcmdfr;       /* Capacity calc chgd   */
        short   vrdcnkov;       /* Non-keyed overhead   */
        short   vrdckovh;       /* Keyed overhead       */
        short   vrdcaltc;       /* 1st alternate cyl    */
        short   vrdcaltr;       /* No. alternate cyls   */
        short   vrdcdig;        /* 1st diagnostic cyl   */
        short   vrdcdign;       /* No. diagnostic cyls  */
        short   vrdcdvcy;       /* 1st dev support cyl  */
        short   vrdcdvtr;       /* No. dev support cyls */
        char    vrdcmdr;        /* MDR record id        */
        char    vrdcobr;        /* OBR record id        */
        char    vrdccuid;       /* Control Unit ID      */
        char    resv00[21];     /* Unused               */
        char    vrdcpgid[11];   /* Path group id        */
        char    resv01[5];      /* Unused               */
} __attribute__ ((packed));
```

```
/*-------------------------------------------------------*/
/* FBA Device Characteristics...                         */
/*-------------------------------------------------------*/
struct fbachar {
        char    vrdcoper;       /* Operation modes      */
        char    vrdcfbaf;       /* Device features      */
        char    vrdcfbac;       /* Device class         */
        char    vrdcfbat;       /* Device type          */
        short   vrdcrcsz;       /* Physical record size */
        int     vrdcbkcg;       /* Blocks/track         */
        int     vrdcbkap;       /* Blocks/access pos    */
        int     vrdcbkma;       /* Blks under mv access */
        int     vrdcbkfa;       /* Blks under fx access */
        short   vrdcbkaa;       /* Blks in alt area     */
        short   vrdcbkce;       /* Blks in CE area      */
        short   vrdcbflg;       /* No. buffered log byts*/
        short   vrdcatmi;       /* Min. access time     */
        short   vrdcatma;       /* Max. access time     */
} __attribute__ ((packed));

/*-------------------------------------------------------*/
/* Device Characteristics...                             */
/*-------------------------------------------------------*/
struct rdvchar {
        short vrdccuty;         /* Control Unit type    */
        char  vrdccumd;         /* Control Unit model   */
        short vrdcdvty;         /* Device type          */
        char  vrdcdvmd;         /* Device model         */
        char  vrdcdvfe[3];      /* Features supported   */
        char  vrdcsdfe;         /* Subsystem features   */
        char  vrdcdvcl;         /* Device class         */
        char  vrdcdvco;         /* Device code          */
        union {
          struct eckdchar eckd; /* ECKD device chars    */
          struct fbachar fba;   /* FBA device chars     */
        } ch;
} __attribute__ ((packed));

/*-------------------------------------------------------*/
/* Virtual/Real Device Block...                          */
/*-------------------------------------------------------*/
struct vrdcblok {
        short   vrdcdvno;       /* Device number        */
        short   vrdclen;        /* Length of block      */
        char    vrdcvcla;       /* Virtual Device class */
#define DC_CONS         0x80    /* Console type device  */
#define DC_GRAF         0x40    /* 3270 type device     */
#define DC_URIN         0x20    /* Unit Record input    */
#define DC_UROT         0x10    /* Unit Record output   */
#define DC_TAPE         0x08    /* Tape type device     */
#define DC_DASD         0x04    /* ECKD Disk type device*/
#define DC_SPEC         0x02    /* Special type device  */
#define DC_FBAD         0x01    /* FBA disk type device */

        char    vrdcvtyp;       /* Virtual Device type  */
/*
 * Device types defined in DC_CONS
 */
#define DT_3215         0x00    /* 3215 console         */
/*
 * Device types defined in DC_GRAF
 */
#define DT_5080         0xc0    /* 5080 terminal        */
```

```
#define DT_2250         0x80    /* 2250 terminal        */
#define DT_3277         0x04    /* 3277 terminal        */
#define DT_3278         0x01    /* 3278 terminal        */
#define DT_3279         DT_3278 /* 3279 terminal        */
/*
 * Device types defined in DC_URIN
 */
#define DT_2520         0x90    /* 1442 card reader     */
#define DT_1442         0x88    /* 1442 card reader     */
#define DT_3505         0x84    /* 3505 card reader     */
#define DT_2540         0x82    /* 2540 card reader     */
#define DT_2501         0x81    /* 2501 card reader     */
/*
 * Device types defined in DC_UROT
 */
#define DT_3525         0x84    /* 3525 card punch      */
#define DT_1403         0x41    /* 1403 printer         */
#define DT_3211         0x42    /* 3211 printer         */
#define DT_3203         0x43    /* 3203 printer         */
#define DT_VAFP         0x48    /* VAFP printer         */
#define DT_AFP1         0x4e    /* 3820 printer         */
#define DT_AFP2         0x4f    /* 3820 printer         */
#define DT_AFP3         0x45    /* 3800 printer         */
#define DT_AFP4         0x49    /* 3800-03 printer      */
#define DT_AFP5         0x4d    /* 3800-08 printer      */
#define DT_3262         0x47    /* 3262 printer         */
#define DT_4245         0x4a    /* 4245 printer         */
#define DT_4248         0x4b    /* 4248 printer         */
/*
 * Device types defined in DC_TAPE
 */
#define DT_NTAP         0x90    /* New tape product     */
#define DT_3590         0x83    /* 3590 tape            */
#define DT_3422         0x82    /* 3422 tape            */
#define DT_3490         0x81    /* 3490 tape            */
#define DT_2401         0x80    /* 2401 tape            */
#define DT_9348         0x44    /* 9348 tape            */
#define DT_3424         0x42    /* 3424 tape            */
#define DT_3420         0x10    /* 3420 tape            */
#define DT_3410         0x08    /* 3480 tape            */
#define DT_3411         DT_3410 /* 3480 tape            */
#define DT_8890         0x04    /* 8809 tape            */
#define DT_3430         0x02    /* 3430 tape            */
#define DT_3480         0x01    /* 3480 tape            */
/*
 * Device types defined in DC_DASD
 */
#define DT_3390         0x82    /* 3390 disk            */
#define DT_9345         0x81    /* 9345 disk            */
#define DT_2311         0x80    /* 2311 disk            */
#define DT_2301         DT_2311 /* 2301 disk            */
#define DT_2303         DT_2311 /* 2303 disk            */
#define DT_2321         DT_2311 /* 2321 disk            */
#define DT_2314         0x40    /* 2314 disk            */
#define DT_2319         DT_2314 /* 2319 disk            */
#define DT_3380         0x20    /* 3380 disk            */
#define DT_3330         0x10    /* 3330 disk            */
#define DT_3333         DT_3330 /* 3330 disk            */
#define DT_3375         0x04    /* 3375 disk            */
#define DT_2305         0x02    /* 2305 disk            */
#define DT_3340         0x01    /* 3340 disk            */
/*
 * Device types defined in DC_SPEC
```

```
 */
#define DT_CTCA         0x80    /* CTCA/3088 device     */
#define DT_37XX         0x40    /* 3704/3705/3725       */
#define DT_OSAD         0x20    /* OSA device           */
/*
 * Device types defined in DC_FBAD
 */
#define DT_9336         0x40    /* 9336 fba disk        */
#define DT_0671         0x20    /* 0671 fba disk        */
#define DT_9313         0x10    /* 9313 fba disk        */
#define DT_9332         0x08    /* 9332 fba disk        */
#define DT_9335         0x04    /* 9335 fba disk        */
#define DT_3370         0x02    /* 3370 fba disk        */
#define DT_3310         0x01    /* 3310 fba disk        */

        char    vrdcvsta;       /* Virtual Device status*/
        char    vrdcvfla;       /* Virtual Device flags */
        char    vrdcrcca;       /* Real Device class    */
        char    vrdccrty;       /* Real Device type     */
        char    vrdccrmd;       /* Real Device model    */
        char    vrdccrft;       /* Real Device features */
        char    vrdcundv;       /* Underlying dev code  */
#define VRDCTNAT        0x00    /* Native non-emulated  */
#define VRDCT120        0x01    /* 3590/3592(128)->3490E*/
#define VRDCTVTS        0x02    /* 3490E within 3494    */
#define VRDCT121        0x03    /* 3590/3591(128)->3490E*/
#define VRDCT255        0x09    /* 3590/3592(256)->3590 */
#define VRDCT254        0x0a    /* 3590/3592(256)->3490E*/
#define VRDCT384        0x0b    /* 3590/3592(384)->3590 */
#define VRDCT383        0x0c    /* 3590/3592(384)->3490E*/
#define VRDCT512        0x10    /* 3590/3592(512)->3590 */
#define VRDCT511        0x11    /* 3590/3592(512)->3490E*/
#define VRDCTUNK        0xff    /* Unknown              */
        char    vrdcrdaf;       /* Additional features  */
#define VRDCFCDS        0x80    /* Dataset level flash  */
#define VRDCFCFV        0x20    /* Full volume flash    */
        struct rdvchar  dvc;    /* Device characteristic*/
} __attribute__ ((packed));
```

6.1.1 Device Discovery – s390x/os/ioinit.c

1. Count the number of devices via STSCH loop (the STSCH instruction will return CC=3 when there are no more devices)
2. Allocate storage for ioDev structures – use devList as anchor for the structures
3. Create double linked list of ioDev entries for each subchannel:
    - Store SCHIB within sch field
    - Use the hypervisor DIAG 0x210 call to place the device characteristics in the dvc field
    - Determine console device via DIAG 0x24

6.1.2 Nexus Node Creation – s390x/os/ddi_impl.c

The CCW nexus device node is created using a call to ndi_devi_alloc(). Directly after this the device node creation begins.

6.1.3 Device Node Creation – zSeries/io/ccw/ccw_autoconfig.c

Called as part of DDI setup to add “reg” and “interrupts” propositions to CCW nexus device node and later during I/O configuration to probe the devices.

The probe process consists of going through the IODev list -

- Determine if it's a device we'll support (e.g. disk, tape, OSA, console)
- Create a device node instance using ddi_add_child()
- Set the “compatible” property of the device
- Link the dip to the ioDev entry using ddi_set_driver_private()
- Enable the device for I/O via MSCH instruction which modifies the “enable” field within the subchannel block (SCHIB).

6.1.4 CCW Nexus Driver – s390x/io/ccwnex.c

This module handles devices being attached and detached from the system. It will field the machine check that is used by System z to inform the kernel when a device is either becomes present or is removed from the configuration.

7. OpenSolaris Source Control

The source repository for OpenSolaris is under the control of the mercurial (`hg`) source tool. It is similar to CVS and SVN but has its own command set and way of doing things. Updates are committed and pushed to this server. In turn, the server's repository may be kept current with the "official" source tree using the same set of mercurial commands.

7.1 Cloning the Source Tree

The source tree may be cloned from the mercurial server running on (for the sake of example) "sirius" using the commands:

1. `cd ~/OpenSolaris/sirius`
2. `hg clone http://localhost:8000/onnv-gate`

For a tutorial on how to use the mercurial command set see the wiki page at: http://www.genunix.org/wiki/index.php/Mercurial

7.2 Entering the Build Environment

To begin work on the system enter the following commands after logging in:

1. `cd ~/OpenSolaris/sirius`
2. `bldenv -d opensolaris.sh`
3. `.addenv.sh`

7.3 Source Structure

All ON sources are found under `~/OpenSolaris/sirius/onnv-gate/usr/src`. This includes both the sources used to build the ON consolidation and sources for tools and other peripheral utilities needed to build but not shipped as part of Solaris. The `onnv-gate/usr/src` directory has several subdirectories which are described here.

- **cmd**: This directory contains sources for the executable programs and scripts that are part of ON. It includes all the basic commands, daemons, startup scripts, and related data. Most subdirectories are named for the command or commands they provide; however, there are some exceptions listed here.
- **common**: Files which are common among `cmd`, `lib`, `stand`, and `uts`. These typically include headers and sources to basic libraries used by both the kernel and user programs.
- **head:** Userland header files (kernel headers are in `uts/`). Note that only libc headers should be stored here; other libraries should have their headers in their own subdirectories under `lib/`.

- **lib:** Libraries. Most subdirectories are named for the library whose sources they contain or are otherwise self-explanatory.
- **pkgdefs:** Contains one subdirectory for each package generated from the ON sources. Each subdirectory contains packaging information files; see `pkginfo`(4), `depend`(4), `prototype`(4), `pkgmap`(1), and `pkgproto`(1) for more information about the contents of these files.
- **prototypes:** Sample files showing format and copyright notices.
- **psm:** Platform-specific modules. Currently this contains only Open Boot PROM (OBP) and most of the boot code (except for s390x).
- **stand:** Standalone environment code. This is used for booting; for example, code for reading from UFS and the network is here.
- **tools:** Development tools and sources. See chapter on tools later in this document for more information about each tool; the file should be updated as tools are added or removed.
- **ucbcmd:** Commands and daemons installed into `/usr/ucb` (for SunOS 4.x compatibility).
- **ucbhead:** Header files installed into `/usr/ucb` (for SunOS 4.x compatibility).
- **ucblib:** Libraries installed into `/usr/ucb` (for SunOS 4.x compatibility).
- **uts:** Kernel sources are here (UTS == UNIX Time Sharing). There are numerous subdirectories of `uts` which are of interest.

7.4 Kernel Makefile Structure

The advent of dynamic loading of kernel modules has obsoleted the 4.x kernel configuration scheme which was centered around a derived Makefile and a collection of derived header files generated by the `config`(8) program. This file describes the structure of the replacement “static” set of Makefiles.

Some additional secondary goals were associated with the generation of these Makefiles. It should be noted that the ability to properly deal with derived Makefiles is an explicit non-goal of the ongoing NSE enhancements, so this project is a necessary consequence of that decision.

All project goals are enumerated below:

1. To provide a set of static Makefiles to support kernel build and installation.
2. To provide a set of static Makefiles which conform to the “Makefiles Guidelines”. (This document is currently available on-line as "`terminator:/usr/integration/doc/make.std`")
3. To completely eliminate the `config`(8) program.

4. To provide a framework for linting the kernel (so that “lint free” can be made an integration criterion, in addition to being general good hygiene).
5. To eliminate the need for the small headers generated by `config`(8). In the ddi/dki world this need is completely eliminated as drivers will be expected to dynamically configure themselves. Interim support for existing drivers will be provided.
6. To be able to “acquire” only the files needed to build a specific module, if that is all that is needed.
7. To provide a framework suitable for the production of “implementation architecture” independent modules.
8. To restructure the assembly language files to support the generation of “lint-libraries” from them.
9. To provide support for the incidental Makefile targets many developers are accustomed to (such as `cscope` and `tags`). These can be added to the Makefiles as required. (`cscope` is currently supported.)

7.4.1 General Structure

The source code layout is not generally affected by the Makefiles. However, the location of the generated files has changed dramatically.

“Implementation architecture” independent modules are produced in individual directories (one per module) under the “instruction-set architecture” directory (e.g.: `sparc`). Similarly, “implementation architecture” dependent modules are produced in individual directories under the “implementation architecture” directory (e.g.: `sun4`, `sun4c`).

The driving Makefile for any module is located in the leaf directory where the module (and associated objects) are built. After a '`make clobber`' operation, the Makefile is the only file remaining in that directory. Common definitions and rules are contained in suffixed Makefiles in non-leaf directories which are included in the leaf Makefiles. Non-suffixed Makefiles in non-leaf directories generally invoke lower level Makefiles to perform the actual tasks.

```
uts/Makefile
uts/sparc/Makefile
uts/sun4c/Makefile
uts/sun4c/svvs/Makefile
```

These Makefiles generally are cognizant of the components made in subdirectories and invoke Makefiles in those sub-directories to perform the actual build. Some targets (or pseudo-targets) may be directly built at this level (such as the `cscope` databases).

```
uts/Makefile.uts
```

Contains common definitions for all possible architectures.

`uts/Makefile.targ`

Contains common targets for all possible architectures.

```
uts/common/Makefile.files
uts/sun/Makefile.files
uts/sparc/Makefile.files
uts/sun4c/Makefile.files
uts/sun4/Makefile.files
```

These Makefiles are divided into two sections. The first section can be viewed as the equivalent of the "files" (`sparc` and `sun4c`) and "files.cmn" (common and sun) files. These define the object lists which define each module. The second section defines the appropriate header search paths and other machine specific global build parameters.

```
uts/common/Makefile.rules
uts/sun/Makefile.rules
uts/sparc/Makefile.rules
uts/sun4c/Makefile.rules
uts/sun4/Makefile.rules
```

The files provide build rules (targets) which allow make to function in a multiple directory environment. Each source tree below the directory containing the Makefile has a build rule in the file.

```
uts/sun4c/Makefile.sun4c
uts/sun4/Makefile.sun4
```

These Makefile contains the definitions specific (defaults) to the obvious "implementation architecture". These rules can be overridden in specific leaf node Makefiles if necessary.

`uts/sun4c/unix/Makefile`

Main driving Makefile for building /unix.

`uts/sun4c/MODULE/Makefile (for MODULE in arp, aoutexec, ...)`

Main driving Makefile for building MODULE.kmod.

`uts/sun4c/unix.static/Makefile`

Main driving Makefile for building a static unix (for development work only). This Makefile is known to NSE, but its targets are not. This makefile may be copied to additional parallel directories to build multiple configurations. This configuration is roughly equivalent to the GENERIC kernel of SunOS 4.x.

`uts/*/Makefile.?.shared`

These denote Makefile contents which are shared between open and closed builds.

The Makefiles are verbosely commented. It is desired that they should stay this way.

7.4.2 Use

- Issuing the command 'dmake' in the `uts` directory will cause all supported, modularized kernels and modules to be built.
- Issuing the command 'dmake' in a `uts/`*`ARCHITECTURE`* directory (e.g.: `uts/sparc`) will cause all supported, "implementation architecture" independent modules for *ARCHITECTURE* to be built.
- Issuing the command 'dmake' in a `uts/`*`MACHINE`* directory (e.g.: `uts/sun4c`) will cause that kernel and all supported, "implementation architecture" dependent modules for *MACHINE* to be built.
- Issuing the command 'dmake' in the `uts/`*`MACHINE`*`/unix` directory will cause the kernel for MACHINE to be built (and `unix.o`).
- Issuing the command 'dmake' in a `uts/`*`MACHINE`*`/`*`MODULE`* or a `uts/`*`ARCHITECTURE`*`/`*`MODULE`* directory will cause *`MODULE`*`.kmod` to be built.

7.4.3 Lint

Linting is fairly similar to the builds, but it has an additional complication. In order to get meaningful output from `lint` pass2, all the modules must be `linted` together. This is accomplished by each module being responsible to produce its own pass1 output (*`file`*`.ln`, one per `.c/.s` file). It is also responsible for placing the a lint-library (`llib-l`*`MODULE`*) in the `uts/`*`MACHINE`*`/lint-libs` directory. The final full lint is accomplished by the Makefile in the `uts/`*`MACHINE`* directory by `linting` all the lint-libraries against each other.

Note that there is no equivalent to `locore.c` in the current source base. The C prototypes are in the `.s` files. As example:

```
#if defined(lint)
int
blort(int, int)
{ return 0 }
#else   /* lint */

        ENTRY(blort)
        ld      [%i0],....
        ....
        SET_SIZE(blort)

#endif  /* lint */
```

7.4.4 Component Hierachy

The component hierarchy has been restructured to allow the acquisition of more finely grained objects; specificly a kernel module. The basic component structure is:

```
:src:uts.all --+--> :sparc --+--> :MODULES...
               |
               +--> :sun4c --+--> :unix
               |             |
               |             +--> :MODULES...
               |             |
               |             +--> :unix.static
               |
               +--> :sun4 ---+--> :unix
               |             |
               |             +--> :MODULES...
               |             |
               |             +--> :unix.static
               ...
```

The above diagram does not reflect the full component tree. The full component tree may be displayed with the "`nsecomp list -r :src:uts.all`" command.

7.5 Common Operations

7.5.1 Adding a New Kernel Module

1. Create the source files (and directories) as usual.
2. Edit `uts/*/Makefile.files` to define the set of objects. By convention the symbolic name of this set is of the form `MODULE_OBJS`, where `MODULE` is the module name (e.g.: `namefs`). The files in each subtree should be defined in the `Makefile.files` in the root directory of that subtree. Note that they are defined using the += operator, so that the set can be built across multiple files. As example:

   ```
   NAMEFS_OBJS +=     namevfs.o namevno.o
   ```

3. Each source file needs a build rule in the corresponding `Makefile.rules` file (compilation and lint). A typical pair of entries would be:

   ```
   $(OBJS_DIR)/mem.o:          $(UTSBASE)/sun4c/io/mem.c
      $(COMPILE.c) -o $@ $(UTSBASE)/sun4c/io/mem.c

   $(LINTS_DIR)/mem.ln:                $(UTSBASE)/sun4c/io/mem.c
      @($(LHEAD) $(LINT.c) $(UTSBASE)/sun4c/io/mem.c $(LTAIL))
   ```

4. Create build directories in the appropriate places. If the module can be built in a machine independent way, this would be in the "instruction set architecture" directory (e.g.: `sparc`). If not, these directories would be created for all appropriate "implementation architecture" dependent directories (e.g.: `sun4`, `sun4c`).
5. In each build directory, create a Makefile. This can usually be accomplished by copying a Makefile from a parallel directory and editing the following lines (in addition to comments).

```
MODULE          = namefs
```

- replace with module name

```
OBJECTS         = $(NAMEFS_OBJS:%=$(OBJS_DIR)/%)
LINTS           = $(NAMEFS_OBJS:%.o=$(LINTS_DIR)/%.ln)
```

- replace with MODULE_OBJS

```
ROOTMODULE      = $(ROOT_FS_DIR)/$(MODULE).kmod
```

- replace directory part with the appropriate installation directory name (see `Makefile.uts`)

If a custom version of `modstubs.o` is needed to check the undefines for this routine, the following lines need to appear in the Makefile (after the inclusion of `Makefile.`*`mach`* - e.g.: `Makefile.sun4c`).

```
MODSTUBS_DIR     = $(OBJS_DIR)
$(MODSTUBS_O)    := AS_CPPFLAGS += -DNAMEFS_MODULE
```

- replace "`-DNAMEFS_MODULE`" with the appropriate flag for the `modstubs.o` assembly.

```
CLEANFILES      += $(MODSTUBS_O)
```

6. Edit the parent `Makefile.`*`mach`* (e.g.: `Makefile.sun4c`) to know about the new module:

```
FS_KMODS        += fd fifo namefs nfs proc spec ufs
```

Any additional questions can be easily answered by looking at the many existing examples.

7.5.2 Moving a Module to the "implementation architecture" Independent Build

1. Create the build directory under the appropriate "instruction set architecture" build directory (e.g.: `sparc/`*`MODULE`*).
2. Move the Makefile from the "implementation architecture" build directory (e.g.: `sun4c/`*`MODULE`*) to the directory created above. Edit this Makefile to reflect the change of parent (trivial: comments, paths and includes).

3. Edit the "implementation architecture" directory Makefile (e.g.: `Makefile.sun4c`) to *not* know about this module and edit the "instruction set architecture" directory Makefile (e.g.: `Makefile.sparc`) to know about it.

8. Tools

This directory contains the tools used to do a full build of the OS/Net workspace. They usually live in the `/opt/onbld` directory on build machines. From here, 'make install' will build and install the tools in `$ROOT/opt/onbld`. If you like, '`make pkg`' will build the `SUNWonbld` package in `$(PKGARCHIVE)`. Installing that package will populate the `/opt/onbld` directory, and create a root account for building called '`gk`', which uses `csh` and has a home directory of `/opt/onbld/gk`. You can use this account to do full builds with '`nightly`'. You don't have to, but the '`gk`' account has the path setup properly, has a `.make.machines` file for `dmake`, and has a `.login` that sets up for `dmake`.

8.1 Layout of /opt/onbld

`/opt/onbld/etc/abi`

Contains Solaris ABI database (`ABI_*.db`) and exceptions for ABI Auditing tool (`intf_check`).

`/opt/onbld/gk`

gk account's home directory.

`/opt/onbld/bin`

Basic bin directory - contains scripts.

`/opt/onbld/bin/${MACH}`

Architecture-specific bin directory for binaries.

`/opt/onbld/env`

Build environment files.

`/opt/onbld/man`

Rudimentary man pages for some of the tools.

8.2 Tool Summary

This section describes the tools used in the build process.

8.2.1 bfu

bonwick/faulkner upgrade. Loads a set of `cpio` archives created by '`mkbfu`' onto a machine, either live or on alternate `root` and `/usr` filesystems. Attempts to preserve important files, but may require manual intervention before reboot to resolve changes to preserved files.

8.2.2 bfuld

Used by `bfu` to survive getting a new runtime linker when extracting new `cpio` archives onto a live system. Patches binaries to use a saved runtime linker in /tmp during the `bfu` process. Not run by anything but `bfu`.

8.2.3 bldenv

Companion to '`nightly`.' Takes the same environment file you used with '`nightly`,' and starts a shell with the environment set up the same way as '`nightly`' set it up. This is useful if you're trying to quickly rebuild portions of a workspace built by '`nightly`'. '`ws`' should not be used for this since it sets the environment up differently and may cause everything to rebuild (because of different `-I` or `-L` paths).

8.2.4 build_cscope

Builds `cscope` databases in the `uts`, the platform subdirectories of `uts`, and in `usr/src`. Uses `cscope-fast`.

8.2.5 check_rtime

Checks ELF attributes used by ELF dynamic objects in the `proto` area. Used by '`nightly`'s `-r` option, to check a number of ELF runtime attributes for consistency with common build rules. `nightly` uses the `-o` option to simplify the output for `diff`ing with previous build results. It also uses the `-i` option to obtain `NEEDED` and `RUNPATH` entries, which help detect changes in software dependencies and makes sure objects don't have any strange runpaths like `/opt/SUNWspro/lib`.

8.2.6 checkproto

Runs `protocmp` and `protolist` on a workspace (or uses the environment variable `CODEMGR_WS` to determine the workspace). Checks the `proto` area against the packages.

8.2.7 codereview

Given two filenames, creates a postscript file with the file differences highlighted.

8.2.8 codesign

Tools for signing cryptographic modules using the official Sun release keys stored on a remote signing server. This directory contains `signit`, a client program for signing files with the signing server; signproto, a shell script that finds crypto modules in `$ROOT` and signs them using `signit`; and

`codesign_server.pl`, the code that runs on the server. The `codesign_server` code is not used on an ON build machine but is kept here for source control purposes.

8.2.9 cscope-fast

The fast version of `cscope` that we use internally. Seems to work, but may need more testing before it's placed in the gate. The source just really needs to be here.

8.2.10 cstyle

Checks C source for compliance with OS/Net guidelines.

8.2.11 ctfconvert

Convert symbolic debugging information in an object file to the Compact ANSI-C Type Format (CTF).

8.2.12 ctfdump

Decode and display CTF data stored in a raw file or in an ELF file.

8.2.13 ctfmerge

Merge the CTF data from one or more object files.

8.2.14 depcheck

A tool to try an assess the dependencies of executables. This tool is not a definitive dependency check, but it does use "`strings`" and "`ldd`" to gather as much information as it can. The dependency check tool can handle filenames and pkgnames. Before using the dependency checker you must build a database which reflects the properties and files in your system.

8.2.15 elfcmp

Compares two ELF modules (e.g. .o files, executables) section by section. Useful for determining whether "trivial" changes - cstyle, lint, etc - actually changed the code. The `-S` option is used to test whether two binaries are the same except for the `elfsign` signature.

8.2.16 elfsign

Built from the same sources as the shipped `elfsign`(1), this version is used in `nightly -t` builds to assure that the signing process and format is the same as will be used on the target system.

8.2.17 elfsigncmp

This script can be used in lieu of `elfsign` during a build. It uses `elfsign` to sign a copy of the object and `elfcmp -S` to verify that the signing caused no damage before updating the object to be signed.

8.2.18 findunref

Finds all files in a source tree that have access times older than a certain time and are not in a specified list of exceptions. Since '`nightly`' timestamps the start of the build, and findunref uses its timestamp (by default), this can be used to find all files that were unreferenced during a nightly build). Since some files are only used during a SPARC or Intel build, '`findunref`' needs to be run on workspaces from both architectures and the results need to be merged. For instance, if `$INTELSRC` and `$SPARCSRC` are set to the `usr/src` directories of your Intel and SPARC nightly workspaces, then you can merge the results like so:

```
$ findunref $INTELSRC $INTELSRC/tools/findunref/exception_list | \
  sort > ~/unref-i386.out
$ findunref $SPARCSRC $SPARCSRC/tools/findunref/exception_list | \
  sort > ~/unref-sparc.out
$ comm -12 ~/unref-i386.out ~/unref-sparc.out > ~/unref.out
```

8.2.19 hdrchk

Checks headers for compliance with OS/Net standards (form, includes, C++ guards).

8.2.20 install.bin

Binary version of `/usr/sbin/install`. Used to be vastly faster (since `/usr/sbin/install` is a shell script), but may only be a bit faster now. One speedup includes avoiding the name service for the well-known, never-changing password entries like 'root' and 'sys.'

8.2.21 intf_check

Detects and reports ABI versioning and stability problems.

8.2.22 lintdump

Dumps the contents of one or more lint libraries; see `lintdump`(1)

8.2.23 keywords

Checks files for proper SCCS keywords.

8.2.24 makebfu

Simple wrapper around 'mkbfu' for use outside nightly (when in a build shell from 'ws' or 'bldenv').

8.2.25 mkbfu

Makes cpio archives out of the proto area suitable for bfu'ing. Used by 'nightly' and 'makebfu'.

8.2.26 nightly

Nightly build script. Takes an environment (or 'env') file describing such things as the workspace, the parent, and what to build. See env/developer and env/gatekeeper for sample, hopefully well-commented env files.

8.2.27 pmodes

Enforces proper file ownership and permissions in pkgmap and package prototype* files. Converts files if necessary.

8.2.28 protocmp

Compares proto lists and the package definitions. Used by nightly to determine if the proto area matches the packages, and to detect differences between a child's proto area and a parent's.

8.2.29 protocmp.terse

Transforms the output of protocmp into something a bit more friendly

8.2.30 protolist

Create a list of what's in the proto area, to feed to protocmp.

8.2.31 sccscp

Copy a file under SCCS control to another location in a workspace. Also updates teamware's nametable.

8.2.32 sccshist

Display the history, comments and diffs, of a file under SCCS control.

8.2.33 sccsmv

Rename a file under SCCS control to another location in a workspace. Also updates teamware's nametable.

8.2.34 sccsrm

Delete a file under SCCS control workspace. also updates teamware's nametable. Actually renames it to `.del-<file>-`date` so that others will see it move when it is brought over (in case they were working on it).

8.2.35 tokenize

Used to build the `sun4u` boot block.

8.2.36 webrev

Generates a set of HTML pages that show side-by-side `diffs` of changes in your workspace, for easy communication of code review materials. Can automagically find edited files or use a manually-generated list; knows how to use `wx`'s active file for lists of checked-out files and proposed SCCS comments.

8.2.37 ws

Creates a shell with the environment set up to build in the given workspace. Used mostly for non-full-build workspaces, so it sets up to pull headers and libraries from the `proto` area of the parent if they aren't in the child's `proto` area.

8.2.38 wsdiff

Detect object differences between two ON `proto` areas. Used by `nightly`(1) to determine what changed between two builds. Handy for identifying the set of built objects impacted by a given source change. This information is needed for patch construction.

8.2.39 wx

A great workspace tool by bonwick. See wx.README for information and warnings.

8.3 How to do a full build

- Find an environment file that might do what you want to do. If you're just a developer wanting to do a full build in a child of the gate, copy the 'developer' environment file to a new name (private to you and/or

the work being done in this workspace, to avoid collisions with others). Then edit the file and tailor it to your workspace. Remember that this file is a shell script, so it can do more than set environment variables.

- Login as 'gk' (or root, but your PATH and .make.machines for dmake will not be right). Run 'nightly' and give it your environment file as an option. 'nightly' will first look for your environment file in /opt/onbld/env, and if it's not there then it will look for it as an absolute or relative path. Some people put their environment files in their workspace to keep them close.
- When 'nightly' is complete, it will send a summary of what happened to $MAILTO. Usually, the less info in the mail the better. If you have failures, you can go look at the full log of what happened, generally in $CODEMGR_WS/log/log.<date>/nightly.log (the mail_msg it sent and the proto list are there too). You can also find the individual build logs, like 'make clobber' and 'make install' output in $SRC, under names like clobber-${MACH}.out and install-${MACH}.out (for a DEBUG build). These will be smaller than nightly.log, and maybe more searchable.

8.4 Files you have to update to add a tool

1. Add the tool in its appropriate place.
2. Update the Makefile as required.
3. Update `usr/src/tools/SUNWonbld/prototype_*`
4. Update `usr/src/tools/README.tools`
5. Repeat 1-4 for any `man` pages.

9. An Overview of the Porting Process

The following sections come from the blog of Tim Marsland the CTO of Software at Sun who was the leader in the port of Solaris to x86/amd64. It gives insight into the porting process and served as a guide to the System z port.

9.1 Solaris 10 on x64 Processors: Part 1 - Prework

9.1.1 SSE Support

Back in 2002, after we had resurrected Solaris on x86, we realized that we needed to get back to basics with a number of core kernel subsystems, because while we'd been slowly disinvesting in Solaris on x86, the x86 hardware world had been scampering off doing some really interesting things e.g. the Streaming SIMD Extensions (SSE) to the instruction set, and introducing fast system call mechanisms. We also knew that these were basic assumptions of the 64-bit architecture that AMD was working on, so we started work on the basic kernel support which allows the xmm and mxcsr registers to be part of the state of every Solaris lwp. At the same time, Matt Simmons helped out with the disassembler and debugger support for the new instructions. This didn't take too long, and the work was integrated into Solaris 10 in October 2003, and Solaris 9 Update 6. One of the immediate benefits was Java floating point performance which used SSE instructions on capable platforms, and on the right hardware, Solaris was now one of those platforms!

9.1.2 Fast System Calls

In earlier Solaris releases, system calls were implemented using the lcall instruction; we'd been faithfully doing this for years, without really noticing that the performance of call gates was falling further and further behind. For Solaris 10, we decided to make fast system calls work using the sysenter instruction that was first introduced on Pentium II processors. Because of some awkward limitations around the register usage of the sysexit instruction (in particular, dealing with system calls that return two values), plus our desire to run on older machines, we also keep the older lcall handler around too.

First, I should explain something about the way Solaris system call handlers work in general. As you can imagine, in a highly observable system like Solaris, we can, in some circumstances, end up doing a lot of work in the system call enter and system call exit path. But, most of the time, we don't actually need to do all the checks, so more than 10 years ago, one of my former colleagues restructured SPARC system calls to do a single test for all

possible pre-work on the per-lwp thread variable called t_presys, and a single test for all possible post-system call handler work on another thread variable called t_postsys. The system call handler is then constructed assuming that the t_presys and t_postsys cases are rare - but if either t_presys or t_postsys is set e.g. by the current system call, previous system call, or via /proc, we handle the relevant rare case in C code, allowing us to code the fast path in a small amount of assembler. To summarize:

```
entry_point:
            if (curthread->t_presys)
                 presys();

            real_handler();

            if (curthread->t_postsys)
                 postsys();

            return-from-trap sequence
```

Obviously the Solaris x86 architecture mirrored this to some extent, but the presys() and postsys() functions had been partially rendered from C into assembler which was, as usual, difficult to understand, port and maintain, and wasn't even particularly fast. So the initial exercise was to turn the slow paths back into C code, and macro-ize the assembler code involved in performing the pre and post checks so that different syscall handlers could easily share code. Then I coded up a sysenter style handler, and we were pretty impressed with the results on our system call microbenchmarks.

9.1.3 Hardware Capability Architecture

All this kernel work was fun, but we didn't have a clear idea of how we were going to let libc use fast instructions on machines capable of handling them, and fall back to lcall on machines that couldn't. We also noted that when AMD processors are running in long mode, sysenter is not supported but syscall (similar but different) is.

Earlier attempts to introduce support for this facility had considered using either the libc_psr mechanism that we introduced in Solaris 2.5 for dealing with the fast bcopy instructions available on UltraSPARC platforms, or using the isalist mechanism. The former scheme assumes that the instruction set extensions were specific to a platform, while the latter implicitly assumes that there are a small set of instructions extensions that were additive, and acted to improve performance. However we realized that in the x86 world we weren't dealing with platform extensions so much as processor extensions, and that processor vendors were adding instruction set extensions orthogonally, so we'd be better describing each instruction set extension by close analogy to the way the vendors were describing them in the cpuid instruction i.e. via a bit value in a feature word. See getisax(3C) for programmatic access to the kernel's view; <sys/aux_386.h> contains the list of capabilities we expose.

What we ended up with is (currently) three copies of the libc binary compiled different ways; the basic version in /lib/libc.so.1 is able to run on the oldest hardware we support, the newer versions in /usr/lib/libc correspond to more modern hardware running on a 32-bit or 64-bit kernel. Thanks to Rod Evans the libraries are marked with the capabilities they require, and the system figures out which is the best library to use on the running system at boot time. Last November, Darren Moffat wrote something up about how the system configures which libc it uses; there's no point in repeating that here.

9.1.4 The Tool Chain

The other key piece we needed for the amd64 kernel was to make the Solaris kernel compile and run with the GNU C compiler and assembler, so I started work on that too. Note that wasn't because we didn't want to use the Sun compiler, it's just that it's easier to bring up an OS using a compiler that works, instead of debugging both the kernel and the compiler simultaneously. More critically, there wasn't a Sun compiler that would build 64-bit objects at the time. GNU C is great for finding bugs, and really complimented the capabilities of the Sun compiler and lint tools. I got the 32-bit kernel working fairly easily, we were able to start the 64-bit project using this compiler, once we'd hacked up an initial configuration.

In the meantime, while we were completing and integrating some of these prerequisites into Solaris 10, we were assembling the main amd64 project team; work really started in earnest in January of 2004.

9.2 Solaris 10 on x64 Processors: Part 2 - Getting Started

9.2.1 Booting and Startup

Some of the trickier issues with porting Solaris to a new platform architecture originate from some of the decisions we made 15 years ago. This is a complex story, and one that we're soon to drastically improve on x86/x64 systems with newboot , (as demoed recently by Jan Setje-Eilers at the first OpenSolaris User Group meeting) but I'll try to relate the minimum needed so that you understand why we took the path that we did for Solaris on x64 processors.

The Solaris 2 booting system was originally designed back in 1991 in the context of the problems and issues we had in the world of SunOS 4.x, our desires to have the hardware and software organizations in the company execute relatively independently, as well as to support the then-blossoming clone market for SPARC workstations. We wanted to enable both ourselves and 3rd parties to deliver support for new hardware without having to change the Solaris product CD bits - except by adding new driver modules to it. Also remember that speeds and feeds were two orders of magnitude slower then, and that kernel text was a larger proportion of the minimum

memory size than it is today so we had to be far more selective about which modules to load as we booted.

The design we came up with starts with the primary bootstrap loading the secondary booter, and the secondary booter putting the core kernel into memory. The kernel then turns around and invokes various services from the secondary booter using the bootops interface to allow the kernel to discover, load and assemble the drivers and filesystem modules it needs as it initializes itself and explores the system it finds itself on. Once it determines it has all the modules it needs to mount the root filesystem, it takes over IO and memory allocation, mounts the root, and (if successful) continues to boot by loading additional kernel modules from that point on.

Note that this early part of the boot process starts out with the secondary boot program being the one true resource allocator i.e. in charge of physical memory, virtual memory and all I/O, and ends with the kernel being that resource allocator at the end. While moving from the former state to the latter sounds simple in principle, it's quite complex in practice because of the incremental nature of the handoff. For example, the DDI (device driver interfaces) aren't usable until the kernel has initialized its infrastructure for managing physical and virtual memory. So we have to load the modules we might need based on the name of the root device given to us by the booter which in turn comes from OpenBoot. Kernel startup somehow has to get most of the VM system initialized and working, yet still allow the boot program and its underlying firmware to do I/O to find the drivers and filesystem modules it needs from the boot filesystem to mount the filesystem. Practically speaking, this entails repeated resynchronization between the boot program and the kernel over the layout and ownership of physical and virtual memory. In other words, the kernel tries to take over physical and virtual memory management while effectively avoiding conflicting with the secondary booter and the firmware using the same resources. It's really quite a dance.

For the x86 port, a similar approach was used, using real-mode drivers that were placed onto a boot floppy as an analogue of the OpenBoot drivers in the SPARC world to construct a primitive device tree data structure analogous to the OpenBoot device tree. (In 1995, for the PowerPC port, we implemented "virtual open firmware" which was an even closer simulation of OpenBoot to make it easier to reuse SPARC boot and configuration code).

Note that the x86 secondary boot program itself runs in protected mode like the kernel; it is responsible for switching to real-mode and back to run the real-mode drivers.

9.2.2 Six years go by ...

During that time, specifically in Solaris 2.5, we made things even more complicated for hardware bringup by splitting the basic kernel into separate modules: unix, genunix and the kernel linker krtld; these make

bringup more difficult because genunix and krtld are not relocated until run-time, thus diagnosing hexadecimal addresses where the kernel has fallen over in genunix or in krtld becomes a significant pain, absent a debugger like kmdb or its predecessor kadb.

Now, in 1997 when we created 64-bit Solaris for SPARC, it was relatively simple to make a 64-bit boot program use the OpenBoot interfaces; there is no "mode" switch between operating a SPARC V9 processor in "32-bit" mode or "64-bit" mode - apart from the address mask, the rest of the difference was entirely about software conventions, not hardware per se. So we didn't really have to do anything to the basic boot architecture, this part of the 64-bit Solaris project on SPARC was really quite straightforward, mostly a matter of getting the boot program LP64-clean.

9.2.3 Meanwhile, in late 2003 ...

The Opteron architecture presented us a far more difficult challenge because the processor needed to switch to long mode via an arcane sequence of instructions including switching between different format page tables and descriptor tables. Worse still, the 64-bit Solaris kernel on Opteron would need to turn around and invoke what it would think of as a 64-bit boot program running in long mode in order to fetch modules from the disk (and as discussed above) invoking real-mode code and the BIOS to do so!

Our initial approach was to use the existing, unmodified, protected mode 32-bit booter and have it boot an interposer called vmx that used the 32-bit booter to load the 64-bit kernel into double mapped memory. In this case, "double mapped" means that there's an (up to) 4Gbyte area of physical memory that are (a) mapped by 32-bits of VA in the protected mode page tables and (b) mapped by the bottom 32-bits of VA and the top 4G of VA in the long mode page tables. The interposer then pretended to be a 64-bit booter to the 64-bit kernel. When the 64-bit kernel asked vmx for a boot service via one the bootops vector (fortunately a relatively small and well-behaved interface), vmx quickly switched back to protected mode, then after massaging the arguments appropriately, invoked the bootops of the 32-bit, protected mode booter to provide the service. That service would in turn often result in the protected mode booter switching the processor back to real mode to deliver it. Though our colleagues at AMD winced at the thought of the poor processor switching back and forth hundreds or thousands of times through so many years of x86 history, Opteron didn't mind this at all.

Finally, before we delivered the code to the Solaris 10 gate, William Kucharski, who took the half-thought-out prototype and made all this insanity actually work correctly, integrated the vmx code inside the 32-bit booter so this component is invisible in the final product.

9.2.4 What else could we have done?

We could've made the boot program be completely 64-bit, and thus have that program deal with the mode switching from long, to protected to real to invoke the BIOS and back again. While possible, it would've involved porting a bunch of code in the x86 boot program to LP64, and reworking both protected-mode and real-mode assembler in the boot program to fit into an ELF64 environment. The latter seemed like a lot more work than we wanted, even if we'd somehow managed to convince the assembler and linker to support it.

Another suggestion was to somehow jump into the 64-bit kernel and do magic to allow us to call back to 32-bit boot services there. But that seemed to us to be just a matter of finding a different place to put the code we had in vmx; we thought putting it into boot where it will be eventually reused by the system was better than making that ugliness live in the kernel proper.

The other option on the table was to become dependent on the newboot project which we were planning at the time to bring us into the modern world of open source kernel booting; but we were unwilling to wait, or to force that dependency to be resolved earlier because of schedule risk.

9.2.5 Descriptor Tables and Segmentation

It quickly becomes apparent to students of the x86 architecture that with x64, AMD tried hard to preserve the better parts of the x86 segmentation architecture, while trying to preserve compatibility sufficient to allow switching to and from long mode relatively painlessly. But to the kernel programmer, it only seems to get more complicated. In previous versions of Solaris, we used to build the descriptor tables (almost) statically, with a single pass over the tables to munge IDT and GDT structures, from a form easy to initialize in software, into the form expected by the hardware. Early on, we realized it was worth bringing this up-to-date too, so we discarded what we had (for both 32-bit and 64-bit Solaris) and used the FreeBSD version which use a series of function calls to build table entries piece by piece.

Some of our more amusing early mistakes here included copying various IDT entries for machine exceptions as "trap" type exceptions instead of interrupt-type exceptions which caused real havoc when an interrupt would sneak in before the initial swapgs instruction of the handler. All terribly obvious now, but less than obvious at the time.

One optimization we made was to exploit the fact that the data segment descriptor registers %ds and %es were effectively ignored in long mode. Further, %cs and %ss were always set correctly by the hardware, %fs was unused by the kernel, and %gs has a special instruction to change the base

address underlying the segment. Taken together this lead us to a scheme of lazy update of segment registers; we only update segment registers on the way out of the kernel if we know that something needs to be changed.

9.3 Solaris 10 on x64 Processors: Part 3 - Kernel

9.3.1 Virtual Memory

One of the most critical components of a 64-bit operating system is it's ability to manage large amounts of memory using the additional addressing capabilities of the hardware. The key to those capabilities in Solaris is the HAT (Hardware Address Translation) "layer" of the otherwise generic VM system. Unfortunately, the 32-bit HAT layer for Solaris x86 was a bit long in the tooth and after years of neglect was extremely difficult to understand, let alone extend. So we decided on a ground-up rewrite pretty early on in the project; the eventual benefit of that was being able to use the same source code for both 32-bit and 64-bit mode, and to bring the benefits of the NX (no-execute) bit to both 32-bit and 64-bit kernels seamlessly. Joe Bonasera, who lead this work, told me a few weeks ago that he'd expand on this in his own blog here, so I'm not going to describe it any further than that.

[*From the blog of Joe Bonasera who was responsible for the HAT layer...]*

> The HAT provides interfaces to the "common" Solaris virtual memory code that manage architectural dependent things like page tables and page mapping lists. If you're not pretty familiar with how x86 page tables look, the rest of these posts will make about as much sense as a gaggle of honking geese. A good reference is the AMD x86-64 Architecture Programmer's Manual, Volume 2 System Programming, Chapter 5 Page Translation and Protection.
>
> You'll eventually also need to know a little bit about the HAT. The major interfaces exported by a Solaris HAT are:
>
> - `hat_memload(address_space, virt_addr, phys_page, permissions, etc)` - loads a translation for the given virtual address to the given physical page for an address space.
> - `hat_devload(address_space, virt_addr, phys_addr, etc)` - similar to above but generally used for device memory.
> - `hat_memload_array(address_space, virt_addr, phys_page list, etc)` - similar to hat_memload(), but allows for multiple or large page mappings in a single call.
> - `hat_unload(address_space, virt_addr, length)` - undoes the above, ie. removes mappings from an address space

- `hat_pageunload(phys_page)` - given a physical page, remove all virtual mappings to that page from all address spaces.

The previous x86 HAT's design was rather tied up in the requirements of running in a 32 bit address space on small memory PCs that were typical of the early/mid 1990s. It contained quite a bit of special code to deal with memory allocation and address space manipulations in order to be have large amounts of page tables and mapping list data structures even though normal kernel virtual address range is limited to the top 1 Gigabyte (or so) of memory. The idea of 2 levels of page tables was pretty much hard coded into it with some slight of hand #ifdef-ing to have the partial 3rd table needed in PAE mode.

At the start of the project we planned to just extend the old HAT code in order to get us able to run in 64 bit mode as quickly as possible. The project started rather late in the release cycle for Solaris 10 and had a very tight schedule to meet. We expected to go back later and possibly rewrite much of the HAT for better 64 bit performance for Solaris 10 updates.

After a week of looking at what needed to be done, I proposed writing all new code from the start. If you're a kernel developer, your reaction to that statement should be the same as my project leaders' were at that time, that it was crazy to propose such a risky approach. But it made sense due to a new design for the HAT that made the code neutral to 32bit non-PAE vs 32 bit PAE vs 64 bit environments. The new HAT would execute almost all the same code paths in all modes. Hence, I could write and debug it in the existing stable 32 bit version of Solaris with a reasonable expectation that the code should just recompile and work in the 64 bit environment. I'd be able to start coding and testing immediately, while the rest of the amd64 team was still working on other startup issues, like 64 bit compilers and boot loaders and other tasks.

The new design idea was to encode all the parameters about the paging hierarchy (ie, page tables, page directories, page directory pointer tables and page-map Level-4 tables) into an mmu description structure. The mmu description would be filled in early at boot once Solaris determines what mode the processor will run in. The HAT then always interprets this description when manipulating page tables.

To illustrate the difference, I'll show some psuedo-code for a mythical HAT function which looks for a PTE for a given virtual

address by walking down the page table hierarchy. I've tried to use "variable" names to make the code self-explanatory. First the old code if it were extended to 64 bits in the most obvious fashion:

```
#if defined(__amd64) || defined(PAE)
typedef uint64_t pte_t;
#else
typedef uint32_t pte_t;
#endif


pte_t
hat_probe(caddr_t address)
{
    pte_t pte;
    uintptr_t va = (uintptr_t)address;
    uint_t index;
    pte_t *ptable;

    ptable = find_top_table(current_addr_space);
    ASSERT(ptable != NULL);

#if defined(__amd64)
    /*
     * 64 bit mode uses 4 levels of page tables
     * MMU_PML4_SHIFT is 39
     * MMU_PLM4_MASK is (512 - 1)
     */
    index = (va >> MMU_PML4_SHIFT) & MMU_PML4_MASK;
    pte = ptable_extract(ptable, index)
    if (pte == 0)
            return 0;
    ptable = find_PDP_table(pte & MMU_PAGEMASK);
    ASSERT(ptable != NULL);
#endif

#if defined(__amd64) || defined(PAE)
    /*
     * 3rd level of pagetables.
     * MMU_PDP_SHIFT is 30
     * MMU_PPD_MASK is either (512 - 1) or (4 - 1)
     */
    index = (va >> MMU_PDP_SHIFT) & MMU_PDP_MASK;
    pte = ptable_extract(ptable, index)
    if (pte == 0)
            return 0;
    ptable = find_PD_table(pte & MMU_PAGEMASK);
    ASSERT(ptable != NULL);
#endif

    /*
     * 2nd level page tables
     * MMU_PD_SHIFT is either 21 or 22
     * MMU_PD_MASK is either (512 - 1) or (1024 - 1)
     */
    index = (va >> MMU_PD_SHIFT) & MMU_PD_MASK;
    pte = ptable_extract(ptable, index)
    if (pte == 0)
            return 0;
    if (pte & PT_PAGESIZE_BIT)
```

```
            return pte;
    ptable = find_PT_table(pte & MMU_PAGEMASK);
    ASSERT(ptable != NULL);

    /*
     * Lowest level page table
     * MMU_PT_SHIFT is 12
     * MMU_PT_MASK is either (512 - 1) or (1024 - 1)
     */
    index = (va >> MMU_PT_SHIFT) & MMU_PT_MASK;
    pte = ptable_extract(ptable, index)
    return (pte);
}
```

Under the new scheme the same interface looks like this:

```
typedef uint64_t pte_t;
typedef void *ptable_t;
struct mmu_description {...} mmu;

pte_t
hat_probe(caddr_t address)
{
    pte_t pte;
    uintptr_t va = (uintptr_t)address;
    uint_t index;
    int level;
    ptable_t ptable;

    for (level = mmu.top_level; level >= 0; --level) {
            ptable = ptable_lookup(va, level, current_addr_space);
            ASSERT(ptable != NULL);
            index = (va >> mmu.shift[level]) &
                    mmu.index_mask[level];
            pte = ptable_extract(ptable, index)
            if (pte == 0)
                    return 0;
            if ((pte & mmu.is_page_mask[level]) != 0)
                    return pte;
    }
    return 0;
}
```

The new code has a small amount of additional looping and memory reference overhead in exchange for it's compactness and improved extensiblity. If a future processor adds additional large pagesizes or more pagetable levels the mmu description might change, but this code would just work. Another thing to note is that you would probably change the new version to use:

```
    for (level = mmu.top_pagesize_level; level >= 0; --level) {
```

as the loop boundaries to improve its performance.

The important thing at the time for the project, was that the old style 64 bit code couldn't have been tested until we had a 64 bit kernel partially working. With the new style code, we could do a lot of testing on 32 bit platforms, long before any other part of the 64 bit kernel was ready and be fairly confident that

the code was correct. In the end this proved to be a great choice as the 64 bit HAT was almost never on the critical path for code development.

One issue that any x86 Operating System has to deal with is how to manage software access to page tables. The hardware does page table lookups using physical (not virtual) addresses. However, in order for an OS to create, modify or remove page table entries it has to have the page table mapped in virtual memory.

Solaris 9 stored page tables in the "user" part of virtual address space. Whenever the kernel had to access a pagetable entry, it would change %cr3 once to switch to the page table address space and then again to get back to the original address space. One of the ramifications of changing %cr3 on x86 is that the entire contents of the TLB may be invalidated.

In Solaris 10 we take a different approach to minimize the impact of page table accesses on the TLB. The kernel maintains 4K page aligned peep holes which are remapped on demand to access pagetables. Remapping a single page requires one INVLPG instruction which can be much quicker than an entire address space change, TLB flush and subsequent TLB reloads. Solaris allocates a unique peep hole for each CPU, to avoid contention or interference between CPUs. To use the peep hole, the HAT does:

- Disable thread pre-emption, so it won't migrate to a different CPU.
- Acquire a per-peep-hole spin lock, to avoid conflicting with interrupt code
- If the peep-hole doesn't already point to the desired physical page
- Update the PTE to the peep hole to the new page.
- Issue an INVLPG for the peep hole address
- Access the desired page table entry (usually one XCHG instruction)
- Release the spin lock
- Enable pre-emption

The spin lock comes into play when an interrupt happens during a page table access and the interrupt code also has to access a pagetable. In that case the spin lock acquisition allows the interrupt thread to yield back to the interrupted, or pinned, thread to allow it to free up the peep hole. The code to disable and enable pre-emption is very quick. On Solaris it's just an

increment of a thread flag for disable and an increment/compare for enable.

When running the 64 bit kernel on a processor that has much more virtual address space than actual physical memory, this is all much easier. The kernel maintains a region of virtual address space that is mapped 1:1 to physical addresses called seg_kpm (kernel physical map). The pagetable code uses addresses in seg_kpm to access page tables instead of using the peep hole. This saves executing a lot of code and is much faster. One of the many benefits of a 64 bit operating system.

The page tables for certain special purpose parts of the kernel address space are always maintained in virtual memory. This includes the PTEs that map peep holes as well as something called segmap which is used frequently in I/O transactions.

One note for anybody looking at the source once Open Solaris hits the streets, is that the code confusingly calls the peep holes "windows" - for the purpose of this blog, the term peep hole seemed better. Maybe I'll get the code changed to match.

A final optimization to mention here is that the Solaris 32 and 64 bit kernels avoid allocating overhead pagetables for 32 bit user processes when using PAE. Since a 32 bit user process has at most 4 page table entries at level 2, the HAT stores the entries in part of the address space data structure. When a thread using that address space starts to run a CPU, the 4 entries are copied to a per-cpu set of pagetables at the start of the current level 2 page table. This saves approximately 1 page for each 32 bit process in the 32 bit kernel and 3 pages for each 32 bit process in the 64 bit kernel using a consistent mechanism.

9.3.2 Interrupts, DMA, DDI, device drivers

The Solaris DDI (Device Driver Interface) was designed to support writing portable drivers between releases, and between instruction sets, to concentrate bus-dependent details and interfaces in specialized bus-dependent drivers (called nexus drivers), and to minimize the amount of low-level, bus-specific code in regular drivers (called leaf drivers). Most of the work we did on the 64-bit SPARC project back in 1997 was completely reused, and the majority of the work on the x86 DDI implementation was essentially making the code LP64 clean, and fixing some of the more hacky internals of some of the nexus drivers.

The most difficult part of the work was porting the low-level interrupt handlers, which were a monumental mass of confusing assembler. Though I had thought that it would be simplest to port the i386 assembler to amd64 conventions, this turned out to have been a poor decision. Sherry Moore

tried to get this done quickly and accurately, but it was a very difficult challenge. We spent many days debugging problems with interrupts that were really rooted in the differences in register allocations between the two instruction set architectures and ABIs, as well as the highly contorted nature of the original code. We spent so much time on it that I eventually became consumed with guilt and rewrote most of it in C, which unsurprisingly turned out to be much easier to debug, and is now probably the best way to understand how the threads-as-interrupts implementation actually works.

The remaining work can be split into two parts. The first was ensuring that the drivers properly described their addressing capabilities, particularly those that hadn't been updated in a while. The second was the usual problem of handling ioctls from 32-bit and 64-bit applications where the two environments use different size and alignments for the data types passed across the interface. Again, Solaris already had a bunch of mechanism for doing this which we simply reused on previously i386-specific drivers to make them usable on amd64 kernels too.

One slight thorn in our side was the different in alignment constraints for the long long data type. On 32-bit SPARC and 64-bit SPARC, the alignment is 8 bytes for both, however, between i386 and amd64, the alignment changes from 4 bytes to 8 bytes. This seems mildly arcane, until you recall that the alignment of these data types controls the way that basic data structures are laid out between the two ABIs. Data structures containing long long types that were compatible between a 32-bit SPARC application and the 64-bit SPARC kernel now needed special handling for a 32-bit x86 application running on a 64-bit amd64 kernel. The same problem was discovered in a few network routing interfaces, cachefs, priocntl etc. Once we'd debugged a couple of these by hand, Ethan Solomita started a more systematic effort to locate the remaining problems; Mike Shapiro suggested that we build a CTF tool that would help us find the rest more automatically, or at least semi-automatically, which was an excellent idea and helped enormously.

9.3.3 MP bringup, EM64-T bringup

Back in 1990, one of the core design goals of the SunOS 5.0 project was to build a multithreaded operating system designed to run on multiprocessor machines. We weren't just doing a simple port of SVR4 to SPARC, we reworked the scheduler, and invested a large amount of effort throughout the kernel, adding fine-grain locking to extract the maximal concurrency from the hardware. Fast forward to 2005, and we're still working on it! The effort to extend scalability remains one of our core activities. However, we didn't have to do a lot of work to make multiprocessor Opteron machines run the 64-bit kernel; apart from porting the locking primitives, the only porting work was around creating a primitive environment around the non-boot processors to switch them into long mode. William Kucharski (of

amd64 booter fame) did this work in a week or so, and impressed us all with how quickly and how well this worked from the beginning.

We also wanted to run our 64-bit kernel on Intel's EM64-T CPUs, since we really do want Solaris to run well on non-Sun x86 and x64 systems. As we were doing other work on the system, we had been anticipating what we needed to do from Intel's documentation, so as soon as the hardware was publically available (unfortunately we weren't able to get them earlier from Intel) Russ Blaine started working on it and had the 64-bit kernel up and running multiuser in about a week. I'm not sure if that's because Intel's specifications are particularly well written, or because Russ's debugging skills were even more excellent that week, or if it's testament to the skills of the Intel engineers at making their processor be so compatible with the Opteron architecture, but we were pretty pleased with the result.

9.3.4 Debugging Infrastructure

Critical aspects of the debugging architecture of Solaris that needed to be ported include the CTF system for embedding dense type information in ELF files, and the corresponding library and toolchain infrastructure that manipulates it, libproc that encapsulates a bunch of /proc operations for the ptools, /proc itself, mdb, and the DTrace infrastructure. I worked on the easy part - /proc - the difficult work was done by Matt Simmons, Eric Schrock and for DTrace, Adam Leventhal and of course Bryan Cantrill.

At the same time as we were starting our bring-up efforts on Opteron, an unrelated project in the kernel group was busy creating a new debugging architecture based on mdb(1). The basic idea was that we wanted to be able to bring most of mdb's capabilities to debugging live kernel problems. The kmdb team observed that our existing kernel debugger, kadb, was always in a state of disrepair, and yet because of it's co-residence with the kernel, needs constant tweaking for new platforms. So rather than continue this state of affairs, they came to the idea that it would be simpler if we could assume that the Solaris kernel would provide the basic infrastructure for the debugger.

This has considerable advantages for incremental development, and for the vast majority of kernel developers who aren't working on new platform bringup this is clearly a Good Thing. But it does make porting to a fresh platform or instruction set a little more difficult because kmdb is sophisticated, and doesn't really work until some of the more difficult kernel code has been debugged into existence. The amd64 project had that problem in a particularly extreme form, because the debugger design and interfaces were under development at the same time as we needed them. As a result, the early amd64 kernel bringup work was really done using a simulator (SIMICS), and then by doing printf-style debugging, and post-mortem trap-tracing, than with kmdb. I still remember debugging init(1M)

using the simulator on the last day of one of our offsites in San Francisco, figuring out the bug while riding BART back home.

At this point of course, kmdb works fine and is of great help when debugging more subtle problems. However, knowing what we know now, we should have built a simple bringup-debugger to get us through those early stages where almost nothing worked. Something that could catch and decode exceptions, do stack traces and dump memory would be enough. I'd certainly recommend that path to anyone thinking of porting Solaris to another instruction set architecture; as soon as you get to the point that the kernel starts taking interrupts and doing context switches, things get way too hard for printf-style debugging!

9.3.5 System calls Revisited

For 64-bit applications we used the syscall instruction. We used the same register calling conventions as Linux; these are somewhat forced upon you by the combination of the behaviour of the instruction, and the C calling convention, and besides, there is no value in being deliberately different.

Interestingly, the 64-bit system call parameter passing convention is extremely similar to SPARC i.e. the first six system call arguments are passed in registers, with additional arguments passed on the stack. As a result, we based the 64-bit system call handler algorithm for amd64 on the 64-bit handler for sparcv9.

The 32-bit system call handlers include the 32-bit variant of the syscall instruction which works sufficiently well when the processor is running the 64-bit kernel to be usable. We also made the sysenter instruction work for Intel CPUs, and of course, the lcall handler; though this is actually handled via a #np trap in C. Our latest version of this assigns a new int trap to 32-bit syscalls which will improve the performance of the various types of system call that don't work well with plain syscall or sysenter.

9.3.6 More Tool Chain Issues

In the earlier "preliminaries" blog, I mentioned our use of gcc; however the Solaris kernel contains its own linker, krtld, based on the same relocation engine used in the userland utility. Fortunately, we had Mike Walker to do the amd64 linker work early on; we had a working linker a week or two ahead of having a linkable kernel.

9.4 Solaris 10 on x64 Processors: Part 4 - Userland

9.4.1 Userland

The amount of work involved in the kernel part of the amd64 project was fairly large, fortunately the userland part was more straightforward because of our prior work on 64-bit Solaris on SPARC back in 1997. So, for

this project, once the kernel work, which abstracts the hardware differences between processors, was done, many smaller tasks appeared that were mostly solved by tweaking Makefiles and finding occasional #ifdefs that needed something added or modified. Fortunately, it was also work that was done in parallel by many people from across the organizations that contribute to the Solaris product.

Of course there were other substantial pieces of work like the Sun C and C++ compilers, and the Java Virtual Machine; though the JVM was already working on 32-bit and 64-bit Solaris on SPARC as well as 32-bit on x86, and the Linux port of the JVM had already caused that team to explore many of the amd64 code generation issues.

One of the things we tried to do was to be compatible with the amd64 ABI on Linux. As we talked to industry partners, we discovered that there was a variety of interpretations of the term "ABI." Many of the people we talked to outside of Sun thought that "ABI" only referred to register usage, C calling conventions, data structure sizes and alignments. A specification for compiler and linker writers, but with little or nothing beyond that about the system interfaces an application can actually invoke. But, the System V ABI is a larger concept than that, and was at least intended to provide a sufficient set of binary specifications to allow complete application binaries to be constructed that could be built once, and run on any ABI-conformant implementation. Thus Sun engineers tend to think of "the ABI" as being the complete set of interfaces used by user applications, rather than just compiler conventions; and over the years we expanded this idea of maintaining a binary compatible interface to applications all the way to the Solaris application guarantee program.

Though we tried to be compatible at this level with Linux on amd64, we discovered a number of issues in the system call and library interfaces that made that difficult, and while we did eliminate gratuitous differences where we could, we eventually decided on a more pragmatic approach. We decided to be completely compatible with the basic "compiler" style view of the ABI, and simply try and make it simple to port applications from 32-bit Solaris to 64-bit Solaris, and from Solaris on sparcv9 to Solaris on x64, and leave the thornier problems of full 64-bit Linux application compatibility to the Linux Application Environment (LAE) project.

9.4.2 Threads and Selectors

In previous releases of Solaris, the 32-bit threads library used the %gs selector to allow each LWP in a process to refer to a private LDT entry to provide the per-thread state manipulated by the internals of the thread library. Each LWP gets a different %gs value that selects a different LDT entry; each LDT entry is initialized to point at per-thread state. On LWP context switch, the kernel loads the per-process LDT register to virtualize all this data to the process. Workable, yes, but the obvious inefficiency here

was requiring every process to have at least one extra locked-down page to contain a minimal LDT. More serious, was the implied upper bound of 8192 LWPs per process (derived from the hardware limit on LDT entries).

For the amd64 port, following the draft ABI document, we needed to use the %fs selector for the analogous purpose in 64-bit processes too. On the 64-bit kernel, we wanted to use the FSBASE and GSBASE MSRs to virtualize the addresses that a specific magic %fs and magic %gs select, and we obviously wanted to use a similar technique on 32-bit applications, and on the 32-bit kernel too. We did this by defining specific %fs and %gs values that point into the GDT, and arranged that context switches update the corresponding underlying base address from predefined lwp-private values - either explicitly by rewriting the relevant GDT entries on the 32-bit kernel, or implicitly via the FSBASE and GSBASE MSRs on the 64-bit kernel. The result of all this work makes the code simpler, it scales cleanly, and the resulting upper bound on the number of LWPs is derived only from available memory (modulo resource controls, obviously).

9.4.3 Floating point

Most of the prework we had done to establish the SSE capabilities in the 32-bit kernel was readily reused for amd64; modulo some restructuring to allow the same code to be compiled appropriately for the two kernel builds. However, late in the development cycle, the guys in our floating point group pointed out that we didn't capture the results of floating point exceptions properly; the result of a subtle difference in the way that AMD and Intel processors presented information to the kernel after the floating point exception had been acknowledged. Fortunately they noticed this, and we rewrote the handler to be more robust and to behave the same way on both flavors of hardware.

9.4.4 Continuous Integration vs. One Giant Putback

To try to keep our merging and synchronization efforts under control, we did our best to integrate many of the changes we were making directly into the Solaris 10 gate so that the rest of the Solaris development organization could see it. This wasn't a willy-nilly integration of modified files, instead each putback was a regression-tested subset of the amd64 project that could stand alone if necessary. Perhaps I should explain this a little further. The Solaris organization has, for many years, tried to adhere to the principle of integrating complete projects, that is, changes that can stand alone, even if the follow-on projects are cancelled, fail, or become too delayed to make the release under development. Some of the code reorganization we needed was done this way, as well as most of the items I described as "prework" in part 1. There were also a bunch of code removal projects we did that helped us avoid the work of porting obsolete subsystems and support for drivers. As an aside, it's interesting to muse on exactly who is responsible

to get rid of drivers for obsolete hardware; it's a very unglamourous task, but one that it's highly necessary if you aren't to flounder under and ever more opaque and untestable collection of crufty old source code.

In the end though, we got to the point where the pain of creating and testing subsets of our change by hand to create partial projects in Solaris 10 became just too painful for the team to countenance. Instead, we focussed on creating a single delivery of all our change in one coherent whole. Our Michigan-based "army of one," Roger Faulkner did all of this, as well as most of the rest of the heavy lifting in userland i.e. creating the 64-bit libc and basic C run-time etc. as well as the threading primitives. Roger really did an amazing job on the project.

Projects of this giant size and scope are always difficult; and everyone gets even more worried when the changes are integrated towards the end of a release. However, we did bring unprecedented levels of testing to the amd64 project, from some incredible, hard working test people. Practically speaking I think we did a reasonable job of getting things right by the end of the release, despite a few last minute scares around our mishandling of process-private LDTs. Fortunately these were only really needed for various forms of Windows emulation, so we disabled them on the 64-bit kernel for the FCS product; this works now in the Solaris development gate, and a backported fix is working its way through the system.

Not to say that there aren't bugs of course ...

9.4.5 Distributed Development

I think it's worth sharing some of the experiences of how the core team worked on this project. First, when we started, Todd Clayton (the engineering lead, who also did the segmentation work, among other things) and I asked to build a mostly-local team. We asked for that because we believed that time-to-market was critical, and we thought that we could go the fastest with all the key contributors in close proximity. However, for a number of reasons, that was not possible, and we ended up instead with a collection of talented people spread over many sites as geographically distributed as New Zealand, Germany, Boston, Michigan, and Colorado as well a small majority of the team back in California. To help unify the team and make rapid progress, we came up with the idea of periodically getting the team together physically in one place (either offsite in California or Colorado) and spending a focussed week together. We spent the first week occupying a contiguous block of adjacent offices in another building; problem was that we didn't really change the dynamics of the way people worked with each other. Our accidental discovery came during our first Colorado meeting where we ended up in one (large!) training room for our kick-off meeting. Rather than trudge back across campus where we had reserved office space, we decided to stay put and just start work where we were, and suddenly everything clicked. We stayed in the room for the rest

of the week, working closely with each other, immersing ourselves in the project, the team, and what needed to be done. This was very effective, because as well as reinforcing the sense of team during the week away, everyone was able to go back to their home sites and work independently and effectively for many weeks before meeting up again - with only an occasional phone call or email between team-members to synchronize.

9.4.6 Looking Back

I've tried to do a reasonable tour of the amd64 project, driven mostly by what stuck in my memory, and biassed by the work I was involved in to some degree, but obviously much detail has been omitted or completely forgotten. To the people at Sun whose work or contribution I've either not mentioned, foolishly glossed over or forgotten completely, sorry, and thanks for your efforts. To the people at AMD that helped support us, another thank you. To our families and loved ones that put up with "one more make," yet more thanks. This was a lot of work, done faster than any of us thought possible, and 2004 was in truth, well, a bit of a blur.

10. Booting the System

The make process in the $SRC/uts subdirectory results in the creation of several objects:

- The UNIX kernel – s390x/unix/debug64/unix
- The kernel support module – zSeries/genunix/debug64/genunix
- Device drivers and other loadable modules – throughout the s390x and zSeries subdirectories
- A RAM disk

10.1 RAMDISK

The RAM disk is used as the medium for the boot process. This is where the configuration files and loadable modules reside.

10.1.1 Creating a RAMDISK

During the kernel build process the Makefile recipe will try and build a RAMDISK image that will be used during the boot process. The make process executes the following script call bldram. If you need to add modules or other files to the RAMDISK image the script will need updating.

This script will populate the RAMDISK which can then be transferred to the boot machine. The source for the script may be found at “16.1 bldram” on page 263.

10.2 Transferring the Files

Once the RAMDISK and the kernel have been built the files need to be transferred to the virtual machine on the z/VM host that will use them. This is usually done via FTP. The source for this script may be found at “16.2 upload” on page 266.

10.3 Loading the Kernel

A script on the z/VM guest is used to take the two objects and boot the OpenSolaris system. The source for this script may be found at

11. GNU Binutils Changes

The GNU binutils consist of the assembler, linker and other related tools.

11.1 Source Code Changes

Changes were required to support a new target called “ibm-s390x-solaris2”. It uses the same ABI etc. as used by “ibm-s390x-linux”.

11.1.1 binutils/ChangeLog

```
--- ./binutils/ChangeLog   2006/12/28 20:53:09     1.1
+++ ./binutils/ChangeLog   2006/12/28 20:53:25
@@ -1,5 +1,9 @@
+2006-12-28 Neale Ferguson <neale@sinenomine.net>
+
+  * : s390x-solaris2 support
+
  2006-10-30  Paul Brook  <paul@codesourcery.com>

    * objdump.c (disassemble_section): Set info->symtab_pos.
    (disassemble_data): Set info->symtab and info->symtab_size.
```

11.1.2 ld/configure.host

```
--- ./ld/configure.host   2006/12/28 21:03:12     1.1
+++ ./ld/configure.host   2006/12/28 21:03:36
@@ -203,10 +203,15 @@

 s390-*-linux-*)
    HOSTING_CRT0=`echo "$HOSTING_CRT0" | sed -e
"s,\\\`specs.*\"\\\`,/lib/ld.so.1,"`
    ;;

+s390x-*-solaris2)
+  HOSTING_CRT0='`if [ -f ../gcc/crt1.o ]; then echo ../gcc/crt1.o; else ${CC}
-print-file-name=crt1.o; fi` `if [ -f ../gcc/crti.o ]; then echo ../gcc/crti.o;
else ${CC} -print-file-name=crti.o; fi` /usr/ccs/lib/values-Xa.o `if [ -f
../gcc/crtbegin.o ]; then echo ../gcc/crtbegin.o; else ${CC} -print-file-
name=crtbegin.o; fi`'
+  HOSTING_LIBS="$HOSTING_LIBS"' `if [ -f ../gcc/crtend.o ]; then echo
../gcc/crtend.o; else ${CC} -print-file-name=crtend.o; fi` `if [ -f
../gcc/crtn.o ]; then echo ../gcc/crtn.o; else ${CC} -print-file-name=crtn.o;
fi`'
+  ;;
+
 sparc-*-solaris2*)
    HOSTING_CRT0='`if [ -f ../gcc/crt1.o ]; then echo ../gcc/crt1.o; else ${CC}
-print-file-name=crt1.o; fi` `if [ -f ../gcc/crti.o ]; then echo ../gcc/crti.o;
else ${CC} -print-file-name=crti.o; fi` /usr/ccs/lib/values-Xa.o `if [ -f
../gcc/crtbegin.o ]; then echo ../gcc/crtbegin.o; else ${CC} -print-file-
name=crtbegin.o; fi`'
    HOSTING_LIBS="$HOSTING_LIBS"' `if [ -f ../gcc/crtend.o ]; then echo
../gcc/crtend.o; else ${CC} -print-file-name=crtend.o; fi` `if [ -f
```

```
../gcc/crtn.o ]; then echo ../gcc/crtn.o; else ${CC} -print-file-name=crtn.o;
fi`'
   ;;
```

11.1.3 ld/configure.tgt

```
--- ./ld/configure.tgt    2006/12/28 21:04:12     1.1
+++ ./ld/configure.tgt    2006/12/28 21:04:33
@@ -458,10 +458,12 @@
                           targ_extra_emuls=elf_s390
                           targ_extra_libpath=$targ_extra_emuls
                           tdir_elf_s390=`echo ${targ_alias} | sed -e
's/s390x/s390/'` ;;
 s390x-*-tpf*)             targ_emul=elf64_s390
                           tdir_elf_s390=`echo ${targ_alias} | sed -e
's/s390x/s390/'` ;;
+s390x-*-solaris2*)       targ_emul=elf64_s390
+                          tdir_elf_s390=`echo ${targ_alias} | sed -e
's/s390x/s390/'` ;;
 s390-*-linux*)           targ_emul=elf_s390
                           if test "${want64}" = "true"; then
                             targ_extra_emuls=elf64_s390
                             targ_extra_libpath=$targ_extra_emuls
                             tdir_elf64_s390=`echo ${targ_alias} | sed -e
's/s390/s390x/'`
```

11.1.4 ld/ChangeLog

```
--- ./ld/ChangeLog        2006/12/28 21:04:46     1.1
+++ ./ld/ChangeLog        2006/12/28 21:05:51
@@ -1,5 +1,9 @@
+2006-12-28  Neale Ferguson <neale@sinenomine.net>
+
+        * configure.tgt, configure.host : s390x-ibm-solaris2 support
+
 2006-11-27  Ian Lance Taylor  <ian@airs.com>

         * Makefile.am (EXTRA_DIST): Put spu_ovl.o in the emultempl
         subdirectory.
         * Makefile.in: Regenerate.
```

11.1.5 gas/ChangeLog

```
--- ./gas/ChangeLog       2006/12/28 20:56:12     1.1
+++ ./gas/ChangeLog       2006/12/28 21:00:08
@@ -1,5 +1,9 @@
+2006-12-28  Neale Ferguson <neale@sinenomine.net>
+
+        * configure.tgt: add s390x-solaris2 support
+
 2006-11-29  Paul Brook  <paul@codesourcery.com>

         * config/tc-arm.c (arm_is_eabi): New function.
         * config/tc-arm.h (arm_is_eabi): New prototype.
         (THUMB_IS_FUNC): Use ELF function type for EABI objects.
```

11.1.6 gas/configure.tgt

```
--- ./gas/configure.tgt   2006/12/28 20:57:32     1.1
+++ ./gas/configure.tgt   2006/12/28 20:59:02
```

```
@@ -314,10 +314,11 @@
   ppc-*-nto*)                           fmt=elf ;;
   ppc-*-kaos*)                          fmt=elf ;;
   ppc-*-lynxos*)                fmt=elf em=lynx ;;

   s390-*-linux-*)                       fmt=elf em=linux ;;
+  s390-*-solaris*)                      fmt=elf em=linux ;;
   s390-*-tpf*)                          fmt=elf ;;

   score-*-elf)                         fmt=elf ;;

   sh*-*-linux*)                         fmt=elf em=linux
```

11.1.7 bfd/elf.c

```
--- ./bfd/elf.c   2006/12/28 22:35:29     1.1
+++ ./bfd/elf.c   2007/01/12 17:36:34
@@ -4820,12 +4820,16 @@
 {
   struct elf_obj_tdata *tdata = elf_tdata (abfd);
   Elf_Internal_Ehdr *i_ehdrp = elf_elfheader (abfd);
   file_ptr off;
   const struct elf_backend_data *bed = get_elf_backend_data (abfd);
+  int genFlag;
+
+  genFlag = (link_info != NULL ? link_info->gen_program_headers : 0);

   if ((abfd->flags & (EXEC_P | DYNAMIC)) == 0
+      && !genFlag
       && bfd_get_format (abfd) != bfd_core)
     {
       Elf_Internal_Shdr ** const i_shdrpp = elf_elfsections (abfd);
       unsigned int num_sec = elf_numsections (abfd);
       Elf_Internal_Shdr **hdrpp;
```

11.1.8 bfd/config.bfd

```
--- ./bfd/config.bfd      2006/12/28 21:00:25     1.1
+++ ./bfd/config.bfd      2006/12/28 21:00:53
@@ -1130,10 +1130,15 @@
   s390x-*-linux*)
     targ_defvec=bfd_elf64_s390_vec
     targ_selvecs=bfd_elf32_s390_vec
     want64=true
     ;;
+  s390x-*-solaris*)
+    targ_defvec=bfd_elf64_s390_vec
+    targ_selvecs=bfd_elf32_s390_vec
+    want64=true
+    ;;
   s390x-*-tpf*)
     targ_defvec=bfd_elf64_s390_vec
     want64=true
     ;;
 #endif
```

11.1.9 bfd/ChangeLog

```
--- ./bfd/ChangeLog       2006/12/28 21:01:16     1.1
+++ ./bfd/ChangeLog       2006/12/28 21:01:59
@@ -1,5 +1,9 @@
```

```
+2006-12-28  Neale Ferguson <neale@sinenomine.net>
+
+        * config.bfd: s390x-ibm-solaris2 support.
+
 2006-12-01  H.J. Lu  <hongjiu.lu@intel.com>

         PR binutils/3609
         * elf.c (rewrite_elf_program_header): Preserve segment physical
         address in the output segment if the first section in the
```

11.1.10 binutils.spec.in

```
--- ./binutils.spec.in    2006/12/28 21:07:08     1.1
+++ ./binutils.spec.in    2006/12/28 21:07:24
@@ -50,11 +50,11 @@
 echo "MAKE=make -j $NRPROC" > makefile
 echo "include Makefile" >> makefile
 ADDITIONAL_TARGETS=""
 %ifos linux
 %if %{all_targets}
-ADDITIONAL_TARGETS="--enable-targets=alpha-linux,arm-linux,cris-linux,hppa-
linux,i386-linux,x86_64-linux,ia64-linux,m68k-linux,mips-linux,mips64-
linux,mips64el-linux,mipsel-linux,ppc-linux,ppc64-linux,s390-linux,s390x-
linux,sh-linux,sparc-linux,sparc64-linux,i386-linuxaout"
+ADDITIONAL_TARGETS="--enable-targets=alpha-linux,arm-linux,cris-linux,hppa-
linux,i386-linux,x86_64-linux,ia64-linux,m68k-linux,mips-linux,mips64-
linux,mips64el-linux,mipsel-linux,ppc-linux,ppc64-linux,s390-linux,s390x-
linux,sh-linux,sparc-linux,sparc64-linux,i386-linuxaout,s390x-solaris2"
 %else
 %ifarch %{ix86}
 ADDITIONAL_TARGETS="--enable-targets=i386-linuxaout,i386-%{coff_target}"
 %endif
 %ifarch ia64
```

11.1.11 config.guess

```
--- ./config.guess        2006/12/28 21:07:51     1.1
+++ ./config.guess        2006/12/28 21:08:35
@@ -933,10 +933,13 @@
         echo hppa64-unknown-linux-gnu
         exit ;;
     s390:Linux:*:* | s390x:Linux:*:*)
         echo ${UNAME_MACHINE}-ibm-linux
         exit ;;
+    s390x:Solaris2:*:*)
+        echo ${UNAME_MACHINE}-ibm-solaris2`echo ${UNAME_RELEASE}|sed -e
's/[^.]*//'`
+        exit 0 ;;
     sh64*:Linux:*:*)
         echo ${UNAME_MACHINE}-unknown-linux-gnu
         exit ;;
     sh*:Linux:*:*)
         echo ${UNAME_MACHINE}-unknown-linux-gnu
```

12. GCC Changes

The initial work on GCC was done on the 4.0.2 level and then brought forward to 4.1.1. then to 4.3.1. A new target ibm-s390x-solaris2 was created. It uses the same ABI as the ibm-s390x-linux target so very little code had to be changed or added.

12.1 Source Code Changes

This section describes the changes made to the GCC tree.

12.1.1 configure

```
===================================================================
--- configure   (revision 139673)
+++ configure   (working copy)
@@ -5892,7 +5892,7 @@
 # being built; programs in there won't even run.
 if test "${build}" = "${host}" && test -d ${srcdir}/gcc; then
   # Search for pre-installed headers if nothing else fits.
-  FLAGS_FOR_TARGET=$FLAGS_FOR_TARGET' -B$(build_tooldir)/bin/ -
B$(build_tooldir)/lib/ -isystem $(build_tooldir)/include -isystem
$(build_tooldir)/sys-include'
+  FLAGS_FOR_TARGET=$FLAGS_FOR_TARGET' -B$(build_tooldir)/bin/ -
B$(build_tooldir)/lib/ -B$(ROOT)/lib/s390x/ -isystem $(ROOT)/usr/include -
isystem $(build_tooldir)/include -isystem $(build_tooldir)/sys-include'
 fi

 if test "x${use_gnu_ld}" = x &&
```

12.1.2 Makefile.in

```
===================================================================
--- Makefile.in (revision 139673)
+++ Makefile.in (working copy)
@@ -353,8 +353,10 @@
 # variable is passed down to the gcc Makefile, where it is used to
 # build libgcc2.a.  We define it here so that it can itself be
 # overridden on the command line.
-GCC_FOR_TARGET=$(STAGE_CC_WRAPPER) @GCC_FOR_TARGET@ $(FLAGS_FOR_TARGET)
-CXX_FOR_TARGET=$(STAGE_CC_WRAPPER) @CXX_FOR_TARGET@ $(FLAGS_FOR_TARGET)
+GCC_FOR_TARGET=$(STAGE_CC_WRAPPER) @GCC_FOR_TARGET@ $(FLAGS_FOR_TARGET)
\
+               -L$(HOME)/OpenSolaris/ibm/onnv-gate/proto/root_s390/lib
+CXX_FOR_TARGET=$(STAGE_CC_WRAPPER) @CXX_FOR_TARGET@ $(FLAGS_FOR_TARGET)
\
+               -L$(HOME)/OpenSolaris/ibm/onnv-gate/proto/root_s390/lib
 RAW_CXX_FOR_TARGET=$(STAGE_CC_WRAPPER) @RAW_CXX_FOR_TARGET@
$(FLAGS_FOR_TARGET)
 GCJ_FOR_TARGET=$(STAGE_CC_WRAPPER) @GCJ_FOR_TARGET@ $(FLAGS_FOR_TARGET)
 GFORTRAN_FOR_TARGET=$(STAGE_CC_WRAPPER) @GFORTRAN_FOR_TARGET@
$(FLAGS_FOR_TARGET)
```

12.1.3 libtool.m4

```
===================================================================
--- libtool.m4  (revision 139673)
+++ libtool.m4  (working copy)
@@ -1133,7 +1133,7 @@
   ;;

 x86_64-*kfreebsd*-gnu|x86_64-*linux*|ppc*-*linux*|powerpc*-*linux*| \
-s390*-*linux*|s390*-*tpf*|sparc*-*linux*)
+s390*-*linux*|s390*-*tpf*|s390*-solaris2*|sparc*-*linux*)
   # Find out which ABI we are using.
   echo 'int i;' > conftest.$ac_ext
   if AC_TRY_EVAL(ac_compile); then
@@ -1152,6 +1152,9 @@
           s390x-*linux*)
             LD="${LD-ld} -m elf_s390"
             ;;
+          s390x-*solaris2*)
+            LD="${LD-ld} -m elf_s390"
+            ;;
           sparc64-*linux*)
             LD="${LD-ld} -m elf32_sparc"
             ;;
@@ -1168,7 +1171,7 @@
           ppc*-*linux*|powerpc*-*linux*)
             LD="${LD-ld} -m elf64ppc"
             ;;
-          s390*-*linux*|s390*-*tpf*)
+          s390*-*linux*|s390*-*tpf*|s390*-*solaris2*)
             LD="${LD-ld} -m elf64_s390"
             ;;
           sparc*-*linux*)
```

12.1.4 libgcc/config.host

```
===================================================================
--- libgcc/config.host  (revision 139673)
+++ libgcc/config.host  (working copy)
@@ -545,6 +545,10 @@
        ;;
 s390x-*-linux*)
        ;;
+s390-*-solaris2*)
+       ;;
+s390x-*-solaris2*)
+       ;;
 s390x-ibm-tpf*)
        ;;
 score-*-elf)
```

12.1.5 libgcc/configure

```
===================================================================
--- libgcc/configure    (revision 139673)
+++ libgcc/configure    (working copy)
@@ -1625,8 +1625,10 @@
 # --srcdir=. covers the toplevel, while "test -d" covers the subdirectories
 if ( test $srcdir = . && test -d gcc ) \
    || test -d $srcdir/../host-${host_noncanonical}; then
+echo "XXXXXXXXXXXXXXX host noncan (from $srcdir from `pwd`)"
```

```
   host_subdir="host-${host_noncanonical}"
 else
+echo "XXXXXXXXXXXXXXX host . (from $srcdir from `pwd`)"
   host_subdir=.
 fi
 # No prefix.
```

12.1.6 config.guess

```
===================================================================
--- config.guess        (revision 139673)
+++ config.guess        (working copy)
@@ -949,6 +949,9 @@
     s390:Linux:*:* | s390x:Linux:*:*)
        echo ${UNAME_MACHINE}-ibm-linux
        exit ;;
+    s390:SunOS:5.*:* | s390x:SunOS:5.*:* )
+       echo ${UNAME_MACHINE}-ibm-solaris2`echo ${UNAME_RELEASE}|sed -e
's/[^.]*//'`
+       exit 0 ;;
     sh64*:Linux:*:*)
        echo ${UNAME_MACHINE}-unknown-linux-gnu
        exit ;;
```

12.1.7 gcc/dwarf2out.c

```
===================================================================
--- gcc/dwarf2out.c     (revision 139673)
+++ gcc/dwarf2out.c     (working copy)
@@ -7344,11 +7344,14 @@
          break;

        case dw_val_class_str:
-         if (AT_string_form (a) == DW_FORM_strp)
+         if (AT_string_form (a) == DW_FORM_strp) {
+            char label[256];
+            snprintf(label, 256, "%s-.debug_str",a->dw_attr_val.v.val_str-
>label);
            dw2_asm_output_offset (DWARF_OFFSET_SIZE,
-                                  a->dw_attr_val.v.val_str->label,
+                                  label,
                                   debug_str_section,
                                   "%s: \"%s\"", name, AT_string (a));
+          }
          else
            dw2_asm_output_nstring (AT_string (a), -1, "%s", name);
          break;
```

12.1.8 gcc/config.gcc

```
===================================================================
--- gcc/config.gcc      (revision 139673)
+++ gcc/config.gcc      (working copy)
@@ -2131,6 +2131,17 @@
        out_file=s390/s390.c
        tmake_file="${tmake_file} t-dfprules s390/t-crtstuff s390/t-linux
s390/t-linux64"
        ;;
+s390*-*-solaris2*)
+       tm_file="s390/s390x.h s390/s390.h dbxelf.h elfos.h svr4.h sol2.h
s390/sol2.h"
```

```
+       tm_p_file="s390/s390-protos.h sol2-protos.h"
+       md_file=s390/s390.md
+       extra_modes=s390/s390-modes.def
+       out_file=s390/s390.c
+       c_target_objs="sol2-c.o"
+       cxx_target_objs="sol2-c.o"
+       extra_objs="sol2.o"
+       tmake_file="${tmake_file} t-slibgcc-elf-ver t-sol2 s390/t-crtstuff
s390/t-sol2-64"
+       ;;
 s390x-ibm-tpf*)
         tm_file="s390/s390x.h s390/s390.h dbxelf.h elfos.h svr4.h s390/tpf.h"
         tm_p_file=s390/s390-protos.h
```

12.1.9 gcc/Makefile.in

```
===================================================================
--- gcc/Makefile.in     (revision 139673)
+++ gcc/Makefile.in     (working copy)
@@ -315,7 +315,7 @@
 # The GCC to use for compiling crt*.o.
 # Usually the one we just built.
 # Don't use this as a dependency--use $(GCC_PASSES).
-GCC_FOR_TARGET = $(STAGE_CC_WRAPPER) ./xgcc -B./ -B$(build_tooldir)/bin/ -
isystem $(build_tooldir)/include -isystem $(build_tooldir)/sys-include -
L$(objdir)/../ld
+GCC_FOR_TARGET = $(STAGE_CC_WRAPPER) ./xgcc -B./ -B$(build_tooldir)/bin/ -
isystem $(build_tooldir)/include -isystem $(build_tooldir)/sys-include -
L$(objdir)/../ld -L$(HOME)/OpenSolaris/ibm/onnv-gate/proto/root_s390/lib/s390x

 # This is used instead of ALL_CFLAGS when compiling with GCC_FOR_TARGET.
 # It omits XCFLAGS, and specifies -B./.
@@ -388,8 +388,9 @@
 # The sed idiom for this is to repeat the search-and-replace until it doesn't
match, using :a ... ta.
 # Use single quotes here to avoid nested double- and backquotes, this
 # macro is also used in a double-quoted context.
-SYSTEM_HEADER_DIR = `echo @SYSTEM_HEADER_DIR@ | sed -e :a -e
's,[^/]*/\.\.\/,,' -e ta`
+SYSTEM_HEADER_DIR = $(shell echo @SYSTEM_HEADER_DIR@ | sed -e :a -e
"s,[^/]*/\.\.\/,," -e ta)

+
 # Control whether to run fixproto and fixincludes.
 STMP_FIXPROTO = @STMP_FIXPROTO@
 STMP_FIXINC = @STMP_FIXINC@
```

12.1.10 gcc/config/s390/s390.c

```
===================================================================
--- gcc/config/s390/s390.c      (revision 139673)
+++ gcc/config/s390/s390.c      (working copy)
@@ -9416,6 +9416,32 @@
 #undef TARGET_LIBGCC_SHIFT_COUNT_MODE
 #define TARGET_LIBGCC_SHIFT_COUNT_MODE s390_libgcc_shift_count_mode

+#define SUBTARGET_INSERT_ATTRIBUTES solaris_insert_attributes
+
+#ifdef SUBTARGET_INSERT_ATTRIBUTES
+#undef TARGET_INSERT_ATTRIBUTES
+#define TARGET_INSERT_ATTRIBUTES SUBTARGET_INSERT_ATTRIBUTES
```

```
+#endif
+
+#ifdef SUBTARGET_ATTRIBUTE_TABLE
+const struct attribute_spec s390_attribute_table[];
+#endif
+
+#ifdef SUBTARGET_ATTRIBUTE_TABLE
+/* Table of valid machine attributes.  */
+const struct attribute_spec s390_attribute_table[] =
+{
+  /* { name, min_len, max_len, decl_req, type_req, fn_type_req, handler } */
+  SUBTARGET_ATTRIBUTE_TABLE,
+  { NULL,        0, 0, false, false, false, NULL }
+};
+#endif
+
+#ifdef SUBTARGET_ATTRIBUTE_TABLE
+#undef TARGET_ATTRIBUTE_TABLE
+#define TARGET_ATTRIBUTE_TABLE s390_attribute_table
+#endif
+
 struct gcc_target targetm = TARGET_INITIALIZER;

 #include "gt-s390.h"
```

12.1.11 gcc/config/s390/s390.h

```
===================================================================
--- gcc/config/s390/s390.h      (revision 139673)
+++ gcc/config/s390/s390.h      (working copy)
@@ -94,9 +94,12 @@
     {                                                   \
       builtin_assert ("cpu=s390");                      \
       builtin_assert ("machine=s390");                  \
+      builtin_define ("__s390");                        \
       builtin_define ("__s390__");                      \
-      if (TARGET_64BIT)                                         \
+      if (TARGET_64BIT)         {                               \
+       builtin_define ("__s390x");                      \
         builtin_define ("__s390x__");                   \
+      }                                                 \
       if (TARGET_LONG_DOUBLE_128)                       \
         builtin_define ("__LONG_DOUBLE_128__");                 \
     }                                                   \
```

12.1.12 gcc/config/s390/t-crtstuff

```
===================================================================
--- gcc/config/s390/t-crtstuff  (revision 139673)
+++ gcc/config/s390/t-crtstuff  (working copy)
@@ -1,5 +1,5 @@
 # crtend*.o cannot be compiled without -fno-asynchronous-unwind-tables,
 # because then __FRAME_END__ might not be the last thing in .eh_frame
 # section.
-CRTSTUFF_T_CFLAGS = -fno-asynchronous-unwind-tables
-TARGET_LIBGCC2_CFLAGS += -mlong-double-128
+CRTSTUFF_T_CFLAGS = -fno-asynchronous-unwind-tables -fPIC
+TARGET_LIBGCC2_CFLAGS += -mlong-double-128 -fPIC
```

12.1.13 gcc/config/s390/crti.s

```
===================================================================
--- gcc/config/s390/crti.s      (revision 0)
+++ gcc/config/s390/crti.s      (revision 0)
@@ -0,0 +1,102 @@
+/* From: #include <sys/stack.h> */
+#define MINFRAME                160
+#define MINFRAME32              96
+#define STACK_REGS              48
+#define STACK_REGS32            24
+#define STACK_ALIGN             8
+#define STACK_ALIGN32           STACK_ALIGN
+#define STACK_ENTRY_ALIGN       8
+#define STACK_BIAS              0
+#define SA(X)                   (((X)+(STACK_ALIGN-1)) & ~(STACK_ALIGN-1))
+#define SA32(X)                 (((X)+(STACK_ALIGN-1)) & ~(STACK_ALIGN-1))
+#define FPSAVESZ                136
+#define FPFPC                   128
+
+/*
+ * These crt*.o modules are provided as the bare minimum required
+ * from a crt*.o for inclusion in building low level system
+ * libraries.  The are only be to included in libraries which
+ * contain *no* C++ code and want to avoid the startup code
+ * that the C++ runtime has introduced into the crt*.o modules.
+ *
+ * For further details - see bug#4433015
+ */
+#pragma ident  "@(#)crti.s     1.2     05/06/08 NAF"
+
+       .file           "crti.s"
+
+#ifdef __s390x__
+/*
+ * _init function prologue
+ */
+       .section        ".init"
+       .global         _init
+       .type           _init, @function
+       .align  4
+
+_init:
+       stmg    %r6,%r14,48(%r15)
+        aghi    %r15,-SA(MINFRAME)
+
+
+/*
+ * _fini function prologue
+ */
+       .section        ".fini"
+       .global         _fini
+       .type           _fini, @function
+       .align  4
+
+_fini:
+       stmg    %r6,%r14,48(%r15)
+        aghi    %r15,-SA(MINFRAME)
+#else
+/*
+ * _init function prologue
+ */
```

```
+       .section        ".init"
+       .global         _init
+       .type           _init, @function
+       .align  4
+
+_init:
+       stm     %r6,%r14,24(%r15)
+        ahi     %r15,-SA(MINFRAME32)
+
+/*
+ * _fini function prologue
+ */
+       .section        ".fini"
+       .global         _fini
+       .type           _fini, @function
+       .align  4
+
+_fini:
+       stm     %r6,%r14,24(%r15)
+        ahi     %r15,-SA(MINFRAME32)
+#endif
```

12.1.14 gcc/config/s390/sol2.h

```
===================================================================
--- gcc/config/s390/sol2.h      (revision 0)
+++ gcc/config/s390/sol2.h      (revision 0)
@@ -0,0 +1,256 @@
+/* Definitions for Solaris & Linux for S/390.
+   Copyright (C) 1999, 2000, 2001, 2002 Free Software Foundation, Inc.
+   Contributed by Hartmut Penner (hpenner@de.ibm.com) and
+                  Ulrich Weigand (uweigand@de.ibm.com).
+
+This file is part of GCC.
+
+GCC is free software; you can redistribute it and/or modify it under
+the terms of the GNU General Public License as published by the Free
+Software Foundation; either version 2, or (at your option) any later
+version.
+
+GCC is distributed in the hope that it will be useful, but WITHOUT ANY
+WARRANTY; without even the implied warranty of MERCHANTABILITY or
+FITNESS FOR A PARTICULAR PURPOSE.  See the GNU General Public License
+for more details.
+
+You should have received a copy of the GNU General Public License
+along with GCC; see the file COPYING.  If not, write to the Free
+Software Foundation, 59 Temple Place - Suite 330, Boston, MA
+02111-1307, USA.  */
+
+#include <config/sol2.h>
+
+#ifndef _SOL2_H
+#define _SOL2_H
+
+/* Switch back to dwarf-2 as default */
+#undef PREFERRED_DEBUGGING_TYPE
+#define PREFERRED_DEBUGGING_TYPE DWARF2_DEBUG
+#undef DWARF2_DEBUGGING_INFO
+#define DWARF2_DEBUGGING_INFO 1
+
+#undef ASM_APP_ON
```

```
+#define ASM_APP_ON "#APP\n"
+#undef ASM_APP_OFF
+#define ASM_APP_OFF "#NO_APP\n"
+
+/* Target specific version string.  */
+
+#ifdef DEFAULT_TARGET_64BIT
+#undef  TARGET_VERSION
+#define TARGET_VERSION fprintf (stderr, " (Linux for zSeries)");
+#else
+#undef  TARGET_VERSION
+#define TARGET_VERSION fprintf (stderr, " (Linux for S/390)");
+#endif
+
+/* Target specific type definitions.  */
+
+/* ??? Do we really want long as size_t on 31-bit?  */
+#undef  SIZE_TYPE
+#define SIZE_TYPE (TARGET_64BIT ? "long unsigned int" : "long unsigned int")
+#undef  PTRDIFF_TYPE
+#define PTRDIFF_TYPE (TARGET_64BIT ? "long int" : "int")
+
+#undef  WCHAR_TYPE
+#define WCHAR_TYPE "int"
+#undef  WCHAR_TYPE_SIZE
+#define WCHAR_TYPE_SIZE 32
+
+/* Target specific assembler settings.  */
+
+#undef  ASM_SPEC
+#define ASM_SPEC "%{m31&m64}%{mesa&mzarch}%{march=*}"
+
+/* Target specific linker settings.  */
+
+#ifdef DEFAULT_TARGET_64BIT
+#define MULTILIB_DEFAULTS { "m64" }
+#else
+#define MULTILIB_DEFAULTS { "m31" }
+#endif
+
+/* Because libgcc can generate references back to libc (via .umul etc.) we
have
+   to list libc again after the second libgcc.  */
+#define LINK_GCC_C_SEQUENCE_SPEC "%G %L %G %L"
+
+#undef  LINK_SPEC
+#define LINK_SPEC \
+  "%(link_arch) \
+   %{shared:-shared -G -dy} \
+   %{h*} \
+   %{b} \
+   %{symbolic:-Bsymbolic -G -dy -z text} \
+   %{Qy:} %{!Qn:-Qy}"
+
+
+#define TARGET_ASM_FILE_END file_end_indicate_exec_stack
+
+#define MD_UNWIND_SUPPORT "config/s390/sol2-unwind.h"
+
+/* Output a simple call for .init/.fini.  */
+#define ASM_OUTPUT_CALL(FILE, FN)                                       \
+  do                                                                    \
+    {                                                                   \
```

```
+      if (TARGET_64BIT)                                                \
+      {                                                                \
+           fprintf (FILE, "\tbrasl %%r14,");                   \
+      }                                                                \
+      else {                                                           \
+           fprintf (FILE, "\tbrasl %%r14,");                   \
+      }                                                                \
+      print_operand (FILE, XEXP (DECL_RTL (FN), 0), 0);                \
+      fprintf (FILE, "\n");                                \
+    }                                                      \
+  while (0)
+
+/* We don't use the standard LIB_SPEC only because we don't yet support c++.
*/
+#undef LIB_SPEC
+#define LIB_SPEC \
+  "%{compat-bsd:-lucb -lsocket -lnsl -lelf -laio} \
+   %{!shared:\
+     %{!symbolic:\
+       %{pthreads|pthread:-lpthread} \
+       %{!pthreads:%{!pthread:%{threads:-lthread}}} \
+       -ldl -lc} \
+      }"
+
+#undef LINK_ARCH31_SPEC_BASE
+#define LINK_ARCH31_SPEC_BASE \
+  " -m elf_s390 \
+   %{G:-G} \
+   %{YP,*} \
+   %{R*} \
+   %{compat-bsd: \
+     %{!YP,*:%{p|pg:-Y
P,/usr/ucblib:/usr/ccs/lib/libp:/usr/lib/libp:/usr/ccs/lib:/usr/lib:/lib \
+            -rpath-
link=/usr/ucblib:/usr/ccs/lib/libp:/usr/lib/libp:/usr/ccs/lib:/usr/lib:/lib \
+              } \
+             %{!p:%{!pg:-Y P,/usr/ucblib:/usr/ccs/lib:/usr/lib:/lib \
+                -rpath-link=/usr/ucblib:/usr/ccs/lib:/usr/lib:/lib \
+      }} \
+      } \
+    -R /usr/ucblib} \
+   %{!compat-bsd: \
+     %{!YP,*:%{p|pg:-Y
P,/usr/ccs/lib/libp:/usr/lib/libp:/usr/ccs/lib:/usr/lib:/lib \
+            -rpath-
link=/usr/ccs/lib/libp:/usr/lib/libp:/usr/ccs/lib:/usr/lib:/lib \
+              } \
+             %{!p:%{!pg:-Y P,/usr/ccs/lib:/usr/lib:/lib \
+                -rpath-link=/usr/ccs/lib:/usr/lib:/lib \
+              }} \
+      } \
+    } \
+   %{!shared: \
+      %{static:-static -dn} \
+      %{!static: \
+       %{rdynamic:-export-dynamic} \
+       %{!dynamic-linker: \
+               -dynamic-linker /lib/ld.so.1 \
+         } \
+       } \
+    } "
+
+#undef LINK_ARCH31_SPEC
```

```
+#define LINK_ARCH31_SPEC LINK_ARCH31_SPEC_BASE
+
+#undef LINK_ARCH64_SPEC_BASE
+#define LINK_ARCH64_SPEC_BASE \
+ " -m elf64_s390 \
+   %{G:-G} \
+   %{YP,*} \
+   %{R*} \
+   %{compat-bsd: \
+     %{!YP,*:%{p|pg:-Y
P,/usr/ucblib/s390x:/usr/lib/libp/s390x:/usr/lib/s390x:/lib/s390x \
+               -rpath-
link=/usr/ucblib/s390x:/usr/lib/libp/s390x:/usr/lib/s390x:/lib/s390x \
+                } \
+       %{!p:%{!pg:-Y P,/usr/ucblib/s390x:/usr/lib/s390x:/lib/s390x \
+          -rpath-link=/usr/ucblib/s390x:/usr/lib/s390x:/lib/s390x \
+        }} \
+      } \
+     -R /usr/ucblib/s390x \
+    } \
+   %{!compat-bsd: \
+     %{!YP,*:%{p|pg:-Y P,/usr/lib/libp/s390x:/usr/lib/s390x:/lib/s390x \
+            -rpath-link=/usr/lib/libp/s390x:/usr/lib/s390x:/lib/s390x \
+              } \
+       %{!p:%{!pg:-Y P,/usr/lib/s390x:/lib/s390x \
+            -rpath-link=/usr/lib/s390x:/lib/s390x \
+        }} \
+      } \
+    } \
+   %{!shared: \
+      %{static:-static -dn} \
+      %{!static: \
+       %{rdynamic:-export-dynamic} \
+       %{!dynamic-linker: \
+               -dynamic-linker /lib/s390x/ld.so.1 \
+         } \
+       } \
+    } "
+
+#undef LINK_ARCH64_SPEC
+#define LINK_ARCH64_SPEC LINK_ARCH64_SPEC_BASE
+
+#undef LINK_ARCH_DEFAULT_SPEC
+#ifdef DEFAULT_TARGET_64BIT
+#define LINK_ARCH_DEFAULT_SPEC LINK_ARCH64_SPEC
+#else
+#define LINK_ARCH_DEFAULT_SPEC LINK_ARCH31_SPEC
+#endif
+
+#undef LINK_ARCH_SPEC
+#if DISABLE_MULTILIB
+#ifdef DEFAULT_TARGET_64BIT
+#define LINK_ARCH_SPEC " \
+%{m31:%edoes not support multilib} \
+%{m64:%(link_arch64)} \
+%{!m31:%{!m64:%(link_arch_default)}} \
+"
+#else
+#define LINK_ARCH_SPEC " \
+%{m31:%(link_arch31)} \
+%{m64:%edoes not support multilib} \
+%{!m31:%{!m64:%(link_arch_default)}} \
+"
```

```
+#endif
+#else
+#define LINK_ARCH_SPEC " \
+%{m31:%(link_arch31)} \
+%{m64:%(link_arch64)} \
+%{!m31:%{!m64:%(link_arch_default)}} \
+"
+#endif
+
+#undef STARTFILE_SPEC
+#define STARTFILE_SPEC "%{!shared: \
+                         %{!symbolic: \
+                           %{p:crt1.o%s} \
+                           %{!p: \
+                             %{pg:crt1.o%s} \
+                             %{!pg:crt1.o%s} \
+                            } \
+                           } \
+                          } \
+                        crti.o%s \
+                        crtbegin.o%s"
+
+#undef ENDFILE_SPEC
+#define ENDFILE_SPEC "crtend.o%s crtn.o%s"
+
+/*  { "startfile_arch",        STARTFILE_ARCH_SPEC },            */
+#undef SUBTARGET_EXTRA_SPECS
+#define SUBTARGET_EXTRA_SPECS \
+  { "link_arch31",       LINK_ARCH31_SPEC },              \
+  { "link_arch64",       LINK_ARCH64_SPEC },              \
+  { "link_arch_default", LINK_ARCH_DEFAULT_SPEC },       \
+  { "link_arch",        LINK_ARCH_SPEC }
+
+#define EXTRA_SPECS SUBTARGET_EXTRA_SPECS
+
+/* Solaris-specific #pragmas are implemented on top of attributes.  Hook in
+   the bits from config/sol2.c.  */
+#define SUBTARGET_INSERT_ATTRIBUTES solaris_insert_attributes
+#define SUBTARGET_ATTRIBUTE_TABLE SOLARIS_ATTRIBUTE_TABLE
+
+#endif
```

12.1.15 gcc/config/s390/crtn.s

```
===================================================================
--- gcc/config/s390/crtn.s      (revision 0)
+++ gcc/config/s390/crtn.s      (revision 0)
@@ -0,0 +1,93 @@
+#define MINFRAME                160
+#define MINFRAME32              96
+#define STACK_REGS              48
+#define STACK_REGS32            24
+#define STACK_ALIGN             8
+#define STACK_ALIGN32           STACK_ALIGN
+#define STACK_ENTRY_ALIGN       8
+#define STACK_BIAS              0
+#define SA(X)                   (((X)+(STACK_ALIGN-1)) & ~(STACK_ALIGN-1))
+#define SA32(X)                 (((X)+(STACK_ALIGN-1)) & ~(STACK_ALIGN-1))
+#define FPSAVESZ                136
+#define FPFPC                   128
+
+/*
```

```
+ * These crt*.o modules are provided as the bare minimum required
+ * from a crt*.o for inclusion in building low level system
+ * libraries.  The are only be to included in libraries which
+ * contain *no* C++ code and want to avoid the startup code
+ * that the C++ runtime has introduced into the crt*.o modules.
+ *
+ * For further details - see bug#4433015
+ */
+#pragma ident  "@(#)crtn.s     1.2     05/06/08 NAF"
+
+       .file           "crtn.s"
+
+#ifdef __s390x__
+/*
+ * _init function epilogue
+ */
+       .section        ".init"
+       .align  4
+
+        aghi    %r15,SA(MINFRAME)
+       lmg     %r6,%r14,48(%r15)
+       br      %r14
+
+/*
+ * _fini function epilogue
+ */
+       .section        ".fini"
+       .align  4
+
+        aghi    %r15,SA(MINFRAME)
+       lmg     %r6,%r14,48(%r15)
+       br      %r14
+#else
+/*
+ * _init function epilogue
+ */
+       .section        ".init"
+       .align  4
+
+        ahi     %r15,SA(MINFRAME32)
+       lm      %r6,%r14,24(%r15)
+       br      %r14
+
+/*
+ * _fini function epilogue
+ */
+       .section        ".fini"
+       .align  4
+
+        ahi     %r15,SA(MINFRAME32)
+       lm      %r6,%r14,24(%r15)
+       br      %r14
+#endif
```

12.1.16 gcc/config/s390/sol2-unwind.h

```
===================================================================
--- gcc/config/s390/sol2-unwind.h       (revision 0)
+++ gcc/config/s390/sol2-unwind.h       (revision 0)
@@ -0,0 +1,139 @@
+/* DWARF2 EH unwinding support for S/390 Solaris.
+   Copyright (C) 2004, 2005 Free Software Foundation, Inc.
```

```
+
+This file is part of GCC.
+
+GCC is free software; you can redistribute it and/or modify it under
+the terms of the GNU General Public License as published by the Free
+Software Foundation; either version 2, or (at your option) any later
+version.
+
+In addition to the permissions in the GNU General Public License, the
+Free Software Foundation gives you unlimited permission to link the
+compiled version of this file with other programs, and to distribute
+those programs without any restriction coming from the use of this
+file.  (The General Public License restrictions do apply in other
+respects; for example, they cover modification of the file, and
+distribution when not linked into another program.)
+
+GCC is distributed in the hope that it will be useful, but WITHOUT ANY
+WARRANTY; without even the implied warranty of MERCHANTABILITY or
+FITNESS FOR A PARTICULAR PURPOSE.  See the GNU General Public License
+for more details.
+
+You should have received a copy of the GNU General Public License
+along with GCC; see the file COPYING.  If not, write to the Free
+Software Foundation, 59 Temple Place - Suite 330, Boston, MA
+02111-1307, USA.  */
+
+/* Do code reading to identify a signal frame, and set the frame
+   state data appropriately.  See unwind-dw2.c for the structs.  */
+
+#define MD_FALLBACK_FRAME_STATE_FOR s390_fallback_frame_state
+
+static _Unwind_Reason_Code
+s390_fallback_frame_state (struct _Unwind_Context *context,
+                           _Unwind_FrameState *fs)
+{
+  unsigned char *pc = context->ra;
+  long new_cfa;
+  int i;
+
+  typedef struct
+  {
+    unsigned long psw_mask;
+    unsigned long psw_addr;
+    unsigned long gprs[16];
+    unsigned int  acrs[16];
+    unsigned int  fpc;
+    unsigned int  __pad;
+    double        fprs[16];
+  } __attribute__ ((__aligned__ (8))) sigregs_;
+
+  sigregs_ *regs;
+  int *signo = NULL;
+
+  /* svc $__NR_sigreturn or svc $__NR_rt_sigreturn  */
+  if (pc[0] != 0x0a || (pc[1] != 119 && pc[1] != 173))
+    return _URC_END_OF_STACK;
+
+  /* New-style RT frame:
+     retcode + alignment (8 bytes)
+     siginfo (128 bytes)
+     ucontext (contains sigregs)  */
+  if (context->ra == context->cfa)
+    {
```

```
+      struct ucontext_
+      {
+       unsigned long     uc_flags;
+       struct ucontext_ *uc_link;
+       unsigned long     uc_stack[3];
+       sigregs_          uc_mcontext;
+      } *uc = context->cfa + 8 + 128;
+
+      regs = &uc->uc_mcontext;
+      signo = context->cfa + sizeof(long);
+    }
+
+  /* Old-style RT frame and all non-RT frames:
+     old signal mask (8 bytes)
+     pointer to sigregs  */
+  else
+    {
+      regs = *(sigregs_ **)(context->cfa + 8);
+
+      /* Recent kernels store the signal number immediately after
+        the sigregs; old kernels have the return trampoline at
+        this location.  */
+      if ((void *)(regs + 1) != context->ra)
+       signo = (int *)(regs + 1);
+    }
+
+  new_cfa = regs->gprs[15] + 16*sizeof(long) + 32;
+  fs->regs.cfa_how = CFA_REG_OFFSET;
+  fs->regs.cfa_reg = 15;
+  fs->regs.cfa_offset =
+    new_cfa - (long) context->cfa + 16*sizeof(long) + 32;
+
+  for (i = 0; i < 16; i++)
+    {
+      fs->regs.reg[i].how = REG_SAVED_OFFSET;
+      fs->regs.reg[i].loc.offset =
+       (long)&regs->gprs[i] - new_cfa;
+    }
+  for (i = 0; i < 16; i++)
+    {
+      fs->regs.reg[16+i].how = REG_SAVED_OFFSET;
+      fs->regs.reg[16+i].loc.offset =
+       (long)&regs->fprs[i] - new_cfa;
+    }
+
+  /* Load return addr from PSW into dummy register 32.  */
+
+  fs->regs.reg[32].how = REG_SAVED_OFFSET;
+  fs->regs.reg[32].loc.offset = (long)&regs->psw_addr - new_cfa;
+  fs->retaddr_column = 32;
+
+  /* If we got a SIGSEGV or a SIGBUS, the PSW address points *to*
+     the faulting instruction, not after it.  This causes the logic
+     in unwind-dw2.c that decrements the RA to determine the correct
+     CFI region to get confused.  To fix that, we *increment* the RA
+     here in that case.  Note that we cannot modify the RA in place,
+     and the frame state wants a *pointer*, not a value; thus we put
+     the modified RA value into the unused register 33 slot of FS and
+     have the register 32 save address point to that slot.
+
+     Unfortunately, for regular signals on old kernels, we don't know
+     the signal number.  We default to not fiddling with the RA;
+     that can fail in rare cases.  Upgrade your kernel.  */
```

```
+
+  if (signo && (*signo == 11 || *signo == 7))
+    {
+      fs->regs.reg[33].loc.exp =
+       (unsigned char *)regs->psw_addr + 1;
+      fs->regs.reg[32].loc.offset =
+       (long)&fs->regs.reg[33].loc.exp - new_cfa;
+    }
+
+  return _URC_NO_REASON;
+}
```

12.1.17 gcc/config/s390/t-sol2-64

```
===================================================================
--- gcc/config/s390/t-sol2-64   (revision 0)
+++ gcc/config/s390/t-sol2-64   (revision 0)
@@ -0,0 +1,31 @@
+MULTILIB_OPTIONS = m64/m31
+MULTILIB_DIRNAMES = s390x s390
+MULTILIB_MATCHES =
+MULTILIB_OSDIRNAMES = s390x .
+
+LIBGCC = stmp-multilib
+INSTALL_LIBGCC = install-multilib
+
+MULTILIB_CFLAGS +=
+#EXTRA_MULTILIB_PARTS=crtbegin.o crt1.o crtend.o
+EXTRA_MULTILIB_PARTS=crtbegin.o crti.o crt1.o crtn.o crtend.o
+
+# Override t-slibgcc-elf-ver to export some libgcc symbols with
+# the symbol versions that glibc used.
+SHLIB_MAPFILES = $(srcdir)/libgcc-std.ver
+#SHLIB_MAPFILES = $(srcdir)/libgcc-std.ver $(srcdir)/config/s390/libgcc-
glibc.ver
+# Override with nothing as ver. files seem to be breaking shared builds
+#SHLIB_MAPFILES =
+
+$(T)crti.o: $(srcdir)/config/s390/crti.s $(GCC_PASSES)
+       $(GCC_FOR_TARGET) $(GCC_CFLAGS) $(MULTILIB_CFLAGS) $(INCLUDES) -Wa,-
mzarch -march=z990 \
+       -c -o $(T)crti.o -x assembler-with-cpp $(srcdir)/config/s390/crti.s
+
+$(T)crtn.o: $(srcdir)/config/s390/crtn.s $(GCC_PASSES)
+       $(GCC_FOR_TARGET) $(GCC_CFLAGS) $(MULTILIB_CFLAGS) $(INCLUDES) -Wa,-
mzarch -march=z990 \
+       -c -o $(T)crtn.o -x assembler-with-cpp $(srcdir)/config/s390/crtn.s
+
+$(T)crt1.o: $(srcdir)/config/s390/crt1.s $(GCC_PASSES)
+       $(GCC_FOR_TARGET) $(GCC_CFLAGS) $(MULTILIB_CFLAGS) $(INCLUDES) -Wa,-
mzarch -march=z990 \
+       -c -o $(T)crt1.o -x assembler-with-cpp $(srcdir)/config/s390/crt1.s
+
```

12.1.18 gcc/config/s390/crt1.s

```
===================================================================
--- gcc/config/s390/crt1.s      (revision 0)
+++ gcc/config/s390/crt1.s      (revision 0)
@@ -0,0 +1,364 @@
+/*
```

```
+ * This crt1.o module is provided as the bare minimum required to build
+ * a 32-bit executable with gcc.  It is installed in /usr/lib
+ * where it will be picked up by gcc, along with crti.o and crtn.o
+ */
+
+       .ident  "@(#)crt1.s     1.2     05/06/08 SMI"
+
+       .file   "crt1.s"
+
+#ifdef __s390x__
+       .globl  _start
+
+/* global entities defined elsewhere but used here */
+       .globl  main
+       .globl  exit
+       .globl  _exit
+       .weak   _DYNAMIC
+
+       .section        .data
+
+       .weak   environ
+       .set    environ,_environ
+       .globl  _environ
+       .type   _environ,@object
+       .size   _environ,8
+       .align  8
+_environ:
+       .quad   0x0
+
+       .globl  ___Argv
+       .type   ___Argv,@object
+       .size   ___Argv,8
+       .align  8
+___Argv:
+       .quad   0x0
+
+       .section        .text
+       .align  8
+pDYNAMIC:
+       .quad   _DYNAMIC
+
+/*
+ * C language startup routine.
+ * R2 - argc
+ * R3 - A(argv)
+ * R4 - A(envp)
+ *
+ * Allocate a NULL return address and a NULL previous SP as if
+ * there was a genuine call to _start.
+ * sdb stack trace shows _start(argc,argv[0],argv[1],...,envp[0],...)
+ */
+       .type   _start,@function
+_start:
+       larl    %r13,pDYNAMIC
+       lgr     %r6,%r2                 // Save argc
+       lgr     %r7,%r3                 // Save **argv
+       lgr     %r8,%r4                 // Save envp
+       lgr     %r9,%r5                 // Save exit function
+       lg      %r5,0(%r13)             // Get _DYNAMIC
+       ltgr    %r5,%r5                 // Set?
+       jz      1f                      // No... Skip
+
+       lgr     %r2,%r9                 // Copy
```

```
+       brasl   %r14,atexit@PLT         // Go set
+1:
+       larl    %r2,_fini               // Our atexit function
+       brasl   %r14,atexit@PLT         // Go set
+
+/*
+ * The following code provides almost standard static destructor handling
+ * for systems that do not have the modified atexit processing in their
+ * system libraries.  It checks for the existence of the new routine
+ * "_get_exit_frame_monitor()", which is in libc.so when the new exit-handling
+ * code is there.  It then check for the existence of "__Crun::do_exit_code()"
+ * which will be in libCrun.so whenever the code was linked with the C++
+ * compiler.  If there is no enhanced atexit, and we do have do_exit_code,
+ * we register the latter with atexit.  There are 5 extra slots in
+ * atexit, so this will still be standard conforming.  Since the code
+ * is registered after the .fini section, it runs before the library
+ * cleanup code, leaving nothing for the calls to _do_exit_code_in_range
+ * to handle.
+ *
+ * Remove this code and the associated code in libCrun when the earliest
+ * system to be supported is Solaris 8.
+ */
+       .weak   _get_exit_frame_monitor
+       .weak   __1cG__CrunMdo_exit_code6F_v_
+
+       .section        .data
+       .align  4
+__get_exit_frame_monitor_ptr:
+       .4byte  _get_exit_frame_monitor
+       .type   __get_exit_frame_monitor_ptr,@object
+       .size   __get_exit_frame_monitor_ptr,4
+
+       .align  4
+__do_exit_code_ptr:
+       .4byte  __1cG__CrunMdo_exit_code6F_v_
+       .type   __do_exit_code_ptr,@object
+       .size   __do_exit_code_ptr,4
+
+       .section        .text
+
+       larl    %r2,__get_exit_frame_monitor_ptr
+       lg      %r2,0(%r2)
+       ltgr    %r2,%r2
+       jz      2f
+
+       larl    %r2,__do_exit_code_ptr
+       lg      %r2,0(%r2)
+       ltgr    %r2,%r2
+       jz      2f
+
+       brasl   %r14,atexit@PLT
+2:
+
+/*
+ * End of destructor handling code
+ */
+
+/*
+ * Calculate the location of the envp array by adding the size of
+ * the argv array to the start of the argv array.
+ */
+
+       larl    %r4,_environ            /* Get A(A(Environment)) */
```

```
+       larl    %r10,___Argv
+       lgr     %r2,%r6                 /* Restore argc */
+       lgr     %r3,%r7                 /* Get Argv */
+       lg      %r11,0(%r4)             /* Get A(Environment) */
+       ltgr    %r11,%r11               /* _environ set? */
+       jnz     3f                      /* Yep... Skip */
+
+       stg     %r8,0(%r4)              /* Copy to _environ */
+3:
+       lgr     %r2,%r6                 /* Restore argc */
+       stg     %r3,0(%r10)
+       lg      %r4,0(%r4)              /* envp */
+       lgr     %r8,%r4                 /* Save envp */
+       brasl   %r14,_init
+       lgr     %r2,%r6                 /* Restore argc - again */
+       lgr     %r3,%r7                 /* .... argv */
+       lgr     %r4,%r8                 /* .... envp */
+       brasl   %r14,main
+       lgr     %r8,%r2                 /* Save return value */
+       brasl   %r14,exit
+       lgr     %r2,%r8
+       jg      _exit
+
+       .size   _start, .-_start
+
+/*
+ * The following is here in case any object module compiled with cc -p
+ *     was linked into this module.
+ */
+       .section        .text
+       .align  4
+       .globl  _mcount
+       .type   _mcount,@function
+_mcount:
+       br      %r14
+       .size   _mcount, .-_mcount
+
+       .section        .data
+
+       .globl  __longdouble_used
+       .type   __longdouble_used,@object
+       .size   __longdouble_used,4
+       .align  4
+__longdouble_used:
+       .4byte  0x0
+#else
+       .globl  _start
+
+/* global entities defined elsewhere but used here */
+       .globl  main
+       .globl  exit
+       .globl  _exit
+       .weak   _DYNAMIC
+
+       .section        .data
+
+       .weak   environ
+       .set    environ,_environ
+       .globl  _environ
+       .type   _environ,@object
+       .size   _environ,4
+       .align  4
+_environ:
```

```
+       .4byte  0x0
+
+       .globl  ___Argv
+       .type   ___Argv,@object
+       .size   ___Argv,4
+       .align  4
+___Argv:
+       .4byte  0x0
+
+       .section        .text
+       .align  4
+.crt1cons:
+pDYNAMIC:
+       .long   _DYNAMIC
+p_fini:
+       .long   _fini
+
+/*
+ * C language startup routine.
+ * R2 - argc
+ * R3 - A(argv)
+ * R4 - A(envp)
+ *
+ * Allocate a NULL return address and a NULL previous SP as if
+ * there was a genuine call to _start.
+ * sdb stack trace shows _start(argc,argv[0],argv[1],...,envp[0],...)
+ */
+       .type   _start,@function
+_start:
+       larl    %r13,pDYNAMIC
+       lr      %r6,%r2                 // Save argc
+       lr      %r7,%r3                 // Save **argv
+       lr      %r8,%r4                 // Save envp
+       lr      %r9,%r5                 // Save exit function
+       l       %r5,0(%r13)
+       ltr     %r5,%r5
+       jz      1f
+
+       lr      %r2,%r9                 // Get exit function
+       brasl   %r14,atexit@PLT         // Go set
+1:
+       larl    %r2,_fini               // Our atexit function
+       brasl   %r14,atexit@PLT         // Go set
+
+/*
+ * The following code provides almost standard static destructor handling
+ * for systems that do not have the modified atexit processing in their
+ * system libraries.  It checks for the existence of the new routine
+ * "_get_exit_frame_monitor()", which is in libc.so when the new exit-handling
+ * code is there.  It then check for the existence of "__Crun::do_exit_code()"
+ * which will be in libCrun.so whenever the code was linked with the C++
+ * compiler.  If there is no enhanced atexit, and we do have do_exit_code,
+ * we register the latter with atexit.  There are 5 extra slots in
+ * atexit, so this will still be standard conforming.  Since the code
+ * is registered after the .fini section, it runs before the library
+ * cleanup code, leaving nothing for the calls to _do_exit_code_in_range
+ * to handle.
+ *
+ * Remove this code and the associated code in libCrun when the earliest
+ * system to be supported is Solaris 8.
+ */
+       .weak   _get_exit_frame_monitor
+       .weak   __1cG__CrunMdo_exit_code6F_v_
```

```
+
+       .section        .data
+       .align  4
+__get_exit_frame_monitor_ptr:
+       .4byte  _get_exit_frame_monitor
+       .type   __get_exit_frame_monitor_ptr,@object
+       .size   __get_exit_frame_monitor_ptr,4
+
+       .align  4
+__do_exit_code_ptr:
+       .4byte  __1cG__CrunMdo_exit_code6F_v_
+       .type   __do_exit_code_ptr,@object
+       .size   __do_exit_code_ptr,4
+
+       .section        .text
+
+       larl    %r2,__get_exit_frame_monitor_ptr
+       l       %r2,0(%r2)
+       ltr     %r2,%r2
+       jz      2f
+
+       larl    %r2,__do_exit_code_ptr
+       l       %r2,0(%r2)
+       ltr     %r2,%r2
+       jz      2f
+
+       brasl   %r14,atexit@PLT
+2:
+
+/*
+ * End of destructor handling code
+ */
+
+/*
+ * Calculate the location of the envp array by adding the size of
+ * the argv array to the start of the argv array.
+ */
+
+       larl    %r4,_environ            /* Get A(A(Environment)) */
+       larl    %r10,___Argv
+       lr      %r2,%r6                 /* Restore argc */
+       lr      %r3,%r7                 /* Get Argv */
+       l       %r11,0(%r4)
+       ltr     %r11,%r11               /* _environ set? */
+       jnz     3f                      /* Yes... Skip */
+
+       st      %r8,0(%r4)              /* copy to _environ */
+3:
+       lr      %r2,%r6                 /* Restore argc */
+       st      %r3,0(%r10)
+       l       %r4,0(%r4)              /* envp */
+       lr      %r8,%r4                 /* Save envp */
+       brasl   %r14,_init
+       lr      %r2,%r6                 /* Restore argc - again */
+       lr      %r3,%r7                 /* .... argv */
+       lr      %r4,%r8                 /* .... envp */
+       brasl   %r14,main
+       lr      %r8,%r2                 /* Save return value */
+       brasl   %r14,exit
+       lr      %r2,%r8
+       jg      _exit
+
+       .size   _start, .-_start
```

```
+
+/*
+ * The following is here in case any object module compiled with cc -p
+ *     was linked into this module.
+ */
+       .section        .text
+       .align  4
+       .globl  _mcount
+       .type   _mcount,@function
+_mcount:
+       br      %r14
+       .size   _mcount, .-_mcount
+
+       .section        .data
+
+       .globl  __longdouble_used
+       .type   __longdouble_used,@object
+       .size   __longdouble_used,4
+       .align  4
+__longdouble_used:
+       .4byte  0x0
+#endif
```

12.1.19 gcc/config/s390/s390x.h

```
===================================================================
--- gcc/config/s390/s390x.h     (revision 139673)
+++ gcc/config/s390/s390x.h     (working copy)
@@ -23,5 +23,6 @@
 #define _S390X_H

 #define DEFAULT_TARGET_64BIT
+#define TARGET_DEFAULT_LONG_DOUBLE_128

 #endif
```

12.1.20 gcc/config/sol2.c

```
===================================================================
--- gcc/config/sol2.c   (revision 139673)
+++ gcc/config/sol2.c   (working copy)
@@ -28,6 +28,8 @@
 #include "toplev.h"
 #include "ggc.h"

+#include <stdio.h>
+
 tree solaris_pending_aligns, solaris_pending_inits, solaris_pending_finis;

 /* Attach any pending attributes for DECL to the list in *ATTRIBUTES.
@@ -41,6 +43,7 @@
 {
   tree *x, next;

+
   if (solaris_pending_aligns != NULL && TREE_CODE (decl) == VAR_DECL)
     for (x = &solaris_pending_aligns; *x; x = &TREE_CHAIN (*x))
       {
```

12.1.21 gcc/config/sol2-c.c

```
===================================================================
--- gcc/config/sol2-c.c (revision 139673)
+++ gcc/config/sol2-c.c (working copy)
@@ -175,8 +175,9 @@
          tree attrs = tree_cons (get_identifier ("used"), NULL, init_list);
          decl_attributes (&decl, attrs, 0);
        }
-      else
+      else {
        solaris_pending_inits = tree_cons (t, NULL, solaris_pending_inits);
+      }

       ttype = pragma_lex (&t);
       if (ttype == CPP_COMMA)
@@ -233,8 +234,9 @@
          tree attrs = tree_cons (get_identifier ("used"), NULL, fini_list);
          decl_attributes (&decl, attrs, 0);
        }
-      else
+      else {
        solaris_pending_finis = tree_cons (t, NULL, solaris_pending_finis);
+      }

       ttype = pragma_lex (&t);
       if (ttype == CPP_COMMA)
@@ -260,6 +262,17 @@
     }
 }

+/* Handle #pragma ident (function [, function]...)  */
+
+static void
+solaris_pragma_ident (cpp_reader *pfile ATTRIBUTE_UNUSED)
+{
+
+  /* TODO - For now, just silently eat idents */
+
+  return;
+}
+
 /* Register Solaris-specific #pragma directives.  */

 void
@@ -268,4 +281,5 @@
   c_register_pragma_with_expansion (0, "align", solaris_pragma_align);
   c_register_pragma (0, "init", solaris_pragma_init);
   c_register_pragma (0, "fini", solaris_pragma_fini);
+  c_register_pragma (0, "ident", solaris_pragma_ident);
 }
```

12.1.22 libstdc++-v3/configure

```
===================================================================
--- libstdc++-v3/configure      (revision 139673)
+++ libstdc++-v3/configure      (working copy)
@@ -2842,6 +2842,9 @@
   gcc_no_link=yes
 fi
```

```
+## HACK
+gcc_no_link=yes
+
 if test x$gcc_no_link = xyes; then
   # Setting cross_compile will disable run tests; it will
   # also disable AC_CHECK_FILE but that's generally
```

12.1.23 libstdc++-v3/Makefile.in

```
===================================================================
--- libstdc++-v3/Makefile.in    (revision 139673)
+++ libstdc++-v3/Makefile.in    (working copy)
@@ -174,7 +174,8 @@
 GLIBCXX_C_HEADERS_EXTRA_TRUE = @GLIBCXX_C_HEADERS_EXTRA_TRUE@
 GLIBCXX_HOSTED_FALSE = @GLIBCXX_HOSTED_FALSE@
 GLIBCXX_HOSTED_TRUE = @GLIBCXX_HOSTED_TRUE@
-GLIBCXX_INCLUDES = @GLIBCXX_INCLUDES@
+## HACK
+GLIBCXX_INCLUDES = -I$(HOME)/OpenSolaris/max/onnv-
gate/proto/root_s390/usr/include @GLIBCXX_INCLUDES@
 GLIBCXX_LDBL_COMPAT_FALSE = @GLIBCXX_LDBL_COMPAT_FALSE@
 GLIBCXX_LDBL_COMPAT_TRUE = @GLIBCXX_LDBL_COMPAT_TRUE@
 GREP = @GREP@
```

12.1.24 libstdc++-v3/src/Makefile.in

```
===================================================================
--- libstdc++-v3/src/Makefile.in        (revision 139673)
+++ libstdc++-v3/src/Makefile.in        (working copy)
@@ -464,7 +464,7 @@
 LTCXXCOMPILE = $(LIBTOOL) --tag CXX --mode=compile $(CXX) $(INCLUDES) \
               $(AM_CPPFLAGS) $(CPPFLAGS) $(AM_CXXFLAGS) $(CXXFLAGS)

-LTLDFLAGS = $(shell $(SHELL) $(top_srcdir)/../libtool-ldflags $(LDFLAGS))
+LTLDFLAGS = $(shell $(SHELL) $(top_srcdir)/../libtool-ldflags $(LDFLAGS)) -
L$(HOME)/OpenSolaris/max/onnv-gate/proto/root_s390/lib -
L$(HOME)/OpenSolaris/max/onnv-gate/proto/root_s390/lib/s390x

 # 3) We'd have a problem when building the shared libstdc++ object if
 # the rules automake generates would be used.  We cannot allow g++ to
```

12.1.25 libstdc++-v3/include/Makefile.in

```
===================================================================
--- libstdc++-v3/include/Makefile.in    (revision 139673)
+++ libstdc++-v3/include/Makefile.in    (working copy)
@@ -142,7 +142,8 @@
 GLIBCXX_C_HEADERS_EXTRA_TRUE = @GLIBCXX_C_HEADERS_EXTRA_TRUE@
 GLIBCXX_HOSTED_FALSE = @GLIBCXX_HOSTED_FALSE@
 GLIBCXX_HOSTED_TRUE = @GLIBCXX_HOSTED_TRUE@
-GLIBCXX_INCLUDES = @GLIBCXX_INCLUDES@
+## HACK
+GLIBCXX_INCLUDES = -I$(HOME)/OpenSolaris/max/onnv-
gate/proto/root_s390/usr/include @GLIBCXX_INCLUDES@
 GLIBCXX_LDBL_COMPAT_FALSE = @GLIBCXX_LDBL_COMPAT_FALSE@
 GLIBCXX_LDBL_COMPAT_TRUE = @GLIBCXX_LDBL_COMPAT_TRUE@
 GREP = @GREP@
```

12.1.26 libstdc++-v3/crossconfig.m4

```
===================================================================
--- libstdc++-v3/crossconfig.m4 (revision 139673)
+++ libstdc++-v3/crossconfig.m4 (working copy)
@@ -337,6 +337,7 @@
          AC_DEFINE(HAVE_WCHAR_H)
          AC_DEFINE(HAVE_WCTYPE_H)
          AC_DEFINE(HAVE_LIBM)
+         AC_DEFINE(HAVE_STRSIGNAL)
         ;;
     esac
     case "$target" in
```

12.1.27 libstdc++-v3/config/os/newlib/ctype_noninline.h

```
===================================================================
--- libstdc++-v3/config/os/newlib/ctype_noninline.h     (revision 139673)
+++ libstdc++-v3/config/os/newlib/ctype_noninline.h     (working copy)
@@ -40,7 +40,7 @@

   const ctype_base::mask*
   ctype<char>::classic_table() throw()
-  { return _ctype_ + 1; }
+  { return (const char *)__ctype + 1; }

   ctype<char>::ctype(__c_locale, const mask* __table, bool __del,
                      size_t __refs)
```

12.1.28 boehm-gc/include/private/gcconfig.h

```
===================================================================
--- boehm-gc/include/private/gcconfig.h (revision 139673)
+++ boehm-gc/include/private/gcconfig.h (working copy)
@@ -452,6 +452,11 @@
 #    define S390
 #    define mach_type_known
 # endif
+# if defined(__s390__) && defined(__sun__)
+#    define S390
+#    define SUNOS5
+#    define mach_type_known
+# endif
 # if defined(__GNU__)
 #   if defined(__i386__)
 /* The Debian Hurd running on generic PC */
```

12.1.29 libiberty/config.in

```
===================================================================
--- libiberty/config.in (revision 139673)
+++ libiberty/config.in (working copy)
@@ -442,3 +442,5 @@

 /* Define as `fork' if `vfork' does not work. */
 #undef vfork
+
+#define HAVE_STRSIGNAL 1
```

12.1.30 libiberty/configure

```
===================================================================
--- libiberty/configure (revision 139673)
+++ libiberty/configure (working copy)
@@ -5235,6 +5235,7 @@
 fi
 echo "$as_me:$LINENO: result: `eval echo '${'$as_ac_var'}'`" >&5
 echo "${ECHO_T}`eval echo '${'$as_ac_var'}'`" >&6
+
 if test `eval echo '${'$as_ac_var'}'` = yes; then
   cat >>confdefs.h <<_ACEOF
 #define `echo "HAVE_$ac_func" | $as_tr_cpp` 1
@@ -5243,6 +5244,16 @@
 fi
 done

+## HACK
+echo "------------------------------------------------------------"
+echo "DIAG: strsignal=$ac_cv_func_strsignal"
+ac_cv_func_strsignal="yes"
+HAVE_STRSIGNAL="yes"
+cat >>confdefs.h <<_ACEOF
+#define `echo "HAVE_STRSIGNAL" | $as_tr_cpp` 1
+_ACEOF
+echo "------------------------------------------------------------"
+
   echo "$as_me:$LINENO: checking whether basename is declared" >&5
 echo $ECHO_N "checking whether basename is declared... $ECHO_C" >&6
 if test "${ac_cv_have_decl_basename+set}" = set; then
```

12.1.31 libssp/Makefile.in

```
===================================================================
--- libssp/Makefile.in  (revision 139673)
+++ libssp/Makefile.in  (working copy)
@@ -123,7 +123,8 @@
 AUTOHEADER = @AUTOHEADER@
 AUTOMAKE = @AUTOMAKE@
 AWK = @AWK@
-CC = @CC@
+## HACK
+CC = @CC@ -L$(HOME)/OpenSolaris/max/onnv-gate/proto/root_s390/lib -
L$(HOME)/OpenSolaris/max/onnv-gate/proto/root_s390/lib/s390x
 CCDEPMODE = @CCDEPMODE@
 CFLAGS = @CFLAGS@
 CPP = @CPP@
```

13. Creating Build Environment

1. SPARC64 platform as build platform
2. Install OpenSolaris Community Edition
3. Install jdk1.5 in local directory (only required for installing Sun Studio)
4. Install Sun Studio 11
5. NFS mount OpenSolaris CD/DVD image
6. Install the following packages:

```
pkgadd -d <mount>/Solaris_11/Product SUNWmercurial
pkgadd -d <mount>/Solaris_11/Product SUNWvim
pkgadd -d <mount>/Solaris_11/Product SUNWsprot
pkgadd -d <mount>/Solaris_11/Product SUNWboot
pkgadd -d <mount>/Solaris_11/Product SUNWgcc
pkgadd -d <mount>/Solaris_11/Product SUNWsfw
```

7. Download SUNWonbld and on-closed-bins from opensolaris.org
8. pkgadd –d ./SUNWonbld
9. From sunfreeware.com download and install ncftp
10. Download and untar usr-local.tar.bz2 (contains x-build gcc etc.)
11. mkdir ~/openSolaris/
12. cd ~/openSolaris
13. hg clone from opensolaris.org
14. Add the following to ./onnv-gate/.hg/hgrc:

```
[web]
allow_push = *
push_ssl = False
style = gitweb
allow_archive = bz2 gz zip
contact = <name>, <email>
```

15. Place opensolaris.sh in ~/openSolaris/
16. Place addenv.sh in ~/openSolaris/
17. bldenv – d opensolaris.sh
18. . addenv.sh
19. cd onnv-gate/usr
20. mkdir –p closed/root_s390
21. cd $SRC
22. dmake install_h
23. Download starter.tar.bz2
24. cd $ROOT/
25. bunzip2 –c starter.tar.bz2 | tar –xf -

14. Changes to Common Code

The following source code changes were made to common code.

Note s390x defines char as unsigned when unqualified.

14.1.1 /usr/src/Makefile.master

- Add definitions for System z platform.

```
--- a/usr/src/Makefile.master    Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/Makefile.master    Tue Sep 02 11:15:47 2008 -0400
@@ -340,6 +340,7 @@
 i386_XARCH=
 amd64_XARCH=       -xarch=amd64 -Ui386 -U__i386
 s390x_XARCH=       -xarch=z9
+s390_XARCH=        -xarch=z9

 # assembler '-xarch' flag.  Different from compiler '-xarch' flag.
 sparc_AS_XARCH=           -xarch=v8plus
@@ -392,7 +393,7 @@
 C99_ENABLE=  -xc99=%all
 C99_DISABLE= -xc99=%none
 C99MODE=     $(C99_DISABLE)
-C99LMODE=    $(C99MODE:-xc99%=-Xc99%)
+#C99LMODE=   $(C99MODE:-xc99%=-Xc99%)

 # In most places, assignments to these macros should be appended with +=
 # (CPPFLAGS.master allows values to be prepended to CPPFLAGS).
@@ -401,8 +402,10 @@
              $(CCSTATICSYM)
 i386_CFLAGS= $(i386_XARCH)
 amd64_CFLAGS=      $(amd64_XARCH)
-s390x_CFLAGS=      $(s390x_XARCH) -_gcc=-gdwarf-2 -D__s390x -_gcc=-mbackchain
-s390_CFLAGS= $(s390x_XARCH) -_gcc=-gdwarf-2 -D__s390 -_gcc=-mbackchain -_gcc=-
m31
+s390x_CFLAGS=      $(s390x_XARCH) -_gcc=-gdwarf-2 -D__s390x -_gcc=-mbackchain
\
+             -_gcc=-Wa,-mzarch -_gcc=-fgnu89-inline
+s390_CFLAGS= $(s390_XARCH) -_gcc=-gdwarf-2 -D__s390 -_gcc=-mbackchain \
+             -_gcc=-Wa,-mzarch -_gcc=-fgnu89-inline

 sparc_ASFLAGS=     $(sparc_AS_XARCH)
 sparcv9_ASFLAGS=$(sparcv9_AS_XARCH)
@@ -455,7 +458,7 @@
 #
 CTF_FLAGS_sparc    = -g -Wc,-Qiselect-T1 $(C99MODE) $(CNOGLOBAL) $(CDWARFSTR)
 CTF_FLAGS_i386     = -g $(C99MODE) $(CNOGLOBAL) $(CDWARFSTR)
-CTF_FLAGS_s390x = -g $(C99MODE) -XO0 $(CNOGLOBAL) $(CDWARFSTR) -_gcc=-fno-
eliminate-unused-debug-types
+CTF_FLAGS_s390  = -g $(C99MODE) -xO0 $(CNOGLOBAL) $(CDWARFSTR) -_gcc=-fno-
eliminate-unused-debug-types
 CTF_FLAGS    = $(CTF_FLAGS_$(MACH)) $(DEBUGFORMAT)

 #
@@ -784,12 +787,18 @@
 s390_BDIRECT=
 BDIRECT=     $($(MACH)_BDIRECT)
 BDYNAMIC=    -Bdynamic
```

```
-BLOCAL=            -Blocal
+sparc_BLOCAL=      -Blocal
+i386_BLOCAL=-Blocal
+s390_BLOCAL=
+BLOCAL=            $($(MACH)_BLOCAL)
 sparc_BNODIRECT=-Bdirect
 i386_BNODIRECT=    -Bdirect
 s390_BNODIRECT=
 BNODIRECT=  $($(MACH)_BNODIRECT)
-BREDUCE=    -Breduce
+sparc_BREDUCE=     -Breduce
+i386_BREDUCE=      -Breduce
+s390_BREDUCE=
+BREDUCE=    $($(MACH)_BREDUCE)
 BSTATIC=    -Bstatic

 ZDEFS=             -zdefs
@@ -800,6 +809,7 @@
 ZLAZYLOAD=  -zlazyload
 ZLOADFLTR=  -zloadfltr
 ZMULDEFS=   -zmuldefs
+ZCOMBRELOC= -zcombreloc
 ZNODEFAULTLIB=     -znodefaultlib
 ZNODEFS=    -znodefs
 ZNODELETE=  -znodelete
@@ -823,7 +833,8 @@
 sparcv9_C_PICFLAGS =      -K pic
 i386_C_PICFLAGS =  -K pic
 amd64_C_PICFLAGS = -K pic
-s390x_C_PICFLAGS = -K pic
+s390x_C_PICFLAGS = -K PIC
+s390_C_PICFLAGS =  -K PIC
 C_PICFLAGS =       $($(MACH)_C_PICFLAGS)
 C_PICFLAGS64 =             $($(MACH64)_C_PICFLAGS)

@@ -832,6 +843,7 @@
 i386_C_BIGPICFLAGS =      -K PIC
 amd64_C_BIGPICFLAGS =     -K PIC
 s390x_C_BIGPICFLAGS =     -K PIC
+s390_C_BIGPICFLAGS =      -K PIC
 C_BIGPICFLAGS =           $($(MACH)_C_BIGPICFLAGS)
 C_BIGPICFLAGS64 =  $($(MACH64)_C_BIGPICFLAGS)

@@ -845,8 +857,14 @@
 CC_PICFLAGS =              $($(MACH)_CC_PICFLAGS)
 CC_PICFLAGS64 =            $($(MACH64)_CC_PICFLAGS)

-AS_PICFLAGS=       $(C_PICFLAGS)
-AS_BIGPICFLAGS=            $(C_BIGPICFLAGS)
+sparc_AS_PICFLAGS =       $(C_PICFLAGS)
+i386_AS_PICFLAGS = $(C_PICFLAGS)
+s390_AS_PICFLAGS =
+AS_PICFLAGS=       $($(MACH)_AS_PICFLAGS)
+sparc_AS_BIGPICFLAGS =    $(C_BIGPICFLAGS)
+i386_AS_BIGPICFLAGS =     $(C_BIGPICFLAGS)
+s390_AS_BIGPICFLAGS =
+AS_BIGPICFLAGS=            $($(MACH)_AS_BIGPICFLAGS)

 #
 # Default label for CTF sections

--- a/usr/src/Makefile.master   Tue Sep 02 11:22:56 2008 -0400
+++ b/usr/src/Makefile.master   Thu Sep 25 10:26:05 2008 -0400
```

```
@@ -412,7 +412,7 @@
 i386_ASFLAGS=  $(i386_AS_XARCH)
 amd64_ASFLAGS= $(amd64_AS_XARCH)
 s390x_ASFLAGS= $(s390x_AS_XARCH)
-s390_ASFLAGS=  $(s390x_AS_XARCH)
+s390_ASFLAGS=  $(s390_AS_XARCH)

 #
 sparc_COPTFLAG=                -xO3
@@ -650,7 +650,10 @@
 #
 # Link time macros
 #
-CCNEEDED                = -lC
+sparc_CCNEEDED          = -lC
+i386_CCNEEDED           = -lC
+s390_CCNEEDED           = -lstdc++ -lgcc_s
+CCNEEDED                = $($(MACH)_CCNEEDED)
 $(__GNUC)CCNEEDED       = -L$(SFWLIBDIR) -R$(SFWLIBDIR) -lstdc++ -lgcc_s

 LINK.c=                 $(CC) $(CFLAGS) $(CPPFLAGS) $(LDFLAGS)
@@ -762,6 +765,14 @@
 NATIVEAS=               $($(NATIVE_MACH)_AS)
 NATIVELD=               $($(NATIVE_MACH)_LD)
 NATIVELINT=             $($(NATIVE_MACH)_LINT)
+
+sparc_XBUILDCC=                 $(NATIVECC)
+sparcv9_XBUILDCC=       $(NATIVECC)
+i386_XBUILDCC=          $(NATIVECC)
+amd64_XBUILDCC=                 $(NATIVECC)
+s390_XBUILDCC=          $(sparc_CC)
+s390x_XBUILDCC=                 $(sparcv9_CC)
+XBUILDCC=               $($(MACH)_XBUILDCC)

 #
 # Makefile.master.64 overrides these settings
```

14.1.2 /usr/src/Makefile.master.64

- System z uses gcc so requires the gcc libraries. Implement CCNEEDED using the $(MACH) mechanism used elsewhere.

```
--- a/usr/src/Makefile.master.64        Tue Sep 02 11:22:56 2008 -0400
+++ b/usr/src/Makefile.master.64        Thu Sep 25 10:26:05 2008 -0400
@@ -76,7 +76,10 @@
 BUILD.SO=       $(CC) $(CFLAGS) -o $@ $(GSHARED) $(DYNFLAGS) \
                 $(PICS) -L $(ROOTLIBDIR64) $(LDLIBS)

-CCNEEDED =              -lCrun
+sparcv9_CCNEEDED        = -lCrun
+amd64_CCNEEDED          = -lCrun
+s390x_CCNEEDED          = -lstdc++ -lgcc_s
+CCNEEDED                = $($(MACH64)_CCNEEDED)
 $(__GNUC64)CCNEEDED =   -L$(SFWLIBDIR) -R$(SFWLIBDIR) -lstdc++  \
                         -lgcc_s_$(MACH64)
```

14.1.3 /usr/src/cmd/Makefile.cmd

- System z uses gcc so requires the gcc libraries. Implement LDLIBS using the $(MACH) mechanism used elsewhere.

```
diff -r 4f051ff1b998 usr/src/cmd/Makefile.cmd
--- a/usr/src/cmd/Makefile.cmd    Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/cmd/Makefile.cmd    Tue Sep 02 11:15:47 2008 -0400
@@ -134,11 +134,17 @@
 ISAEXEC=     $(ROOT)/usr/lib/isaexec
 PLATEXEC=    $(ROOT)/usr/lib/platexec

-LDLIBS =     $(LDLIBS.cmd:-L%=-L% -Wl,--rpath-link %) -ldl -lxml2
+s390_XLDLIBS=       -ldl -lgcc_s
+XLDLIBS=     $($(MACH)_XLDLIBS)
+LDLIBS =     $(LDLIBS.cmd:-L%=-L% -Wl,--rpath-link %) $(XLDLIBS)
+
+sparc_MAPFILES=     $(MAPFILE.NES:%=-M%) $(MAPFILE.PGA:%=-M%) $(MAPFILE.NED:%=-
M%)
+i386_MAPFILES=      $(MAPFILE.NES:%=-M%) $(MAPFILE.PGA:%=-M%) $(MAPFILE.NED:%=-
M%)
+s390_MAPFILES=

 LDFLAGS.cmd = \
       $(BDIRECT) $(ENVLDFLAGS1) $(ENVLDFLAGS2) $(ENVLDFLAGS3) \
-      $(MAPFILE.NES:%=-M%) $(MAPFILE.PGA:%=-M%) $(MAPFILE.NED:%=-M%)
+      $($(MACH)_MAPFILES)

 LDFLAGS =    $(LDFLAGS.cmd)
```

14.1.4 /usr/src/cmd/audio/include/aiff.h

```
--- a/usr/src/cmd/audio/include/aiff.h  Tue Sep 02 11:22:56 2008 -0400
+++ b/usr/src/cmd/audio/include/aiff.h  Thu Sep 25 10:26:06 2008 -0400
@@ -141,7 +141,7 @@


 /* byte swapping macros */
-#if defined(__sparc))                          /* big endian */
+#if defined(__sparc) || defined(__s390)                        /* big endian
*/
 #define        AUDIO_AIFF_FILE2HOST_INT(from, to)
\
                *((int *)(to)) = *((int *)(from))
 #define        AUDIO_AIFF_FILE2HOST_SHORT(from, to)
\
```

14.1.5 /usr/src/cmd/abi/apptracecmd/apptrace.c

```
--- a/usr/src/cmd/abi/apptracecmd/apptrace.c   Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/cmd/abi/apptracecmd/apptrace.c   Tue Sep 02 11:15:47 2008 -0400
@@ -85,7 +85,6 @@
 #elif defined(__s390) || defined(__s390x)
 static char *LD_AUDIT_64 =
       "LD_AUDIT_64=/usr/lib/abi/s390x/apptrace.so.1";
-#else
```

```
 #else
 #error Unsupported Platform
 #endif
```

14.1.6 /usr/src/cmd/agents/snmp/mib/Makefile

```
--- a/usr/src/cmd/agents/snmp/mib/Makefile     Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/cmd/agents/snmp/mib/Makefile     Tue Sep 02 11:15:47 2008 -0400
@@ -53,5 +53,6 @@

 _msg:

+all:

 include $(SRC)/cmd/Makefile.targ
```

14.1.7 /usr/src/cmd/agents/snmp/snmprelayd/Makefile

- It appears the program doesn't need `libl` to build or run.
- Using gnu linker requires changes to mapfile handling

```
--- a/usr/src/cmd/agents/snmp/snmprelayd/Makefile    Tue Aug 19 11:19:58 2008 -
0400
+++ b/usr/src/cmd/agents/snmp/snmprelayd/Makefile    Tue Sep 02 11:15:47 2008 -
0400
@@ -115,13 +115,17 @@
 MYLIBS=      ../agent/$(MACH)/libssagent.so.$(EXT)
../snmplib/$(MACH)/libssasnmp.so.$(EXT)
 MYLINKLIBS += -L../agent/$(MACH) -lssagent -L../snmplib -lssasnmp

-LIBS= -lsocket -lnsl -ll
+LIBS= -lsocket -lnsl
+#LIBS=       -lsocket -lnsl -ll

 # snmpdx has a name clash with main() and libl.so.1.  However, snmpdx must
 # still export a number of "yy*" (libl) interfaces, and interfaces for
 # libssagent.so.1.  Reduce all other symbols to local scope.
 MAPFILES += $(MAPFILE.INT) $(MAPFILE.LEX) $(MAPFILE.NGB)
-MAPOPTS =    $(MAPFILES:%=-M%)
+sparc_MAPOPTS = $(MAPFILES:%=-M%)
+i386_MAPOPTS =      $(MAPFILES:%=-M%)
+s390_MAPOPTS =      $(MAPFILES:%=-_gcc=--version-script=%)
+MAPOPTS =    $($(MACH)_MAPOPTS)

 CPPFLAGS += -I. -I../agent -I../snmplib -I../../include/netmgt -I${BIN}
 CFLAGS +=    -c -D$(TARG_SYS)
```

14.1.8 /usr/src/cmd/apt/msgcc/Makefile

- When cross-building it is necessary to look in `$ROOT/usr/include` rather than default to the building platform's `/usr/include`

```
--- a/usr/src/cmd/ast/msgcc/Makefile    Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/cmd/ast/msgcc/Makefile    Tue Sep 02 11:15:47 2008 -0400
@@ -38,6 +38,7 @@
 CPPFLAGS = \
         $(DTEXTDOM) $(DTS_ERRNO) \
         -I$(ROOT)/usr/include/ast \
+      -I$(ROOT)/usr/include      \
```

```
       -D_PACKAGE_ast \
       '-DUSAGE_LICENSE=\
       "[-author?Glenn Fowler <gsf@research.att.com>]"\
@@ -49,7 +50,9 @@
         $(CCVERBOSE) \
         -xstrconst

-LDLIBS += -last
+s390_LDLIBS=-lgcc_s
+LDLIBS += -last $($(MACH)_LDLIBS)
+
 msgcpp   := LDLIBS += -lpp

 msgcc:      msgcc.sh
```

14.1.9 /usr/src/cmd/audit/audit.c

- `getopt()` returns an integer. On some platforms char defaults to unsigned which means tests against -1 will fail (gcc will also flag a warning)

```
--- a/usr/src/cmd/audit/audit.c   Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/cmd/audit/audit.c   Tue Sep 02 11:15:47 2008 -0400
@@ -92,7 +92,7 @@
 {
       pid_t pid; /* process id of auditd read from auditdatafile */
       int    sig = 0; /* signal to send auditd */
-      char   c;
+      int    c;
       char   *first_option;

       /* Internationalization */
```

14.1.10 /usr/src/cmd/availdevs/Makefile

- When cross-building `libxml2` may not be installed on the base system and may be placed in `$ROOT`

```
--- a/usr/src/cmd/availdevs/Makefile    Tue Aug 19 11:19:58 2008 -0400
 +++ b/usr/src/cmd/availdevs/Makefile   Tue Sep 02 11:15:47 2008 -0400
@@ -32,7 +32,7 @@

 ROOTCMDDIR= $(ROOTLIB)/zfs

-INCS += -I/usr/include/libxml2
+INCS += -I$(ROOT)/usr/include/libxml2 -I/usr/include/libxml2

 #
 # There is no lint library for libxml2, so we need to avoid linking against
```

14.1.11 /usr/src/cmd/awk/Makefile

- Changed recipe to build maketab using sparc compiler. This has changed in a subsequent update to use a new global variable “XBUILDCC” which is being set in Makefile.master. The recipe returns to its original form except that it uses $(XBUILDCC) as the compiler.

```
diff -r 4f051ff1b998 usr/src/cmd/awk/Makefile
```

```
--- a/usr/src/cmd/awk/Makefile    Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/cmd/awk/Makefile    Tue Sep 02 11:15:47 2008 -0400
@@ -73,7 +73,11 @@
       rm -f $@; ./maketab > $@

 maketab: maketab.c
-      $(NATIVECC) -O maketab.c -o $@ $(LDLIBS)
+      @if [ $(MACH) = "s390" ]; then                        \
+            $(sparc_CC) -O maketab.c -o $@;                 \
+      else                                                  \
+            $(NATIVECC) -O maketab.c -o $@ $(LDLIBS);       \
+      fi

 install: all $(ROOTPROG) $(ROOTLINK)
```

```
--- a/usr/src/cmd/awk/Makefile  Tue Sep 02 11:22:56 2008 -0400
+++ b/usr/src/cmd/awk/Makefile  Thu Sep 25 10:26:06 2008 -0400
@@ -73,11 +73,7 @@
        rm -f $@; ./maketab > $@

 maketab: maketab.c
-       @if [ $(MACH) = "s390" ]; then                          \
-               $(sparc_CC) -O maketab.c -o $@;                 \
-       else                                                    \
-               $(NATIVECC) -O maketab.c -o $@ $(LDLIBS);       \
-       fi
+       $(XBUILDCC) -O maketab.c -o $@

 install: all $(ROOTPROG) $(ROOTLINK)
```

14.1.12 /usr/src/cmd/awk_xpg4/awk1.c

- Move static function prototypes from within routine to global area: gcc doesn't cope with prototypes declared this way.

```
--- a/usr/src/cmd/awk_xpg4/awk1.c Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/cmd/awk_xpg4/awk1.c Tue Sep 02 11:15:48 2008 -0400
@@ -72,7 +72,7 @@
 static void  awkierr(int perr, char *fmt, va_list ap);
 static int   usage(void);
 void         strescape(wchar_t *str);
-static const char  *toprint(wint_t);
+static const char  *toprint(wchar_t);
 char *_cmdname;
 static wchar_t *mbconvert(char *str);
```

14.1.13 /usr/src/cmd/backup/dump/Makefile

- gcc build requires `-ldl` and `-lgcc_s` so add a new variable `XLDLIBS` that is set on a per-platform basis via `<mach>_XLDLIBS`

```
--- a/usr/src/cmd/backup/dump/Makefile  Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/cmd/backup/dump/Makefile  Tue Sep 02 11:15:48 2008 -0400
@@ -71,8 +71,10 @@
              -D_LARGEFILE64_SOURCE=1
 LIBDUMP=     ../lib/libdump.a
 LINTLIBDUMP= ../lib/llib-ldump.ln -lkstat
+s390_XLDLIBS=       -ldl -lgcc_s
```

```
+XLDLIBS=     $($(MACH)_XLDLIBS)
 LDLIBS +=    $(BSTATIC) -L../lib -ldump $(BDYNAMIC) \
-             -lsocket -lnsl -ladm -lm -lkstat
+             -lsocket -lnsl -ladm -lm -lkstat $(XLDLIBS)

 UFSROOTDUMPDATES=  $(UFSROOTETC)/dumpdates
 UFSROOTLINK=       $(UFSROOTUSRSBIN)/$(PROG)
```

14.1.14 /usr/src/cmd/backup/restore/Makefile

- gcc build requires `-ldl` and `-lgcc_s` so add a new variable `XLDLIBS` that is set on a per-platform basis via `<mach>_XLDLIBS`

```
--- a/usr/src/cmd/backup/restore/Makefile      Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/cmd/backup/restore/Makefile      Tue Sep 02 11:15:48 2008 -0400
@@ -46,7 +46,9 @@
                     $(CPPFLAGS.master) -D_LARGEFILE64_SOURCE=1
 LIBDUMP=     ../lib/libdump.a
 LINTLIBDUMP= ../lib/llib-ldump.ln
-LDLIBS +=    $(BSTATIC) -L../lib -ldump $(BDYNAMIC) -lsocket -lnsl -lsec
+s390_XLDLIBS=      -ldl -lgcc_s
+XLDLIBS=     $($(MACH)_XLDLIBS)
+LDLIBS +=    $(BSTATIC) -L../lib -ldump $(BDYNAMIC) -lsocket -lnsl -lsec
$(XLDLIBS)

 UFSROOTLINK= $(UFSROOTUSRSBIN)/$(PROG)
 LINKVALUE=   ../lib/fs/$(FSTYPE)/$(PROG)
```

14.1.15 /usr/src/cmd/backup/restore/interactive.c

- There is a namespace clash for `lookupname` (it's defined in `<sys/pathname.h>`) so I renamed the one used by backup to `lookup_name`.

```
--- a/usr/src/cmd/backup/restore/interactive.c  Tue Sep 02 11:22:56 2008 -0400
+++ b/usr/src/cmd/backup/restore/interactive.c  Thu Sep 25 10:26:06 2008 -0400
@@ -138,7 +138,7 @@
                         goto bad;
                 if (name[0] == '\0')
                         break;
-                np = lookupname(name);
+                np = lookup_name(name);
                 if (np == NIL || (np->e_flags & NEW) == 0) {
                         (void) fprintf(stderr,
                                 gettext("%s: not on extraction list\n"), name);
@@ -626,7 +626,7 @@
         if (size == 0) {
                 struct entry *ep;

-                ep = lookupname(arg);
+                ep = lookup_name(arg);
                 single.fnum = ep ? ep->e_ino : 0;
                 single.fname = savename(arg);
                 ap->head = &single;
```

14.1.16 /usr/src/cmd/backup/restore/restore.c

- There is a namespace clash for `lookupname` (it's defined in `<sys/pathname.h>`) so I renamed the one used by backup to `lookup_name`.

```
--- a/usr/src/cmd/backup/restore/restore.c      Tue Sep 02 11:22:56 2008 -0400
+++ b/usr/src/cmd/backup/restore/restore.c      Thu Sep 25 10:26:06 2008 -0400
@@ -222,7 +222,7 @@
         /*
          * Check to see if the name exists, and if the name is a link.
          */
-        np = lookupname(name);
+        np = lookup_name(name);
         if (np != NIL) {
                 key |= NAMEFND;
                 ip = lookupino(np->e_ino);
@@ -935,7 +935,7 @@
         struct entry *np, *ep;
         long descend = GOOD;

-        ep = lookupname(name);
+        ep = lookup_name(name);
         if (ep == NIL) {
                 (void) fprintf(stderr,
                     gettext("Warning: missing name %s\n"), name);
```

14.1.17 /usr/src/cmd/backup/restore/restore.h

- There is a namespace clash for `lookupname` (it's defined in `<sys/pathname.h>`) so I renamed the one used by backup to `lookup_name`.

```
--- a/usr/src/cmd/backup/restore/restore.h      Tue Sep 02 11:22:56 2008 -0400
+++ b/usr/src/cmd/backup/restore/restore.h      Thu Sep 25 10:26:06 2008 -0400
@@ -124,7 +124,7 @@
  */
 #ifdef __STDC__
 extern struct entry *lookupino(ino_t);
-extern struct entry *lookupname(char *);
+extern struct entry *lookup_name(char *);
 extern struct entry *addentry(char *, ino_t, int);
 extern void deleteino(ino_t);
 extern char *myname(struct entry *);
```

14.1.18 /usr/src/cmd/backup/restore/symtab.c

- There is a namespace clash for `lookupname` (it's defined in `<sys/pathname.h>`) so I renamed the one used by backup to `lookup_name`.

```
--- a/usr/src/cmd/backup/restore/symtab.c       Tue Sep 02 11:22:56 2008 -0400
+++ b/usr/src/cmd/backup/restore/symtab.c       Thu Sep 25 10:26:06 2008 -0400
@@ -143,7 +143,7 @@
  *     NULL characters.
  */
```

```
 struct entry *
-lookupname(name)
+lookup_name(name)
        char *name;
 {
        struct entry *ep;
@@ -152,7 +152,7 @@

        if (strlen(name) > (sizeof (buf) - 1)) {
                (void) fprintf(stderr, gettext("%s: ignoring too-long name\n"),
-                   "lookupname");
+                   "lookup_name");
                return (NIL);
        }

@@ -215,7 +215,7 @@
        }
        savechar = *(tailindex+1);
        *(tailindex+1) = '\0';
-       ep = lookupname(name);
+       ep = lookup_name(name);
        if (ep != NIL && !xattrparent && ep->e_type != NODE)
                panic(gettext("%s is not a directory\n"), name);
        if (!xattrparent) *tailindex = '/';
```

14.1.19 /usr/src/cmd/backup/restore/tape.c

- There is a namespace clash for `lookupname` (it's defined in `<sys/pathname.h>`) so I renamed the one used by backup to `lookup_name`.

```
-- a/usr/src/cmd/backup/restore/tape.c Tue Sep 02 11:22:56 2008 -0400
+++ b/usr/src/cmd/backup/restore/tape.c Thu Sep 25 10:26:06 2008 -0400
@@ -837,7 +837,7 @@

        case IFDIR:
                if (mflag) {
-                       ep = lookupname(name);
+                       ep = lookup_name(name);
                        if (ep == NIL || ep->e_flags & EXTRACT) {
                                panic(gettext(
                                    "directory %s was not restored\n"),
```

14.1.20 /usr/src/cmd/backup/restore/utilitites.c

- There is a namespace clash for `lookupname` (it's defined in `<sys/pathname.h>`) so I renamed the one used by backup to `lookup_name`.

```
--- a/usr/src/cmd/backup/restore/utilities.c    Tue Sep 02 11:22:56 2008 -0400
+++ b/usr/src/cmd/backup/restore/utilities.c    Thu Sep 25 10:26:06 2008 -0400
@@ -25,7 +25,7 @@

 /*
  * Insure that all the components of a pathname exist. Note that
- * lookupname() and addentry() both expect complex names as
+ * lookup_name() and addentry() both expect complex names as
  * input arguments, so a double NULL needs to be added to each name.
  */
```

```
 void
@@ -44,7 +44,7 @@
                 *cp = '\0';
                 save = *(cp+1);
                 *(cp+1) = '\0';
-                ep = lookupname(name);
+                ep = lookup_name(name);
                 if (ep == NIL) {
                         ep = addentry(name, psearch(name), NODE);
                         newnode(ep);
```

14.1.21 /usr/src/cmd/bnu/grades.c

- Move static function prototypes from within routine to global area: gcc doesn't cope with prototypes declared this way.

```
--- a/usr/src/cmd/bnu/grades.c  Tue Sep 02 11:22:56 2008 -0400
+++ b/usr/src/cmd/bnu/grades.c  Thu Sep 25 10:26:06 2008 -0400
@@ -49,6 +49,7 @@

 extern int rdfulline(), jsize(), gdirf(), gnamef();
 extern void wfcommit();
+static void lcase();

 static void    mailAdmin();            /* Send mail to administrator. */

@@ -336,7 +337,6 @@
 #define ONE_K (1024)
 #define ONE_MEG ((1024)*(1024))

-       static void lcase();
        char rest[SMBUF];
        char msg[BUFSIZ], *p;

@@ -553,7 +553,6 @@
 #define G_GRP "group"
 #define G_NGRP "non-group"

-       static void lcase();
        char actn[SMBUF];
        char ufld[SMBUF];
        char msg[BUFSIZ];
```

14.1.22 /usr/src/cmd/bnu/uucico.c

- Move static function prototypes from within routine to global area: gcc doesn't cope with prototypes declared this way.

```
--- a/usr/src/cmd/bnu/uucico.c  Tue Sep 02 11:22:56 2008 -0400
+++ b/usr/src/cmd/bnu/uucico.c  Thu Sep 25 10:26:06 2008 -0400
@@ -77,6 +77,7 @@
 static void checkrmt();                /* See if we want to talk to remote. */
 #endif /* NOSTRANGERS */

+static char *pskip();
 extern char *Mytype;

 int
@@ -87,7 +88,6 @@
```

```
 {

        extern void intrEXIT(), onintr(), timeout();
-       static char *pskip();
        extern void setservice();
 #ifndef ATTSVR3
        void setTZ();
```

14.1.23 /usr/src/cmd/bnu/uucp.c

- Move static function prototypes from within routine to global area: gcc doesn't cope with prototypes declared this way.

```
--- a/usr/src/cmd/bnu/uucp.c    Tue Sep 02 11:22:56 2008 -0400
+++ b/usr/src/cmd/bnu/uucp.c    Thu Sep 25 10:26:06 2008 -0400
@@ -51,6 +51,7 @@
 int eaccess(), guinfo(), vergrd(), gwd(), ckexpf(), uidstat(), uidxcp(),
        copy(), gtcfile();
 void commitall(), wfabort(), mailst(), gename(), svcfile();
+static FILE *syscfile();

 char   Sfile[MAXFULLNAME];

@@ -485,7 +486,6 @@
 char *s1, *f1, *s2, *f2;
 {
        FILE *cfp;
-       static FILE *syscfile();
        struct stat stbuf, stbuf1;
        int type, statret;
        char dfile[NAMESIZE];
```

14.1.24 /usr/src/cmd/bnu/uux.c

- Move static function prototypes from within routine to global area: gcc doesn't cope with prototypes declared this way.

```
--- a/usr/src/cmd/bnu/uux.c     Tue Sep 02 11:22:56 2008 -0400
+++ b/usr/src/cmd/bnu/uux.c     Thu Sep 25 10:26:06 2008 -0400
@@ -61,6 +61,7 @@
 char Sgrade[NAMESIZE];
 void cleanup();
 static void usage();
+static void onintr();

 /*
  *     uux
@@ -73,7 +74,6 @@
 {
        char *jid();
        FILE *fprx = NULL, *fpc = NULL, *fpd = NULL, *fp = NULL;
-       static void onintr();
        int cfileUsed = 0;      /*  >0 if commands put in C. file flag  */
        int cflag = 0;          /* if > 0 make local copy of files to be sent
*/
        int nflag = 0;          /* if != 0, do not request error notification
*/
```

14.1.25 /usr/src/cmd/boot/bootadm/bootadm.h

- Add System z support.

```
--- a/usr/src/cmd/boot/bootadm/bootadm.h        Tue Sep 02 11:22:56 2008 -0400
+++ b/usr/src/cmd/boot/bootadm/bootadm.h        Thu Sep 25 10:26:06 2008 -0400
@@ -244,6 +244,7 @@
 #define        DIRECT_BOOT_ARCHIVE     "/platform/s390x/boot_archive"
 #define        DIRECT_BOOT_ARCHIVE_32  "/platform/s390x/boot_archive"
 #define        DIRECT_BOOT_ARCHIVE_64  "/platform/s390x/boot_archive"
+#define        FAILSAFE_ARCHIVE        "/boot/s390x.miniroot-safe"
 #endif
 #define        MULTIBOOT_ARCHIVE       DIRECT_BOOT_ARCHIVE_32
```

14.1.26 /usr/src/cmd/cmd-crypt/cryptoadm/cryptoadm.c

- `getopt()` returns an integer. On some platforms char defaults to unsigned which means tests against -1 will fail (gcc will also flag a warning)

```
--- a/usr/src/cmd/cmd-crypto/cryptoadm/cryptoadm.c      Tue Sep 02 11:22:56
2008 -0400
+++ b/usr/src/cmd/cmd-crypto/cryptoadm/cryptoadm.c      Thu Sep 25 10:26:06
2008 -0400
@@ -539,7 +539,7 @@
        boolean_t       mflag = B_FALSE;
        boolean_t       pflag = B_FALSE;
        boolean_t       vflag = B_FALSE;
-       char    ch;
+       int     ch;
        cryptoadm_provider_t    *prov = NULL;
        int     rc = SUCCESS;
```

14.1.27 /usr/src/cmd/cmd-crypto/decrypt/decrypt.c

- `getopt()` returns an integer. On some platforms char defaults to unsigned which means tests against -1 will fail (gcc will also flag a warning)

```
--- a/usr/src/cmd/cmd-crypto/decrypt/decrypt.c Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/cmd/cmd-crypto/decrypt/decrypt.c Tue Sep 02 11:15:48 2008 -0400
@@ -193,7 +193,7 @@
       extern char *optarg;
       extern int optind;
       char *optstr;
-      char c;                  /* current getopts flag */
+      int c;              /* current getopts flag */
       char *algo_str = NULL;     /* algorithm string */
       struct CommandInfo *cmd;
       char *cmdname;             /* name of command */
```

14.1.28 /usr/src/cmd/cmd-crypto/digest/digest.c

- `getopt()` returns an integer. On some platforms char defaults to unsigned which means tests against -1 will fail (gcc will also flag a warning)

```
--- a/usr/src/cmd/cmd-crypto/digest/digest.c   Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/cmd/cmd-crypto/digest/digest.c   Tue Sep 02 11:15:48 2008 -0400
@@ -132,7 +132,7 @@
       extern char *optarg;
       extern int optind;
       int errflag = 0;    /* We had an optstr parse error */
-      char c;                   /* current getopts flag */
+      int c;              /* current getopts flag */
       char *algo_str;           /* mechanism/algorithm string */
       int filecount;
       boolean_t mac_cmd;  /* if TRUE, do mac, else do digest */
```

14.1.29 /usr/src/cmd/cmd-crypto/elfsign/elfsign.c

- `getopt()` returns an integer. On some platforms char defaults to unsigned which means tests against -1 will fail (gcc will also flag a warning)

```
--- a/usr/src/cmd/cmd-crypto/elfsign/elfsign.c  Tue Sep 02 11:22:56 2008 -0400
+++ b/usr/src/cmd/cmd-crypto/elfsign/elfsign.c  Thu Sep 25 10:26:06 2008 -0400
@@ -147,7 +147,7 @@
        char *scmd = NULL;
        char *opts;             /* The set of flags for cmd */
        int errflag = 0;        /* We had an options parse error */
-       char c;                 /* current getopts flag */
+       int c;                  /* current getopts flag */
        ret_t (*action)(char *);        /* Function pointer for the action */
        ret_t ret;
```

14.1.30 /usr/src/cmd/cmd-crypto/kmfcfg/Makefile

- When cross-building `libxml2` may not be installed on the base system and may be placed in `$ROOT`

```
--- a/usr/src/cmd/cmd-crypto/kmfcfg/Makefile   Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/cmd/cmd-crypto/kmfcfg/Makefile   Tue Sep 02 11:15:48 2008 -0400
@@ -49,7 +49,7 @@
 POFILE              = $(PROG)_msg.po
 MSGFILES     = $(SRCS:%.c=%.i)

-CPPFLAGS     += -I/usr/include/libxml2 -I$(KMFDIR)/include -I.
+CPPFLAGS     += -I$(ROOT)/usr/include/libxml2 -I/usr/include/libxml2 -
I$(KMFDIR)/include -I.
 LDLIBS              += -L$(ROOT)/usr/lib -lkmf -lcryptoutil
 XMLLIB              = -lxml2
```

14.1.31 /usr/src/cmd/cmd-crypto/pktool/Makefile

- When cross-building `libxml2` may not be installed on the base system and may be placed in `$ROOT`

```
--- a/usr/src/cmd/cmd-crypto/pktool/Makefile    Tue Sep 02 11:22:56 2008 -0400
+++ b/usr/src/cmd/cmd-crypto/pktool/Makefile    Thu Sep 25 10:26:06 2008 -0400
@@ -48,7 +48,7 @@
 POFILE = $(PROG)_msg.po
 MSGFILES=$(SRCS:%.c=%.i)
```

```
-CPPFLAGS += -I. -I$(KMFDIR)/include -$(ROOT)/usr/include/libxml2 -
I/usr/include/libxml2
+CPPFLAGS += -I. -I$(KMFDIR)/include -I$(ROOT)/usr/include/libxml2 -
I/usr/include/libxml2
 CFLAGS += $(CCVERBOSE) -DDEBUG

 LDFLAGS += -L$(SRC)/lib/libkmf/libkmf/$(MACH)
```

14.1.32 /usr/src/cmd/cmd-inet/sbin/dhcpagent/request.c

- Cast parameter to avoid warning.

```
--- a/usr/src/cmd/cmd-inet/sbin/dhcpagent/request.c     Tue Sep 02 11:22:56
2008 -0400
+++ b/usr/src/cmd/cmd-inet/sbin/dhcpagent/request.c     Thu Sep 25 10:26:06
2008 -0400
@@ -1033,7 +1033,7 @@
        if (!isv6) {
                sock = dsmp->dsm_lif->lif_sock_ip_fd;

-               if (getsockname(sock, (struct sockaddr *)&sin, &sinlen) != -1
&&
+               if (getsockname(sock, (struct sockaddr *)&sin, (Psocklen_t)
&sinlen) != -1 &&
                    sin.sin_addr.s_addr == INADDR_ANY) {
                        reason = "handled by lif_sock_ip_fd";
                        goto drop;
```

14.1.33 /usr/src/cmd/cmd-inet/usr.bin/nca/Makefile

- Add libraries for System z build.

```
--- a/usr/src/cmd/cmd-inet/usr.bin/nca/Makefile Tue Sep 02 11:22:56 2008 -0400
+++ b/usr/src/cmd/cmd-inet/usr.bin/nca/Makefile Thu Sep 25 10:26:06 2008 -0400
@@ -48,6 +48,8 @@
 INC_PATH +=    ../../../../uts/common/inet/nca -I.
 CFLAGS +=      $(CCVERBOSE)
 CPPFLAGS +=    -I$(INC_PATH)
+s390_LDLIBS =  -L$(ROOT)/usr/lib -lc -ldl
+LDLIBS =       $($(MACH)_LDLIBS)

 .KEEP_STATE:

@@ -56,7 +58,7 @@
 all: $(PROG)

 $(PROG): $(OBJS)
-       $(LINK.c) $(OBJS) -o $@
+       $(LINK.c) $(OBJS) -o $@ $(LDLIBS)
        $(POST_PROCESS)

 #
```

14.1.34 /usr/src/cmd/cmd-inet/usr.lib/mdnsd/uds_daemon.c

- Cast parameter to avoid warning.

```
diff -r 6b9772e86b00 usr/src/cmd/cmd-inet/usr.lib/mdnsd/uds_daemon.c
```

```
--- a/usr/src/cmd/cmd-inet/usr.lib/mdnsd/uds_daemon.c   Tue Sep 02 11:22:56
2008 -0400
+++ b/usr/src/cmd/cmd-inet/usr.lib/mdnsd/uds_daemon.c   Thu Sep 25 10:26:06
2008 -0400
@@ -1259,7 +1259,7 @@

        len = (dnssd_socklen_t) sizeof(cliaddr);

-       sd = accept(listenfd, (struct sockaddr*) &cliaddr, &len);
+       sd = accept(listenfd, (struct sockaddr*) &cliaddr, (Psocklen_t) &len);

     if (sd == dnssd_InvalidSocket)
         {
```

14.1.35 /usr/src/cmd/cmd-inet/usr.lib/wanboot/netbootinfo/Makefile

- Ensure System z searches proto area for `libcrypto`.

```
--- a/usr/src/cmd/cmd-inet/usr.lib/wanboot/netbootinfo/Makefile Tue Sep 02
11:22:56 2008 -0400
+++ b/usr/src/cmd/cmd-inet/usr.lib/wanboot/netbootinfo/Makefile Thu Sep 25
10:26:06 2008 -0400
@@ -33,7 +33,9 @@

 # The OpenSSL libraries need to be linked against in order to resolve
 # references made to them by libwanboot.
-LDLIBS +=      -lwanbootutil -lwanboot $(OPENSSL_LDFLAGS)
+s390_XLDLIBS=  -lssl -L$(ROOT)/usr/lib -lcrypto
+XLDLIBS=       $($(MACH)_XLDLIBS)
+LDLIBS +=      -lwanbootutil -lwanboot $(OPENSSL_LDFLAGS) $(XLDLIBS)
 CPPFLAGS +=     -I$(CMNCRYPTDIR)

 all:           $(PROG)
```

14.1.36 /usr/src/cmd/cmd-inet/usr.lib/wanboot/pl2split/Makefile

- Ensure System z searches proto area for `libcrypto`.

```
--- a/usr/src/cmd/cmd-inet/usr.lib/wanboot/p12split/Makefile    Tue Sep 02
11:22:56 2008 -0400
+++ b/usr/src/cmd/cmd-inet/usr.lib/wanboot/p12split/Makefile    Thu Sep 25
10:26:06 2008 -0400
@@ -30,7 +30,9 @@
 include $(SRC)/lib/openssl/Makefile.openssl

 PROG=          p12split
-LDLIBS +=      -lwanboot -linetutil -lwanbootutil $(OPENSSL_LDFLAGS) -lcrypto
+s390_XLDLIBS=  -lssl -L$(ROOT)/usr/lib
+XLDLIBS=       $($(MACH)_XLDLIBS) -lcrypto
+LDLIBS +=      -lwanboot -linetutil -lwanbootutil $(OPENSSL_LDFLAGS)
$(XLDLIBS)
 LDFLAGS +=     $(OPENSSL_DYNFLAGS)
 CPPFLAGS =     $(OPENSSL_CPPFLAGS) -I$(CMNCRYPTDIR) $(CPPFLAGS.master)
```

14.1.37 /usr/src/cmd/cmd-inet/usr.lib/wanboot/wanboot-cgi/Makefile

- Ensure System z searches proto area for `libcrypto`.

```
--- a/usr/src/cmd/cmd-inet/usr.lib/wanboot/wanboot-cgi/Makefile Tue Sep 02
11:22:56 2008 -0400
+++ b/usr/src/cmd/cmd-inet/usr.lib/wanboot/wanboot-cgi/Makefile Thu Sep 25
10:26:06 2008 -0400
@@ -31,7 +31,7 @@

 PROG =          wanboot-cgi
 LDLIBS +=       -lgen -lnsl -lwanbootutil -lnvpair -lwanboot \
-                $(OPENSSL_LDFLAGS) -lcrypto
+                $(OPENSSL_LDFLAGS) -lcrypto -lssl
 LDFLAGS +=      $(OPENSSL_DYNFLAGS)
 CPPFLAGS =      $(OPENSSL_CPPFLAGS) -I$(CMNCRYPTDIR) $(CPPFLAGS.master)
```

14.1.38 /usr/src/cmd/cmd-inet/usr.lib/wpad/Makefile

- Ensure System z searches proto area for `libcrypto`.

```
-- a/usr/src/cmd/cmd-inet/usr.lib/wpad/Makefile        Tue Sep 02 11:22:56 2008
-0400
+++ b/usr/src/cmd/cmd-inet/usr.lib/wpad/Makefile        Thu Sep 25 10:26:06
2008 -0400
@@ -35,11 +35,11 @@

 ROOTMANIFESTDIR = $(ROOTSVCNETWORK)

-LDFLAGS   +=   -L/usr/sfw/lib -R/usr/sfw/lib
+LDFLAGS   +=   -L$(ROOT)/usr/sfw/lib -L/usr/sfw/lib -R/usr/sfw/lib
 LDLIBS    +=   -ldladm -ldlpi
 all install := LDLIBS += -lcrypto

-CPPFLAGS  +=   -I/usr/sfw/include
+CPPFLAGS  +=   -I$(ROOT)/usr/sfw/include -I/usr/sfw/include
 LINTFLAGS +=   -u

 .KEEP_STATE:
```

14.1.39 /usr/src/cmd/cmd-inet/usr/sbin/bootconfchk/Makefile

- Ensure System z searches proto area for `libcrypto`.

```
--- a/usr/src/cmd/cmd-inet/usr.sbin/bootconfchk/Makefile        Tue Sep 02
11:22:56 2008 -0400
+++ b/usr/src/cmd/cmd-inet/usr.sbin/bootconfchk/Makefile        Thu Sep 25
10:26:06 2008 -0400
@@ -33,7 +33,9 @@

 # Need to be told where the OpenSSL libraries are because libwanboot is
 # linked to them and they are in a non standard place.
-LDLIBS +=       -lwanbootutil -lwanboot $(OPENSSL_LDFLAGS)
+s390_XLDLIBS=   -lssl -L$(ROOT)/usr/lib -lcrypto
+XLDLIBS=        $($(MACH)_XLDLIBS)
+LDLIBS +=       -lwanbootutil -lwanboot $(OPENSSL_LDFLAGS) $(XLDLIBS)

 CPPFLAGS +=     -I$(SRC)/common/net/wanboot/crypt
```

14.1.40 /usr/src/cmd/cmd-inet/usr.sbin/ifconfig/revarp.c

- Cast parameter to avoid gcc warning.

```
--- a/usr/src/cmd/cmd-inet/usr.sbin/ifconfig/revarp.c   Tue Sep 02 11:22:56
2008 -0400
+++ b/usr/src/cmd/cmd-inet/usr.sbin/ifconfig/revarp.c   Thu Sep 25 10:26:06
2008 -0400
@@ -363,7 +363,7 @@
                return;
        }

-       retv = dlpi_get_physaddr(dh, DL_CURR_PHYS_ADDR, physaddr,
&physaddrlen);
+       retv = dlpi_get_physaddr(dh, DL_CURR_PHYS_ADDR, physaddr, (size_t *)
&physaddrlen);
        if (retv != DLPI_SUCCESS) {
                Perrdlpi("dlpi_get_physaddr failed", linkname, retv);
                dlpi_close(dh);
```

14.1.41 /usr/src/cmd/cmd-inet/usr.sbin/in.ftpd/Makefile

- Ensure System z searches proto area for its libraries.

```
--- a/usr/src/cmd/cmd-inet/usr.sbin/in.ftpd/Makefile    Tue Sep 02 11:22:56
2008 -0400
+++ b/usr/src/cmd/cmd-inet/usr.sbin/in.ftpd/Makefile    Thu Sep 25 10:26:06
2008 -0400
@@ -49,6 +49,8 @@
                -DSOLARIS_GSS_USEROK -DSOLARIS_PRIVS
 LDLIBS +=      -lsocket -lnsl -lpam -lbsm -lsendfile -lgss
 YFLAGS +=      -d
+s390_LDDIR=    -L$(ROOT)/usr/lib -lmd -lmp -lscf -luutil -ldl -lgen
+LDDIR=         $($(MACH)_LDDIR)

 in.ftpd :=     LDFLAGS += $(MAPFILE.NGB:%=-M%)

@@ -68,7 +70,7 @@
                $(POST_PROCESS)

 ftpcount:      $(FTPCOUNT_OBJS)
-               $(LINK.c) $(FTPCOUNT_OBJS) -o $@ -lsocket -lnsl
+               $(LINK.c) $(FTPCOUNT_OBJS) -o $@ $(LDDIR) -lsocket -lnsl
                $(POST_PROCESS)

 ftpwho:                ftpcount
@@ -76,19 +78,19 @@
                $(LN) ftpcount $@

 ftpshut:       $(FTPSHUT_OBJS)
-               $(LINK.c) $(FTPSHUT_OBJS) -o $@ -lsocket -lnsl
+               $(LINK.c) $(FTPSHUT_OBJS) -o $@ $(LDDIR) -lsocket -lnsl
                $(POST_PROCESS)

 ftprestart:    $(FTPREST_OBJS)
-               $(LINK.c) $(FTPREST_OBJS) -o $@ -lsocket -lnsl
+               $(LINK.c) $(FTPREST_OBJS) -o $@ $(LDDIR) -lsocket -lnsl
                $(POST_PROCESS)

 ckconfig:      $(CKCONFIG_OBJS)
-               $(LINK.c) $(CKCONFIG_OBJS) -o $@ -lsocket -lnsl
```

```
+               $(LINK.c) $(CKCONFIG_OBJS) -o $@ $(LDDIR) -lsocket -lnsl
                $(POST_PROCESS)

 privatepw:     $(PRIVATE_OBJS)
-               $(LINK.c) $(PRIVATE_OBJS) -o $@
+               $(LINK.c) $(PRIVATE_OBJS) -o $@ $(LDDIR)
                $(POST_PROCESS)

# This causes y.tab.c to be renamed to ftpcmd.c, needed by tcov.
```

14.1.42 /usr/src/cmd/cmd-inet/usr.sbin/in.rarpd.c

- Cast parameter to avoid gcc warning.

```
--- a/usr/src/cmd/cmd-inet/usr.sbin/in.rarpd.c  Tue Sep 02 11:22:56 2008 -0400
+++ b/usr/src/cmd/cmd-inet/usr.sbin/in.rarpd.c  Thu Sep 25 10:26:06 2008 -0400
@@ -429,7 +429,7 @@
         * Save our mac address.
         */
        if ((retval = dlpi_get_physaddr(dh, DL_CURR_PHYS_ADDR, rdev->physaddr,
-           &physaddrlen)) != DLPI_SUCCESS) {
+           (size_t *) &physaddrlen)) != DLPI_SUCCESS) {
                dlpi_close(dh);
                error("dlpi_get_physaddr failed: %s", dlpi_strerror(retval));
        }
@@ -509,7 +509,7 @@
        for (;;) {
                saddrlen = DLPI_PHYSADDR_MAX;
                retval = dlpi_recv(rdev->dh_rarp, shost,
-                   &saddrlen, ans, &anslen, -1, NULL);
+                   (size_t *) &saddrlen, ans, &anslen, -1, NULL);
                if (retval == DLPI_ETIMEDOUT) {
                        continue;
                } else if (retval != DLPI_SUCCESS) {
```

14.1.43 /usr/src/cmd/cmd-inet/usr/sbin/in.rlogind.c

- `getopt()` returns an integer. On some platforms char defaults to unsigned which means tests against -1 will fail (gcc will also flag a warning)

```
--- a/usr/src/cmd/cmd-inet/usr.sbin/in.rlogind.c        Tue Sep 02 11:22:56
2008 -0400
+++ b/usr/src/cmd/cmd-inet/usr.sbin/in.rlogind.c        Thu Sep 25 10:26:07
2008 -0400
@@ -188,7 +188,7 @@
        int fd = -1;

        extern char *optarg;
-       char c;
+       int c;
        int tos = -1;
        krb5_context krb_context;
        krb5_keytab keytab = NULL;
```

14.1.44 /usr/src/cmd/cmd-net/usr.sbin/in.routed/if.c

- Correct prototypes to match function declaration to avoid gcc warnings.

```
--- a/usr/src/cmd/cmd-inet/usr.sbin/in.routed/if.c      Tue Sep 02 11:22:56
2008 -0400
+++ b/usr/src/cmd/cmd-inet/usr.sbin/in.routed/if.c      Thu Sep 25 10:26:07
2008 -0400
@@ -99,9 +99,9 @@
 static void            if_bad(struct interface *, boolean_t);
 static boolean_t       addrouteforif(struct interface *);
 static int     get_if_kstats(struct interface *, struct phyi_data *);
-static uint_t  ahash(const void *, uint_t);
-static uint_t  ihash(const void *, uint_t);
-static uint_t  nhash(const void *, uint_t);
+static uint_t  ahash(const void *, size_t);
+static uint_t  ihash(const void *, size_t);
+static uint_t  nhash(const void *, size_t);
 static void    htbl_grow(struct htbl *);

 /*
```

14.1.45 /usr/src/cmd/cmd-inet/usr.sbin/in.telnetd.c

- `getopt()` returns an integer. On some platforms char defaults to unsigned which means tests against -1 will fail (gcc will also flag a warning)

```
--- a/usr/src/cmd/cmd-inet/usr.sbin/in.telnetd.c     Tue Aug 19 11:19:58 2008 -
0400
+++ b/usr/src/cmd/cmd-inet/usr.sbin/in.telnetd.c     Tue Sep 02 11:15:48 2008 -
0400
@@ -2200,7 +2200,7 @@
       boolean_t standalone = 0;
 #endif /* defined(DEBUG) */
       extern char *optarg;
-      char c;
+      int c;
       int tos = -1;

       while ((c = getopt(argc, argv, TELNETD_OPTS DEBUG_OPTS)) != -1) {
```

14.1.46 /usr/src/cmd/devfsadm/devfsadm.c

- `getopt()` returns an integer. On some platforms char defaults to unsigned which means tests against -1 will fail (gcc will also flag a warning)

```
--- a/usr/src/cmd/devfsadm/devfsadm.c   Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/cmd/devfsadm/devfsadm.c   Tue Sep 02 11:15:48 2008 -0400
@@ -521,7 +521,7 @@
 static void
 parse_args(int argc, char *argv[])
 {
-      char opt;
+      int opt;
       char get_linkcompat_opts = FALSE;
       char *compat_class;
       int num_aliases = 0;
```

14.1.47 /usr/srcmd/cmd/dfs.cmds/sharemgr/Makefile.com

- When cross-building `libxml2` may not be installed on the base system and may be placed in `$ROOT`

```
--- a/usr/src/cmd/dfs.cmds/sharemgr/Makefile.com     Tue Aug 19 11:19:58 2008 -
0400
+++ b/usr/src/cmd/dfs.cmds/sharemgr/Makefile.com     Tue Sep 02 11:15:48 2008 -
0400
@@ -41,7 +41,8 @@

 ROOTLINKS = $(ROOTUSRSBIN)/share $(ROOTUSRSBIN)/unshare

-MYCPPFLAGS = -I../../../../lib/libfsmgt/common -I/usr/include/libxml2 \
+MYCPPFLAGS = -I../../../../lib/libfsmgt/common \
+             -I$(ROOT)/usr/include/libxml2 -I/usr/include/libxml2 \
              -I../..
 CPPFLAGS += $(MYCPPFLAGS)
 LDLIBS += -lshare -lscf -lsecdb -lumem
```

14.1.48 /usr/src/cmd/dladm/dladm.c

- `getopt()` returns an integer. On some platforms char defaults to unsigned which means tests against -1 will fail (gcc will also flag a warning)

```
--- a/usr/src/cmd/dladm/dladm.c   Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/cmd/dladm/dladm.c   Tue Sep 02 11:15:48 2008 -0400
@@ -874,7 +874,7 @@
 static void
 do_create_aggr(int argc, char *argv[], const char *use)
 {
-       char                option;
+       int                 option;
        int                 key = 0;
        uint32_t            policy = AGGR_POLICY_L4;
        aggr_lacp_mode_t    lacp_mode = AGGR_LACP_OFF;
@@ -1048,7 +1048,7 @@
 static void
 do_delete_aggr(int argc, char *argv[], const char *use)
 {
-       char                option;
+       int                 option;
        char                *altroot = NULL;
        uint32_t            flags = DLADM_OPT_ACTIVE | DLADM_OPT_PERSIST;
        dladm_status_t              status;
@@ -1089,7 +1089,7 @@
 static void
 do_add_aggr(int argc, char *argv[], const char *use)
 {
-       char                option;
+       int                 option;
        uint_t              n, ndev, nlink;
        char                *altroot = NULL;
        uint32_t            flags = DLADM_OPT_ACTIVE | DLADM_OPT_PERSIST;
@@ -1185,7 +1185,7 @@
 static void
 do_remove_aggr(int argc, char *argv[], const char *use)
 {
-       char                        option;
```

```
+       int                             option;
        dladm_aggr_port_attr_db_t       port[MAXPORT];
        uint_t                          n, ndev, nlink;
        char                            *devs[MAXPORT];
@@ -1261,7 +1261,7 @@
 static void
 do_modify_aggr(int argc, char *argv[], const char *use)
 {
-       char                    option;
+       int                     option;
        uint32_t                policy = AGGR_POLICY_L4;
        aggr_lacp_mode_t        lacp_mode = AGGR_LACP_OFF;
        aggr_lacp_timer_t       lacp_timer = AGGR_LACP_TIMER_SHORT;
@@ -1388,7 +1388,7 @@
        uint_t          ppa;
        datalink_id_t   linkid;
        int             vid = 0;
-       char            option;
+       int             option;
        uint32_t        flags = (DLADM_OPT_ACTIVE | DLADM_OPT_PERSIST);
        char            *altroot = NULL;
        char            vlan[MAXLINKNAMELEN];
@@ -1467,7 +1467,7 @@
 static void
 do_delete_vlan(int argc, char *argv[], const char *use)
 {
-       char            option;
+       int             option;
        uint32_t        flags = (DLADM_OPT_ACTIVE | DLADM_OPT_PERSIST);
        char            *altroot = NULL;
        datalink_id_t   linkid;
@@ -1539,7 +1539,7 @@
 static void
 do_rename_link(int argc, char *argv[], const char *use)
 {
-       char            option;
+       int             option;
        char            *link1, *link2;
        char            *altroot = NULL;
        dladm_status_t      status;
```

14.1.49 /usr/src/cmd/fm/modules/common/fabric-xlate/Makefile

- When cross-building `libxml2` may not be installed on the base system and may be placed in `$ROOT`

```
--- a/usr/src/cmd/fm/modules/common/fabric-xlate/Makefile   Tue Aug 19 11:19:58
2008 -0400
+++ b/usr/src/cmd/fm/modules/common/fabric-xlate/Makefile   Tue Sep 02 11:15:48
2008 -0400
@@ -31,7 +31,7 @@

 include ../../Makefile.plugin

-CPPFLAGS += -I/usr/include/libxml2 -I$(KMFDIR)/include -I.
+CPPFLAGS += -I$(ROOT)/usr/include/libxml2 -I/usr/include/libxml2 -
I$(KMFDIR)/include -I.
 INCDIRS = $(SRC)/uts/common
 CFLAGS += -I$(INCDIRS)
 LINTFLAGS += -I$(INCDIRS)
```

14.1.50 /usr/src/cmd/fuser/fuser.c

- `getopt()` returns an integer. On some platforms char defaults to unsigned which means tests against -1 will fail (gcc will also flag a warning)

```
--- a/usr/src/cmd/fuser/fuser.c   Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/cmd/fuser/fuser.c   Tue Sep 02 11:15:48 2008 -0400
@@ -452,7 +452,8 @@
 main(int argc, char **argv)
 {
       fu_data_t     *fu_data;
-      char          *mntname, c;
+      char          *mntname;
+      int           c;
       int           newfile = 0, errors = 0, opts = 0, flags = 0;
       int           uts_flags, sig, okay, err;
```

14.1.51 /usr/src/cmd/hal/Makefile.hal

- When cross-building `dbus` and `glib` may not be installed on the base system and may be placed in `$ROOT`

```
--- a/usr/src/cmd/hal/Makefile.hal      Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/cmd/hal/Makefile.hal      Tue Sep 02 11:15:48 2008 -0400
@@ -55,9 +55,11 @@
                     -DHAL_GROUP=\"$(HAL_GROUP)\"

 HAL_DBUS_CPPFLAGS =       -DDBUS_API_SUBJECT_TO_CHANGE -
DDBUS_SYSTEMD_DIR=\"/etc/dbus-1/system.d\" \
+                     -I$(ROOT)/usr/include/dbus-1.0 -I$(ROOT)/usr/lib/dbus-
1.0/include   \
                      -I/usr/include/dbus-1.0 -I/usr/lib/dbus-1.0/include

-HAL_GLIB_CPPFLAGS =       -I/usr/include/glib-2.0 -I/usr/lib/glib-2.0/include
+HAL_GLIB_CPPFLAGS =       -I$(ROOT)/usr/include/glib-2.0 -
I$(ROOT)/usr/lib/glib-2.0/include \
+                     -I/usr/include/glib-2.0 -I/usr/lib/glib-2.0/include

 HAL_GETTEXT_PACKAGE =     $(TEXT_DOMAIN)
```

14.1.52 /usr/src/cmd/cmd-inet/usr.sbin/ipsecutils/ikeadm.c

- `getopt()` returns an integer. On some platforms char defaults to unsigned which means tests against -1 will fail (gcc will also flag a warning)

```
--- a/usr/src/cmd/cmd-inet/usr.sbin/ipsecutils/ikeadm.c Tue Sep 02 11:22:56
2008 -0400
+++ b/usr/src/cmd/cmd-inet/usr.sbin/ipsecutils/ikeadm.c Thu Sep 25 10:26:07
2008 -0400
@@ -3004,7 +3004,7 @@
 int
 main(int argc, char **argv)
 {
-       char    ch;
+       int     ch;
```

```
        (void) setlocale(LC_ALL, "");
 #if !defined(TEXT_DOMAIN)
```

14.1.53 /usr/src/cmd/iscsi/Makefile.iscsi

- When cross-building `libxml2` may not be installed on the base system and may be placed in `$ROOT`

```
--- a/usr/src/cmd/iscsi/Makefile.iscsi  Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/cmd/iscsi/Makefile.iscsi  Tue Sep 02 11:15:48 2008 -0400
@@ -25,7 +25,7 @@
 # ident      "%Z%%M%       %I%    %E% SMI"
 #

-CPPFLAGS += -D_FILE_OFFSET_BITS=64 -I/usr/include/libxml2
+CPPFLAGS += -D_FILE_OFFSET_BITS=64 -I$(ROOT)/usr/include/libxml2 -
I/usr/include/libxml2

 ISCSISRC     = $(SRC)/cmd/iscsi
 ISCSICOMMONDIR      = $(ISCSISRC)/common
diff -r 4f051ff1b998 usr/src/cmd/iscsi/iscsitgtd/Makefile.com
--- a/usr/src/cmd/iscsi/iscsitgtd/Makefile.com Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/cmd/iscsi/iscsitgtd/Makefile.com Tue Sep 02 11:15:48 2008 -0400
@@ -49,7 +49,7 @@
 NATIVE_CFLAGS += $(CTF_FLAGS)

 CFLAGS +=    $(CCVERBOSE)
-CPPFLAGS += -D_LARGEFILE64_SOURCE=1 -I/usr/include/libxml2
+CPPFLAGS += -D_LARGEFILE64_SOURCE=1 -I$(ROOT)/usr/include/libxml2 -
I/usr/include/libxml2
 CFLAGS64 += $(CCVERBOSE)

 GROUP = sys
```

14.1.54 /usr/src/cmd/cmd-inet/usr.sbin/kssl/kssladm/kssladm_create.c

- `getopt()` returns an integer. On some platforms char defaults to unsigned which means tests against -1 will fail (gcc will also flag a warning)

```
--- a/usr/src/cmd/cmd-inet/usr.sbin/kssl/kssladm/kssladm_create.c       Tue Sep
02 11:22:56 2008 -0400
+++ b/usr/src/cmd/cmd-inet/usr.sbin/kssl/kssladm/kssladm_create.c       Thu Sep
25 10:26:07 2008 -0400
@@ -1040,7 +1040,7 @@
        struct sockaddr_in server_addr;
        char *format = NULL;
        char *port, *addr;
-       char c;
+       int c;
        int pcnt;
        kssl_params_t *kssl_params;
        int bufsize;
```

14.1.55 /usr/src/cmd/cmd-inet/usr.sbin/kssl/kssladm/kssladm_delete.c

- `getopt()` returns an integer. On some platforms char defaults to unsigned which means tests against -1 will fail (gcc will also flag a warning)

```
--- a/usr/src/cmd/cmd-inet/usr.sbin/kssl/kssladm/kssladm_delete.c       Tue Sep
02 11:22:56 2008 -0400
+++ b/usr/src/cmd/cmd-inet/usr.sbin/kssl/kssladm/kssladm_delete.c       Thu Sep
25 10:26:07 2008 -0400
@@ -48,9 +48,8 @@
 do_delete(int argc, char *argv[])
 {
        struct sockaddr_in server_addr;
-       char c;
        char *port, *addr;
-       int pcnt;
+       int c, pcnt;

        if (argc < 3) {
                goto err;
```

14.1.56 /usr/src/cmd/cmd-inet/usr.sbin/kssl/ksslcfg/ksslcfg_create.c

- `getopt()` returns an integer. On some platforms char defaults to unsigned which means tests against -1 will fail (gcc will also flag a warning)

```
--- a/usr/src/cmd/cmd-inet/usr.sbin/kssl/ksslcfg/ksslcfg_create.c       Tue Sep
02 11:22:56 2008 -0400
+++ b/usr/src/cmd/cmd-inet/usr.sbin/kssl/ksslcfg/ksslcfg_create.c       Thu Sep
25 10:26:07 2008 -0400
@@ -452,7 +452,7 @@
 int
 do_create(int argc, char *argv[])
 {
-       char c;
+       int c;
        char *buf, *ptr, *instance_name;
        char *inaddr_any_name = NULL;
        int i, status, len, pcnt;
```

14.1.57 /usr/src/cmd/cmd-inet/usr.sbin/kssl/ksslcfg/ksslcfg_delete.c

- `getopt()` returns an integer. On some platforms char defaults to unsigned which means tests against -1 will fail (gcc will also flag a warning)

```
--- a/usr/src/cmd/cmd-inet/usr.sbin/kssl/ksslcfg/ksslcfg_delete.c       Tue Sep
02 11:22:56 2008 -0400
+++ b/usr/src/cmd/cmd-inet/usr.sbin/kssl/ksslcfg/ksslcfg_delete.c       Thu Sep
25 10:26:07 2008 -0400
@@ -231,7 +231,7 @@
 int
 do_delete(int argc, char *argv[])
 {
-       char c;
+       int c;
        int status, len, pcnt;
```

```
        char address_port[MAX_ADRPORT_LEN + 1];
        char *instance_name;
```

14.1.58 /usr/src/cmd/cmd-inet/usr.sbin/ping/ping.c

- `getopt()` returns an integer. On some platforms char defaults to unsigned which means tests against -1 will fail (gcc will also flag a warning)

```
--- a/usr/src/cmd/cmd-inet/usr.sbin/ping/ping.c Tue Sep 02 11:22:56 2008 -0400
+++ b/usr/src/cmd/cmd-inet/usr.sbin/ping/ping.c Thu Sep 25 10:26:07 2008 -0400
@@ -1198,7 +1198,7 @@
        }

        /* get the local sock info */
-       if (getsockname(tmp_fd, sock, &sock_len) < 0) {
+       if (getsockname(tmp_fd, sock, (Psocklen_t) &sock_len) < 0) {
                Fprintf(stderr, "%s: getsockname: %s\n", progname,
                    strerror(errno));
                exit(EXIT_FAILURE);
@@ -1365,7 +1365,7 @@
                }

                /* .... and see what port kernel picked for us */
-               if (getsockname(send_sock, sp, &slen) < 0) {
+               if (getsockname(send_sock, sp, (Psocklen_t) &slen) < 0) {
                        Fprintf(stderr, "%s: getsockname %s\n", progname,
                            strerror(errno));
                        exit(EXIT_FAILURE);
```

14.1.59 /usr/src/cmd/cmd-inet/usr.sbin/snoop/snoop_capture.c

- Cast operand to avoid gcc warning.

```
--- a/usr/src/cmd/cmd-inet/usr.sbin/snoop/snoop_capture.c       Tue Sep 02
11:22:56 2008 -0400
+++ b/usr/src/cmd/cmd-inet/usr.sbin/snoop/snoop_capture.c       Thu Sep 25
10:26:07 2008 -0400
@@ -764,7 +764,7 @@
        nhdr.sbh_msglen  = ohdrp->o_msglen;
        nhdr.sbh_totlen  = ohdrp->o_totlen;
        nhdr.sbh_drops   = ohdrp->o_drops;
-       nhdr.sbh_timestamp = ohdrp->o_time;
+       nhdr.sbh_timestamp = (struct timeval) ohdrp->o_time;

        *(struct sb_hdr *)ohdrp = nhdr;
 }
```

14.1.60 /usr/src/cmd/deroff/deroff.c

- `getopt()` returns an integer. On some platforms char defaults to unsigned which means tests against -1 will fail (gcc will also flag a warning)

```
--- a/usr/src/cmd/deroff/deroff.c       Tue Sep 02 11:22:56 2008 -0400
+++ b/usr/src/cmd/deroff/deroff.c       Thu Sep 25 10:26:07 2008 -0400
@@ -92,7 +92,7 @@
```

```
 static char *line = NULL;

-static char c;
+static int c;
 static int pc;
 static int ldelim      = NOCHAR;
 static int rdelim      = NOCHAR;
```

14.1.61 /usr/src/cmd/dfs.cmds/general/general.c

- Move static function prototypes from within routine to global area: gcc doesn't cope with prototypes declared this way.

```
--- a/usr/src/cmd/dfs.cmds/general/general.c    Tue Sep 02 11:22:56 2008 -0400
+++ b/usr/src/cmd/dfs.cmds/general/general.c    Thu Sep 25 10:26:07 2008 -0400
@@ -58,13 +58,13 @@

 static char *getfs();
 void perror();
+static int invalid();

 int
 main(argc, argv)
 int argc;
 char **argv;
 {
-       static int invalid();
        extern char *optarg;
        extern int optind;
        FILE *dfp;              /* fp for dfs list */
```

14.1.62 /usr/src/cmd/ed/Makefile

- GNU linker does not understand mapfiles used by the Solaris linker so we make them disappear.

```
--- a/usr/src/cmd/ed/Makefile   Tue Sep 02 11:22:56 2008 -0400
+++ b/usr/src/cmd/ed/Makefile   Thu Sep 25 10:26:07 2008 -0400
@@ -46,7 +46,10 @@
 LDLIBS += -lmapmalloc -lgen -lcrypt_i
 XGETFLAGS += -a -x ed.xcl

-MAPFILE.INT =  ../expr/mapfile-intf
+sparc_MAPFILEINT = ../expr/mapfile-intf
+i386_MAPFILEINT = ../expr/mapfile-intf
+s390_MAPFILEINT =
+MAPFILE.INT =  $($(MACH)_MAPFILEINT)
 LDFLAGS +=     $(MAPFILE.INT:%=-M%)

 POFILE= ed_cmd.po
```

14.1.63 /usr/src/cmd/fdisk/fdisk.c

- Not all platforms default to `signed char` when `char` is declared. Ensure comparison with values less than 0 will work and not automatically evaluated as false.

```
--- a/usr/src/cmd/fdisk/fdisk.c Tue Sep 02 11:22:56 2008 -0400
+++ b/usr/src/cmd/fdisk/fdisk.c Thu Sep 25 10:26:07 2008 -0400
@@ -1860,7 +1860,7 @@
                                        return;
                                break;
                        case '3':
-                               if (pdelete() == -1)
+                               if ((signed char) pdelete() == -1)
                                        return;
                                break;
                        case '4':
```

14.1.64 usr/src/cmd/fmthard/fmthard.c

- Add support for System z.

```
--- a/usr/src/cmd/fmthard/fmthard.c     Tue Sep 02 11:22:56 2008 -0400
+++ b/usr/src/cmd/fmthard/fmthard.c     Thu Sep 25 10:26:07 2008 -0400
@@ -128,6 +128,9 @@
 #elif defined(i386)
 /* use installgrub(1M) to install boot blocks */
 static char *uboot = "";
+#elif defined(__s390__)
+/* use installgrub(1M) to install boot blocks */
+static char *uboot = "";
 #else
 #error No platform defined.
 #endif /* various platform-specific definitions */
@@ -163,6 +166,9 @@
 #elif defined(i386)
        while ((c = getopt(argc, argv, "ed:u:in:qb:p:s:")) != EOF)

+#elif defined(__s390__)
+       fprintf(stderr,"fmthard not required for System z\n");
+       return 0;
 #else
 #error No platform defined.
 #endif
@@ -685,6 +691,9 @@
 "Usage:        fmthard [ -i ] [ -S ] [-I geom_file]  \
 -n volumename | -s datafile  [ -d arguments] raw-device\n");

+#elif defined(__s390__)
+"fmthard is not required under System z and does nothing\n");
+
 #else
 #error No platform defined.
 #endif
```

14.1.65 /usr/src/cmd/format/ctlr_scsi.c

- Add System z support.

```
--- a/usr/src/cmd/format/ctlr_scsi.c    Tue Sep 02 11:22:56 2008 -0400
+++ b/usr/src/cmd/format/ctlr_scsi.c    Thu Sep 25 10:26:07 2008 -0400
@@ -100,7 +100,7 @@
 static int     scsi_read_defect_data(struct defect_list *, int);
 static int     scsi_ck_format(void);

-#ifdef i386
```

```
+#if defined(i386) || defined(__s390__)
 static int     scsi_rdwr(int, int, diskaddr_t, int, caddr_t, int, int *);
 #endif /* i386 */

@@ -131,7 +131,7 @@
 static int     scsi_read_defect_data();
 static int     scsi_ck_format();

-#ifdef i386
+#if defined(i386) || defiined(__s390__)
 static int     scsi_rdwr(int, int, diskaddr_t, int, caddr_t, int, int *);
 #endif /* i386 */
```

14.1.66 /usr/src/cmd/format/menu_fdisk.c

- Add System z support.

```
--- a/usr/src/cmd/format/menu_fdisk.c   Tue Sep 02 11:22:56 2008 -0400
+++ b/usr/src/cmd/format/menu_fdisk.c   Thu Sep 25 10:26:07 2008 -0400
@@ -85,7 +85,7 @@
 /* Function prototypes */
 #ifdef __STDC__

-#if    defined(sparc)
+#if    defined(sparc) || defined(__s390__)

 static int getbyte(uchar_t **);
 static int getlong(uchar_t **);
@@ -96,7 +96,7 @@

 #else  /* __STDC__ */

-#if    defined(sparc)
+#if    defined(sparc) || defined(__s390__)

 static int getbyte();
 static int getlong();
@@ -140,15 +140,33 @@
          * to another data structure to avoid an alignment exception.
          */
         (void) bcopy(bootptr, partp, sizeof (struct ipart));
+#elif defined(__s390__)
+       /*
+        * System z platform:
+        *
+        * Packing short/word for struct ipart to resolve
+        *      little endian on System z since it is not
+        *      properly aligned on System z.
+        */
+       partp->bootid = getbyte((uchar_t **)&bootptr);
+       partp->beghead = getbyte((uchar_t **)&bootptr);
+       partp->begsect = getbyte((uchar_t **)&bootptr);
+       partp->begcyl = getbyte((uchar_t **)&bootptr);
+       partp->systid = getbyte((uchar_t **)&bootptr);
+       partp->endhead = getbyte((uchar_t **)&bootptr);
+       partp->endsect = getbyte((uchar_t **)&bootptr);
+       partp->endcyl = getbyte((uchar_t **)&bootptr);
+       partp->relsect = getlong((uchar_t **)&bootptr);
+       partp->numsect = getlong((uchar_t **)&bootptr);
 #else
 #error  No Platform defined
```

```
 #endif /* defined(sparc) */
 }

 /*
- * Get a correct byte/short/word routines for Sparc platform.
+ * Get a correct byte/short/word routines for Sparc/System z platform.
  */
-#if defined(sparc)
+#if defined(sparc) || defined(__s390__)
 static int
 getbyte(uchar_t **bp)
 {
```

14.1.67 /usr/src/cmd/halt/halt.c

- Make variables be of the same type used as parameters in the functions in which they are used.

```
--- a/usr/src/cmd/halt/halt.c   Tue Sep 02 11:22:56 2008 -0400
+++ b/usr/src/cmd/halt/halt.c   Thu Sep 25 10:26:07 2008 -0400
@@ -375,7 +375,7 @@
 {
        pid_t pid;
        zoneid_t *zones;
-       size_t nz = 0, old_nz;
+       uint_t nz = 0, old_nz;
        int i;
        char zname[ZONENAME_MAX];

@@ -461,7 +461,7 @@
 check_zones_haltedness()
 {
        int t = 0, t_prog = 0;
-       size_t nz = 0, last_nz;
+       uint_t nz = 0, last_nz;

        do {
                last_nz = nz;
```

14.1.68 /usr/src/cmd/init/init.c

- Not all platforms default to `signed char` when `char` is declared. Ensure comparison with values less than 0 will work and not automatically evaluated as false.

```
--- a/usr/src/cmd/init/init.c   Tue Sep 02 11:22:56 2008 -0400
+++ b/usr/src/cmd/init/init.c   Thu Sep 25 10:26:08 2008 -0400
@@ -778,7 +778,7 @@
                        chg_lvl_flag = FALSE;

                        if (state_to_flags(cur_state) & LSEL_RUNLEVEL) {
-                               char rl = state_to_name(cur_state);
+                               signed char rl = state_to_name(cur_state);

                                if (rl != -1)
                                        lscf_set_runlevel(rl);
```

14.1.69 /usr/src/cmd/link/Makefile

- Ensure System z searches proto area for its libraries.

```
--- a/usr/src/cmd/link/Makefile Tue Sep 02 11:22:56 2008 -0400
+++ b/usr/src/cmd/link/Makefile Thu Sep 25 10:26:08 2008 -0400
@@ -33,6 +33,10 @@
 include ../Makefile.cmd

 clean $(XPG4) := OBJS += values-xpg4.o
+
+s390_XLIBS =   -L$(ROOT)/usr/lib -ldl
+$(PROG)        :=      LDFLAGS += $($(MACH)_XLIBS)
+$(XPG4)        :=      LDFLAGS += $($(MACH)_XLIBS)

 FILEMODE= 0555
```

14.1.70 /usr/srcm/cmd/lp/Makefile.lp

- Remove flags not supported by gcc-4 by specifying platform specific flags and <*mach*>_CFLAGS[64] and then using $($(MACH)_CFLAGS[64]) to retrieve them

```
--- a/usr/src/cmd/lp/Makefile.lp  Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/cmd/lp/Makefile.lp  Tue Sep 02 11:15:48 2008 -0400
@@ -93,8 +93,14 @@
 LINTSEC     =      $(LPLIB)/secure/llib-llpsec.ln
 LINTUSR     =      $(LPLIB)/users/llib-llpusr.ln

-CFLAGS += -_gcc=-fwritable-strings -_gcc=-Wno-sequence-points
-CFLAGS64 += -_gcc=-fwritable-strings -_gcc=-Wno-sequence-points
+sparc_CFLAGS += -_gcc=-fwritable-strings -_gcc=-Wno-sequence-points
+i386_CFLAGS +=     -_gcc=-fwritable-strings -_gcc=-Wno-sequence-points
+s390_CFLAGS +=
+CFLAGS =    $($(MACH)_CFLAGS)
+sparc_CFLAGS64 +=  -_gcc=-fwritable-strings -_gcc=-Wno-sequence-points
+i386_CFLAGS64 +=   -_gcc=-fwritable-strings -_gcc=-Wno-sequence-points
+s390_CFLAGS64 +=
+CFLAGS64 =  $($(MACH)_CFLAGS64)

 all:=           TARGET= all
 install:=       TARGET= install
```

14.1.71 /usr/srcmd/cmd/lp/filter/postscript/font/Makefile

- Use cross-build compiler to build the native program (changed in later revision to use XBUILDCC)

```
--- a/usr/src/cmd/lp/filter/postscript/font/Makefile  Tue Aug 19 11:19:58 2008 -
0400
+++ b/usr/src/cmd/lp/filter/postscript/font/Makefile  Tue Sep 02 11:15:48 2008 -
0400
@@ -46,9 +46,12 @@

 POFILE =    lp_filter_postscript_font.po
```

```
+sparc_NATCC =      $(NATIVECC)
+i386_NATCC =$(NATIVECC)
+s390_NATCC =$(sparc_CC)
 NATIVEDIR =        native
 NATIVEPROG =       $(NATIVEDIR)/$(PROG)
-$(NATIVEPROG) :=   CC=$(NATIVECC)
+$(NATIVEPROG) :=   CC=$($(MACH)_NATCC)
 $(NATIVEPROG) :=   LDLIBS=
 $(NATIVEPROG) :=   CPPFLAGS.master=

--- a/usr/src/cmd/lp/filter/postscript/font/Makefile  Tue Sep 02 11:22:56 2008
-0400
+++ b/usr/src/cmd/lp/filter/postscript/font/Makefile  Thu Sep 25 10:26:08 2008
-0400
@@ -46,12 +46,9 @@

 POFILE =       lp_filter_postscript_font.po

-sparc_NATCC =  $(NATIVECC)
-i386_NATCC =   $(NATIVECC)
-s390_NATCC =   $(sparc_CC)
 NATIVEDIR =            native
 NATIVEPROG =           $(NATIVEDIR)/$(PROG)
-$(NATIVEPROG) :=       CC=$($(MACH)_NATCC)
+$(NATIVEPROG) :=       CC=$(XBUILDCC)
 $(NATIVEPROG) :=       LDLIBS=
 $(NATIVEPROG) :=       CPPFLAGS.master=
```

14.1.72 /usr/srcm/cmd/lvm/metassist/controller/Makefile

- When cross-building `libxml2` may not be installed on the base system and may be placed in `$ROOT`

```
--- a/usr/src/cmd/lvm/metassist/controller/Makefile  Tue Aug 19 11:19:58 2008 -
0400
+++ b/usr/src/cmd/lvm/metassist/controller/Makefile  Tue Sep 02 11:15:48 2008 -
0400
@@ -36,6 +36,7 @@
 include $(METASSIST_TOPLEVEL)/Makefile.env

 INCLUDES += -I.. -I../common -I../xml -I../layout \
+            -I$(ROOT)/usr/include/libxml2     \
             -I/usr/include/libxml2
 CFLAGS   += $(INCLUDES)
```

14.1.73 /usr/src/cmd/lvm/metassist/xml/Makefile

- When cross-building `libxml2` may not be installed on the base system and may be placed in `$ROOT`

```
--- a/usr/src/cmd/lvm/metassist/xml/Makefile   Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/cmd/lvm/metassist/xml/Makefile   Tue Sep 02 11:15:48 2008 -0400
@@ -35,7 +35,7 @@
 include $(METASSIST_TOPLEVEL)/../../Makefile.cmd
 include $(METASSIST_TOPLEVEL)/Makefile.env

-INCLUDES += -I /usr/include/libxml2 -I../common
+INCLUDES += -I$(ROOT)/usr/include/libxml2 -I/usr/include/libxml2 -I../common
```

```
 CFLAGS   += $(INCLUDES)

 POFILE = xmlp.po
```

14.1.74 /usr/src/cmd/man/src/util/instant.src/Makefile

- Ensure System z searches proto area for its libraries.

```
--- a/usr/src/cmd/man/src/util/instant.src/Makefile  Tue Sep 02 11:22:56 2008 -
0400
+++ b/usr/src/cmd/man/src/util/instant.src/Makefile  Thu Sep 25 10:26:08 2008 -
0400
@@ -83,6 +83,8 @@
 #OPT   = -O
 CFLAGS += $(OPT) $(REGEX_INC) $(DEFINES)
 LDFLAGS        += $(OPT)
+s390_LDLIBS = -L$(ROOT)/usr/lib -ldl
+LDLIBS      = $($(MACH)_LDLIBS)
 REGEX  = -L$(REGEX_LIB) -ltptregexp

 CFILES = main.c util.c info.c translate.c traninit.c tranvar.c tables.c \
@@ -98,7 +100,7 @@
        cd tptregexp; $(MAKE) all

 instant: $(OBJ)
-       $(CC) $(CFLAGS) $(CPPFLAGS) $(LDFLAGS) -o $@ $(OBJ) $(REGEX)
+       $(CC) $(CFLAGS) $(CPPFLAGS) $(LDFLAGS) -o $@ $(OBJ) $(REGEX) $(LDLIBS)
        $(POST_PROCESS)
        cp $@ ..
```

14.1.75 /usr/src/cmd/man/src/util/instant.src/tables.c

- Not all platforms default to `signed char` when `char` is declared. Ensure comparison with values less than 0 will work and not automatically evaluated as false.

```
--- a/usr/src/cmd/man/src/util/instant.src/tables.c  Tue Sep 02 11:22:56 2008 -
0400
+++ b/usr/src/cmd/man/src/util/instant.src/tables.c  Thu Sep 25 10:26:08 2008 -
0400
@@ -1838,7 +1838,7 @@
        for (i = 0;  i < ep->ncont;  i++)       {
                if (ep->cont[i].type == '-')    {
                        for (cp = ep->cont[i].ch.data; *cp; cp++, count++)
-                               if (*cp == -1)
+                               if ((signed char) *cp == -1)
                                        return (BOFTTextThresh + 1);
                } else
                if (ep->cont[i].type == '(')    {
```

14.1.76 /usr/src/cmd/man/src/util/instant.src/translate.c

- Not all platforms default to `signed char` when `char` is declared. Ensure comparison with values less than 0 will work and not automatically evaluated as false.

```
--- a/usr/src/cmd/man/src/util/instant.src/translate.c  Tue Sep 02 11:22:56
2008 -0400
+++ b/usr/src/cmd/man/src/util/instant.src/translate.c  Thu Sep 25 10:26:08
2008 -0400
@@ -690,7 +690,7 @@
                         break;
                 }
             }
-            if ( *cp == -1 )    *cp = '^Q';
+            if ( (signed char) *cp == -1 )      *cp = '^Q';
             if (!mapped && t->trim && (strchr(t->trim, *cp) != NULL))   {
                 continue;
             }
```

14.1.77 /usr/src/cmd/man/src/util/nsgmls.src/Makefile

```
--- a/usr/src/cmd/man/src/util/nsgmls.src/Makefile   Tue Sep 02 11:22:56 2008 -
0400
+++ b/usr/src/cmd/man/src/util/nsgmls.src/Makefile   Thu Sep 25 10:26:08 2008 -
0400
@@ -56,7 +56,8 @@
 # If you defined SP_HAVE_SOCKET, add any libraries that are needed for sockets
 #-lsocket -lnsl needed on Solaris 2.x
 # -lnsl on SunOS 4.1.3
-XLIBS=-lsocket -lnsl
+s390_XLIBS=-ldl -lmd -lmp -lm -lscf -luutil -lgen
+XLIBS=-lsocket -lnsl $($(MACH)_XLIBS)
 # -L/usr/local/lib may be needed on the RS/6000
 LIBS += $(ENVLDLIBS1) $(ENVLDLIBS2) $(XLIBS) $(CCNEEDED) -lc
 # If you're building in another directory, copy or link this Makefile
```

14.1.78 /usr/src/cmd/man/src/util/nsgmls.src/include/config.h

- Support gcc v4

```
--- a/usr/src/cmd/man/src/util/nsgmls.src/include/config.h      Tue Sep 02
11:22:56 2008 -0400
+++ b/usr/src/cmd/man/src/util/nsgmls.src/include/config.h      Thu Sep 25
10:26:08 2008 -0400
@@ -33,6 +33,9 @@
 #endif
 #if __GNUC__ > 2 || (__GNUC__ == 2 && __GNUC_MINOR__ >= 7)
 #define        SP_ANSI_FOR_SCOPE
+#endif
+#if __GNUC__ >= 3
+#define        SP_DEFINE_TEMPLATES
 #endif

 #endif /* __GNUG__ */
```

14.1.79 /usr/src/cmd/man/src/util/nsgmls.src/lib/events.h

- gcc v4.1 and above does not allow `#pragma` statements within a structure.

```
--- a/usr/src/cmd/man/src/util/nsgmls.src/lib/events.h  Tue Sep 02 11:22:56
2008 -0400
+++ b/usr/src/cmd/man/src/util/nsgmls.src/lib/events.h  Thu Sep 25 10:26:08
2008 -0400
@@ -1,6 +1,5 @@
 // Copyright (c) 1995 James Clark
```

```
 // See the file COPYING for copying permission.
-#pragma ident  "%Z%%M% %I%     %E% SMI"

 EVENT(MessageEvent, message)
 EVENT(DataEvent, data)
```

14.1.80 /usr/src/cmd/ptools/pargs/pargs.c

```
--- a/usr/src/cmd/ptools/pargs/pargs.c  Tue Sep 02 11:22:56 2008 -0400
+++ b/usr/src/cmd/ptools/pargs/pargs.c  Thu Sep 25 10:26:08 2008 -0400
@@ -701,7 +701,8 @@
         (void) elfcap_hw1_to_str(ELFCAP_STYLE_UC, val, str, n,
             ELFCAP_FMT_PIPSPACE, EM_386);
 #elif defined(__s390)
-       (void) hwcap_1_val2str(val, str, n, CAP_FMT_PIPSPACE, EM_S390);
+       (void) elfcap_hw1_to_str(ELFCAP_STYLE_UC, val, str, n,
+           ELFCAP_FMT_PIPSPACE, EM_S390);
 #else
 #error "port me"
 #endif
```

14.1.81 /usr/src/cmd/sgs/elfdump/Makefile.com

- Add support for System z.

```
--- a/usr/src/cmd/sgs/elfdump/Makefile.com      Tue Sep 02 11:22:56 2008 -0400
+++ b/usr/src/cmd/sgs/elfdump/Makefile.com      Thu Sep 25 10:26:08 2008 -0400
@@ -33,7 +33,8 @@

 COMOBJ =       main.o  corenote.o      struct_layout.o \
                struct_layout_i386.o    struct_layout_amd64.o \
-               struct_layout_sparc.o   struct_layout_sparcv9.o
+               struct_layout_sparc.o   struct_layout_sparcv9.o \
+               struct_layout_s390.o    struct_layout_s390x.o

 COMOBJ32 =     elfdump32.o fake_shdr32.o
```

14.1.82 /usr/src/cmd/sgs/elfdump/Makefile.targ

```
--- a/usr/src/cmd/sgs/elfdump/Makefile.targ     Tue Sep 02 11:22:56 2008 -0400
+++ b/usr/src/cmd/sgs/elfdump/Makefile.targ     Thu Sep 25 10:26:08 2008 -0400
@@ -44,6 +44,7 @@

 gen_struct_layout:     ../common/gen_struct_layout.c
                $(NATIVECC) $(NATIVECC_CFLAGS) $(CFLAGS) -o $@ \
+                       $(STRUCT_CFLAGS)        \
                        ../common/gen_struct_layout.c

 $(PROG):       $(OBJS) $(MAPFILE) gen_struct_layout
```

14.1.83 /usr/src/cmd/sgs/elfdump/common/elfdump.c

- In the System z GOT there are a couple of entries at the top that don't relate to relocation types. In the first pass through the table we've ignored them so ignore them now.

```
--- a/usr/src/cmd/sgs/elfdump/common/elfdump.c  Tue Sep 02 11:22:56 2008 -0400
```

```
+++ b/usr/src/cmd/sgs/elfdump/common/elfdump.c  Thu Sep 25 10:26:08 2008 -0400
@@ -3462,6 +3462,9 @@

                gip = &gottable[gotndx];

+               if (gip->g_reltype == 0)
+                       continue;
+
                gaddr = gotbgn + (gotndx * gentsize);
                gindex = (Sword)(gaddr - gotsymaddr) / (Sword)gentsize;
```

14.1.84 /usr/src/cmd/sgs/elfdump/common/gen_struct_layout.c

- Add System z support.

```
--- a/usr/src/cmd/sgs/elfdump/common/gen_struct_layout.c        Tue Sep 02
11:22:56 2008 -0400
+++ b/usr/src/cmd/sgs/elfdump/common/gen_struct_layout.c        Thu Sep 25
10:26:08 2008 -0400
@@ -86,6 +86,14 @@
 #elif defined(__sparc)

 #define        MACH    "sparc"
+
+#elif defined(__s390x)
+
+#define        MACH    "s390x"
+
+#elif defined(__s390)
+
+#define        MACH    "s390"

 #else
```

14.1.85 /usr/src/cmd/sgs/include/machdep.h

- Add support for System z

```
--- a/usr/src/cmd/sgs/include/machdep.h Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/cmd/sgs/include/machdep.h Tue Sep 02 11:15:49 2008 -0400
@@ -44,6 +44,10 @@

 #include <i386/machdep_x86.h>

+#elif defined(__s390)
+
+#include <s390/machdep_s390.h>
+
 #else

 #error "machdep.h does not understand current machine"
```

14.1.86 /usr/src/cmd/sgs/ld/Makefile.com

- GNU linker does not understand mapfiles used by the Solaris linker so we make them disappear.

```
--- a/usr/src/cmd/sgs/ld/Makefile.com   Tue Sep 02 11:22:56 2008 -0400
+++ b/usr/src/cmd/sgs/ld/Makefile.com   Thu Sep 25 10:26:08 2008 -0400
@@ -38,7 +38,11 @@
 .PARALLEL:     $(OBJS)

 MAPFILES =     ../common/mapfile-intf $(MAPFILE.NGB)
-MAPOPTS =      $(MAPFILES:%=-M%)
+sparc_MAPOPTS =        $(MAPFILES:%=-M%)
+i386_MAPOPTS = $(MAPFILES:%=-M%)
+s390_MAPOPTS =
+MAPOPTS =      $($(MACH)_MAPOPTS)
+

 LDFLAGS +=     $(VERSREF) $(USE_PROTO) $(MAPOPTS) $(VAR_LD_LLDFLAGS)
 LDLIBS +=      $(LDLIBDIR) $(LD_LIB) $(ELFLIBDIR) -lelf \
```

14.1.87 usr/src/cmd/sgs/libconv/common/arch.c

- Add System z support.

```
--- a/usr/src/cmd/sgs/libconv/common/arch.c     Tue Sep 02 11:22:56 2008 -0400
+++ b/usr/src/cmd/sgs/libconv/common/arch.c     Thu Sep 25 10:26:08 2008 -0400
@@ -49,7 +49,8 @@
                return (ELFCLASS32);

        if ((strstr(buf, MSG_ORIG(MSG_ARCH_SPARCV9)) != NULL) ||
-           (strstr(buf, MSG_ORIG(MSG_ARCH_AMD64)) != NULL))
+           (strstr(buf, MSG_ORIG(MSG_ARCH_AMD64)) != NULL) ||
+           (strstr(buf, MSG_ORIG(MSG_ARCH_S390X)) != NULL))
                return (ELFCLASS64);

        return (ELFCLASS32);
```

14.1.88 /usr/src/cmd/sgs/rtld/Makefile.com

- Add System z support.
- To create partially linked objects with program headers with gnu ld we cannot use the `-i` flag.

```
--- a/usr/src/cmd/sgs/rtld/Makefile.com Tue Sep 02 11:22:56 2008 -0400
+++ b/usr/src/cmd/sgs/rtld/Makefile.com Thu Sep 25 11:44:45 2008 -0400
@@ -98,7 +98,10 @@
                $(RTLDLIB) -lrtld \
                $(LDLIB) $(LD_LIB)

-DYNFLAGS +=    -i -e _rt_boot $(VERSREF) $(ZNODLOPEN) \
+sparc_DYNFLAGS = -i
+i386_DYNFLAGS = -i
+s390_DYNFLAGS =
+DYNFLAGS +=    $($(MACH)_DYNFLAGS) -e _rt_boot $(VERSREF) $(ZNODLOPEN) \
                $(ZINTERPOSE) -zdtrace=dtrace_data '-R$$ORIGIN'

 BUILD.s=       $(AS) $(ASFLAGS) $< -o $@
@@ -118,11 +121,15 @@
 SGSMSGINTEL=   ../common/rtld.intel.msg
 SGSMSGINTEL32= ../common/rtld.intel32.msg
 SGSMSGINTEL64= ../common/rtld.intel64.msg
+SGSMSGZSERIES=  ../common/rtld.zSeries.msg
+SGSMSGZSERIES32=../common/rtld.zSeries32.msg
+SGSMSGZSERIES64=../common/rtld.zSeries64.msg
```

```
 SGSMSGCHK=      ../common/rtld.chk.msg
 SGSMSGTARG=     $(SGSMSGCOM)
 SGSMSGALL=      $(SGSMSGCOM) $(SGSMSG32) $(SGSMSG64) \
                 $(SGSMSGSPARC) $(SGSMSGSPARC32) $(SGSMSGSPARC64) \
-                $(SGSMSGINTEL) $(SGSMSGINTEL32) $(SGSMSGINTEL64)
+                $(SGSMSGINTEL) $(SGSMSGINTEL32) $(SGSMSGINTEL64) \
+                $(SGSMSGZSERIES) $(SGSMSGS39032) $(SGSMSGS39064)

 SGSMSGFLAGS1=   $(SGSMSGFLAGS) -m $(BLTMESG)
 SGSMSGFLAGS2=   $(SGSMSGFLAGS) -h $(BLTDEFS) -d $(BLTDATA) -n rtld_msg
```

14.1.89 /usr/src/cmd/sgs/rtld/Makefile.targ

```
--- a/usr/src/cmd/sgs/rtld/Makefile.targ        Tue Sep 02 11:22:56 2008 -0400
+++ b/usr/src/cmd/sgs/rtld/Makefile.targ        Thu Sep 25 10:26:08 2008 -0400
@@ -71,6 +71,8 @@
                     $(LDLIBS) $(CRTN)
                 $(POST_PROCESS_SO)

+#              $(SGSPROTO)/ld -o $@ -dy -G $(DYNFLAGS) $(CRTI) $(PICS) \
+
 delete:
                 -$(RM) $(RTLD)
```

14.1.90 /usr/src/cmd/svc/configd/configd.h

- Correct parameter to match function declaration.

```
--- a/usr/src/cmd/svc/configd/configd.h Tue Sep 02 11:22:56 2008 -0400
+++ b/usr/src/cmd/svc/configd/configd.h Thu Sep 25 10:26:09 2008 -0400
@@ -705,7 +705,7 @@
 int rc_node_next_snaplevel(rc_node_ptr_t *, rc_node_ptr_t *);

 int rc_node_setup_iter(rc_node_ptr_t *, rc_node_iter_t **, uint32_t,
-    size_t, const char *);
+    uint32_t, const char *);

 int rc_iter_next(rc_node_iter_t *, rc_node_ptr_t *, uint32_t);
 int rc_iter_next_value(rc_node_iter_t *, struct rep_protocol_value_response *,
```

14.1.91 /usr/src/cmd/svc/svccfg/Makefile

- When cross-building `libxml2` may not be installed on the base system and may be placed in `$ROOT`
- Gnu ld doesn't know how to handle mapfiles used by Solaris linker so get rid of them.

```
--- a/usr/src/cmd/svc/svccfg/Makefile   Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/cmd/svc/svccfg/Makefile   Tue Sep 02 11:15:49 2008 -0400
@@ -66,7 +66,7 @@
 MAPFILES += $(MAPFILE.LEX) $(MAPFILE.NGB)
 MAPOPTS =   $(MAPFILES:%=-M%)

-MYCPPFLAGS =-I ../common -I/usr/include/libxml2
+MYCPPFLAGS =-I ../common -I$(ROOT)/usr/include/libxml2 -I/usr/include/libxml2
 CPPFLAGS += $(MYCPPFLAGS)
 LDFLAGS +=  $(MAPOPTS)
```

```
--- a/usr/src/cmd/svc/svccfg/Makefile   Tue Sep 02 11:22:56 2008 -0400
+++ b/usr/src/cmd/svc/svccfg/Makefile   Thu Sep 25 10:26:09 2008 -0400
@@ -64,7 +64,10 @@
 # still export a number of "yy*" (libl) interfaces.  Reduce all other symbols
 # to local scope.
 MAPFILES +=    $(MAPFILE.LEX) $(MAPFILE.NGB)
-MAPOPTS =      $(MAPFILES:%=-M%)
+sparc_MAPOPTS =        $(MAPFILES:%=-M%)
+i386_MAPOPTS = $(MAPFILES:%=-M%)
+s390_MAPOPTS =
+MAPOPTS        =       $($(MACH)_MAPOPTS)

 MYCPPFLAGS =   -I ../common -I$(ROOT)/usr/include/libxml2 -
I/usr/include/libxml2
 CPPFLAGS +=    $(MYCPPFLAGS)
```

14.1.92 /usr/src/cmd/svc/svcs/svcs.c

- `getopt()` returns an integer. On some platforms char defaults to unsigned which means tests against -1 will fail (gcc will also flag a warning)

```
--- a/usr/src/cmd/svc/svcs/svcs.c       Tue Sep 02 11:22:56 2008 -0400
+++ b/usr/src/cmd/svc/svcs/svcs.c       Thu Sep 25 10:26:09 2008 -0400
@@ -2956,8 +2956,8 @@
 int
 main(int argc, char **argv)
 {
-       char opt, opt_mode;
-       int i, n;
+       char opt_mode;
+       int opt, i, n;
        char *columns_str = NULL;
        char *cp;
        const char *progname;
```

14.1.93 /usr/src/cmd/tar/tar.c

- Move static function prototypes from within routine to global area: gcc doesn't cope with prototypes declared this way.

```
--- a/usr/src/cmd/tar/tar.c     Tue Sep 02 11:22:56 2008 -0400
+++ b/usr/src/cmd/tar/tar.c     Thu Sep 25 10:26:09 2008 -0400
@@ -550,6 +550,7 @@
     char *name, int oflag, mode_t mode);
 static char *skipslashes(char *string, char *start);
 static void chop_endslashes(char *path);
+static off_t   lookup(char *);

 static struct stat stbuf;

@@ -4718,7 +4719,6 @@
        time_t  mtime;
        long nsecs;
        off_t seekp;
-       static off_t    lookup(char *);

        rewind(tfile);
        if ((seekp = lookup(arg)) < 0)
```

14.1.94 /usr/src/cmd/th_tools/th_define.c

- `getopt()` returns an integer. On some platforms char defaults to unsigned which means tests against -1 will fail (gcc will also flag a warning)

```
--- a/usr/src/cmd/th_tools/th_define.c  Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/cmd/th_tools/th_define.c  Tue Sep 02 11:15:49 2008 -0400
@@ -2021,7 +2021,7 @@
       extern char *optarg;
       extern int optind;

-      char   c;           /* for parsing getopts */
+      int    c;           /* for parsing getopts */
       int    nopts = 0;   /* for backward compatibility */
       int    err = 0;
```

14.1.95 /usr/src/cmd/ttymon/tmextern.h

- System z (or gcc actually) needs tmstruct.h included to correctly compile.

```
--- a/usr/src/cmd/ttymon/tmextern.h     Tue Sep 02 11:22:56 2008 -0400
+++ b/usr/src/cmd/ttymon/tmextern.h     Thu Sep 25 10:26:09 2008 -0400
@@ -32,6 +32,8 @@
 #define        __TMEXTERN_H

 #pragma ident  "%Z%%M% %I%     %E% SMI"
+
+#include "tmstruct.h"

 #ifdef __cplusplus
 extern "C" {
```

14.1.96 /usr/src/cmd/ttymon/tmstruct.h

- Add standard mechanism to avoid problems if an include file is included more than once.

```
--- a/usr/src/cmd/ttymon/tmstruct.h     Tue Sep 02 11:22:56 2008 -0400
+++ b/usr/src/cmd/ttymon/tmstruct.h     Thu Sep 25 10:26:09 2008 -0400
@@ -22,6 +22,8 @@
 /*     Copyright (c) 1984, 1986, 1987, 1988, 1989 AT&T */
 /*       All Rights Reserved   */

+#ifndef __TMSTRUCT_DEF
+# define __TMSTRUCT_DEF

 #ident "%Z%%M% %I%     %E% SMI"        /* SVr4.0 1.4   */

@@ -105,3 +107,5 @@
                (unsigned)sizeof(struct pmtab)))

 #define        PNULL   ((struct pmtab *)NULL)
+
+#endif
```

14.1.97 /usr/src/cmd/ttymon/ttymon.c

- Move static function prototypes from within routine to global area: gcc doesn't cope with prototypes declared this way.

```
--- a/usr/src/cmd/ttymon/ttymon.c       Tue Sep 02 11:22:56 2008 -0400
+++ b/usr/src/cmd/ttymon/ttymon.c       Thu Sep 25 10:26:09 2008 -0400
@@ -61,6 +61,7 @@
 static int     set_poll();
 static int     check_spawnlimit();
 static int     mod_ttydefs();
+static void    free_defs();

 void   open_device();
 void   set_softcar();
@@ -269,7 +270,6 @@
 {
        struct  pmtab   *tp;
        int     check_modtime;
-       static  void    free_defs();
        sigset_t cset;
        sigset_t tset;
```

14.1.98 /usr/src/cmd/vi/port/ex.h

- Correct prototype to match parameter being passed (`read`/`write` is defined in `stdio.h` and returns `ssize_t`).

```
--- a/usr/src/cmd/vi/port/ex.h  Tue Sep 02 11:22:56 2008 -0400
+++ b/usr/src/cmd/vi/port/ex.h  Thu Sep 25 10:26:09 2008 -0400
@@ -531,7 +531,7 @@
 int nqcolumn(unsigned char *, unsigned char *);
 void syserror(int);
 void cleanup(bool);
-void blkio(short, unsigned char *, int (*)());
+void blkio(short, unsigned char *, ssize_t (*)());
 void tflush(void);
 short partreg(unsigned char);
 void kshift(void);
```

14.1.99 usr/src/cmd/vi/port/ex_temp.c

- `read`/`write` is defined in `stdio.h` and returns `ssize_t`.
- Fix function declarations to match parameters being passed.

```
--- a/usr/src/cmd/vi/port/ex_temp.c     Tue Sep 02 11:22:56 2008 -0400
+++ b/usr/src/cmd/vi/port/ex_temp.c     Thu Sep 25 10:26:09 2008 -0400
@@ -208,9 +208,6 @@
        return (tl);
 }

-int    read();
-int    write();
-
 unsigned char *
 getblock(atl, iof)
```

```
        line atl;
@@ -312,7 +309,7 @@
 #endif

 void
-blkio(short b, unsigned char *buf, int (*iofcn)())
+blkio(short b, unsigned char *buf, ssize_t (*iofcn)())
 {

 #ifdef VMUNIX
@@ -494,7 +491,7 @@
 unsigned char  *rbufcp;

 void
-regio(short b, int (*iofcn)())
+regio(short b, ssize_t (*iofcn)())
 {

        if (rfile == -1) {
@@ -544,7 +541,7 @@
        return (isdigit(c) ? &strregs[('z'-'a'+1)+(c-'0')] : &strregs[c-'a']);
 }

-int    shread();
+ssize_t shread();

 void
 KILLreg(int c)
@@ -567,7 +564,7 @@
 }

 /*VARARGS*/
-int
+ssize_t
 shread(void)
 {
        struct front { short a; short b; };
```

14.1.100 /usr/src/cmd/vi/port/ex_voper.c

- Move static function prototypes from within routine to global area: gcc doesn't cope with prototypes declared this way.

```
--- a/usr/src/cmd/vi/port/ex_voper.c    Tue Sep 02 11:22:56 2008 -0400
+++ b/usr/src/cmd/vi/port/ex_voper.c    Thu Sep 25 10:26:09 2008 -0400
@@ -53,6 +53,8 @@
 #define        blank()         isspace(wcursor[0])
 #endif /* PRESUNEUC */
 #define        forbid(a)       if (a) goto errlab;
+
+static int get_addr();

 unsigned char  vscandir[2] =   { '/', 0 };

@@ -84,7 +86,6 @@
        int mouse_x;
        int mouse_y;
        int oline;
-       static int get_addr();
 /* #endif PTR_ADDRESSES */

        moveop = vmove, deleteop = (int (*)())vdelete;
```

14.1.101 /usr/src/cmd/vi/port/exrecover.c

- `read`/`write` is defined in `stdio.h` and returns `ssize_t`.
- Fix function declarations to match parameters being passed.

```
--- a/usr/src/cmd/vi/port/exrecover.c   Tue Sep 02 11:22:56 2008 -0400
+++ b/usr/src/cmd/vi/port/exrecover.c   Thu Sep 25 10:26:09 2008 -0400
@@ -48,6 +48,8 @@
 #include <errno.h>

 #define DIRSIZ MAXNAMLEN
+
+static void catch();

 short tfile = -1;      /* ditto */

@@ -774,9 +776,6 @@
                }
 }

-int    read();
-int    write();
-
 unsigned char *
 getblock(atl)
        line atl;
@@ -807,7 +806,7 @@
 }

 void
-blkio(short b, unsigned char *buf, int (*iofcn)())
+blkio(short b, unsigned char *buf, ssize_t (*iofcn)())
 {

        int rc;
@@ -842,7 +841,6 @@
        int c;
        static unsigned char pbuf[9];
        void    (*sig)();
-       static void catch();

        setbuf(stdin, (char*)NULL);
        sig = signal(SIGINT, catch);
```

14.1.102 /usr/src/cmd/zfs/zfs_main.c

- `getopt()` returns an integer. On some platforms char defaults to unsigned which means tests against -1 will fail (gcc will also flag a warning)

```
--- a/usr/src/cmd/zfs/zfs_main.c  Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/cmd/zfs/zfs_main.c  Tue Sep 02 11:15:49 2008 -0400
@@ -1507,7 +1507,7 @@
       boolean_t showversions = B_FALSE;
       int ret;
       upgrade_cbdata_t cb = { 0 };
-      char c;
+      int c;

       /* check options */
       while ((c = getopt(argc, argv, "rvV:a")) != -1) {
```

```
@@ -2246,7 +2246,7 @@
 {
       boolean_t recursive = B_FALSE;
       int ret;
-      char c;
+      int c;
       nvlist_t *props;

       if (nvlist_alloc(&props, NV_UNIQUE_NAME, 0) != 0) {
```

14.1.103 /usr/src/cmd/zlogin/zlogin.c

- Not all platforms default to `signed char` when `char` is declared. Ensure comparison with values less than 0 will work and not automatically evaluated as false.

```
--- a/usr/src/cmd/zlogin/zlogin.c       Tue Sep 02 11:22:56 2008 -0400
+++ b/usr/src/cmd/zlogin/zlogin.c       Thu Sep 25 10:26:09 2008 -0400
@@ -561,7 +561,7 @@
        if (isprint(c)) {
                cc[0] = c;
                cc[1] = '\0';
-       } else if (c >= 0 && c <= 31) { /* ^@ through ^_ */
+       } else if ((signed char) c >= 0 && (signed char) c <= 31) {     /* ^@
through ^_ */
                cc[0] = '^';
                cc[1] = c + '@';
                cc[2] = '\0';
```

14.1.104 /usr/src/cmd/zonecfg/Makefile

- GNU linker does not understand mapfiles used by the Solaris linker so we make them disappear.

```
--- a/usr/src/cmd/zonecfg/Makefile      Tue Sep 02 11:22:56 2008 -0400
+++ b/usr/src/cmd/zonecfg/Makefile      Thu Sep 25 10:26:09 2008 -0400
@@ -34,7 +34,10 @@
 # still export a number of "yy*" (libl) interfaces.  Reduce all other symbols
 # to local scope.
 MAPFILES +=    $(MAPFILE.LEX) $(MAPFILE.NGB)
-MAPOPTS =      $(MAPFILES:%=-M%)
+sparc_MAPOPTS =        $(MAPFILES:%=-M%)
+i386_MAPOPTS = $(MAPFILES:%=-M%)
+s390_MAPOPTS =
+MAPOPTS =      $($(MACH)_MAPOPTS)

 LFLAGS =       -t
 YFLAGS =       -d -b zonecfg_grammar
```

14.1.105 /usr/src/common/bignum/bignum.h

- Add support for System z platform

```
--- a/usr/src/common/bignum/bignum.h    Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/common/bignum/bignum.h    Tue Sep 02 11:15:49 2008 -0400
@@ -34,7 +34,7 @@

 #include <sys/types.h>
```

```
-#ifndef __sparcv9
+#if !defined(__sparcv9) && !defined(__s390x)
 #define     BIGNUM_CHUNK_32
 #else
 #ifndef UMUL64
```

14.1.106 /usr/src/common/bignum/bignumimpl.c

- Correct `#error` statement for gcc-4.

```
--- a/usr/src/common/bignum/bignumimpl.c       Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/common/bignum/bignumimpl.c       Tue Sep 02 11:15:49 2008 -0400
@@ -1369,7 +1369,7 @@
 #else /* ! UMUL64 */

 #if (BIG_CHUNK_SIZE != 32)
-#error Don't use 64-bit chunks without defining UMUL64
+#error Do not use 64-bit chunks without defining UMUL64
 #endif
```

14.1.107 /usr/src/common/crypto/ecc/ecp_224.c

- Temporary work around for a gcc bug. There is a fix but it's not yet been applied to my gcc suite.

```
--- a/usr/src/common/crypto/ecc/ecp_224.c       Tue Sep 02 11:22:56 2008 -0400
+++ b/usr/src/common/crypto/ecc/ecp_224.c       Thu Sep 25 10:26:10 2008 -0400
@@ -279,7 +279,14 @@

                while (r3b < 0) {
                        MP_ADD_CARRY (r0, 1, r0, 0,     carry);
-                       MP_ADD_CARRY (r1, MP_DIGIT_MAX <<32, r1, carry, carry);
+//                     MP_ADD_CARRY (r1, MP_DIGIT_MAX <<32, r1, carry, carry);
+// FIXME - Temporary workaround for GCC bug
+                       {
+                       uint64_t val = MP_DIGIT_MAX;
+                       val = val << 32;
+                       MP_ADD_CARRY (r1, val, r1, carry, carry);
+                       }
+// END OF FIXME
                        MP_ADD_CARRY (r2, MP_DIGIT_MAX, r2, carry, carry);
                        MP_ADD_CARRY (r3, MP_DIGIT_MAX >> 32, r3, carry,
carry);
                        r3b = (int)(r3 >>32);
```

14.1.108 /usr/src/common/devid/devid_scsi.c

- Not all platforms default to `signed char` when `char` is declared. Ensure comparison with values less than 0 will work and not automatically evaluated as false.

```
--- a/usr/src/common/devid/devid_scsi.c Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/common/devid/devid_scsi.c Tue Sep 02 11:15:49 2008 -0400
@@ -1233,7 +1233,7 @@
 #endif  /* _KERNEL */
 {
```

```
        int i;
-       char cl, ch;
+       signed char cl, ch;
        uint64_t tmp;

        if (wwn == NULL || strlen(string) != 16) {
```

14.1.109 /usr/src/common/mapfiles/gen/Makefile

- Allow for platform specific `LDFLAGS` setting using `$($(MACH)_LDFLAGS)` mechanism.

```
diff -r 4f051ff1b998 usr/src/common/mapfiles/gen/Makefile
--- a/usr/src/common/mapfiles/gen/Makefile     Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/common/mapfiles/gen/Makefile     Tue Sep 02 11:15:49 2008 -0400
@@ -41,6 +41,9 @@
 MAIN1=             main.1
 MAIN2=             main.2

+s390_LDFLAGS=      -L/usr/local/lib -L/usr/local/lib/s390x -ldl
+
+LDFLAGS=     $($(MACH)_LDFLAGS)
 TEMPLATE1=   map.noexeglobs.1.template
 TEMPLATE2=   map.noexeglobs.2.template

@@ -96,10 +99,10 @@
 # global.

 %map.noexeglobs:main.c $(TEMPLATE1) $(TEMPLATE2)
-       $(LINK) -o $(MAIN1) -M$(TEMPLATE1) main.c
+       $(LINK) -o $(MAIN1) -M$(TEMPLATE1) main.c $(LDFLAGS)
        $(ELFDUMP) -s -N.dynsym $(MAIN1) | $(EGREP) "WEAK|GLOB" | \
            $(GREP) -v UNDEF | $(NAWK) '{print $$9 }' | $(SORT) > $(SYMS1)
-       $(LINK) -o $(MAIN2) -M$(TEMPLATE2) main.c
+       $(LINK) -o $(MAIN2) -M$(TEMPLATE2) main.c $(LDFLAGS)
        $(ELFDUMP) -s -N.dynsym $(MAIN2) | $(EGREP) "WEAK|GLOB" | \
            $(GREP) -v UNDEF | $(NAWK) '{print $$9 }' | $(SORT) > $(SYMS2)
        $(ECHO) "# GENERATED FILE - DO NOT EDIT"       >  $@
```

14.1.110 /usr/src/common/openssl/crypto/asn1/asn1.h

- Forward fit of openssl fixes.

```
--- a/usr/src/common/openssl/crypto/asn1/asn1.h Tue Sep 02 11:22:56 2008 -0400
+++ b/usr/src/common/openssl/crypto/asn1/asn1.h Thu Sep 25 10:26:10 2008 -0400
@@ -900,23 +900,26 @@

 /* Used to implement other functions */
 void *ASN1_dup(i2d_of_void *i2d, d2i_of_void *d2i, char *x);
+static void * __attribute__((unused)) __ASN1_dup=openssl_fcast(ASN1_dup);
 #define ASN1_dup_of(type,i2d,d2i,x) \
-       ((type *(*)(I2D_OF(type),D2I_OF(type),type
*))openssl_fcast(ASN1_dup))(i2d,d2i,x)
+       ((type *(*)(I2D_OF(type),D2I_OF(type),type *))__ASN1_dup)(i2d,d2i,x)
 #define ASN1_dup_of_const(type,i2d,d2i,x) \
-       ((type *(*)(I2D_OF_const(type),D2I_OF(type),type
*))openssl_fcast(ASN1_dup))(i2d,d2i,x)
+       ((type *(*)(I2D_OF_const(type),D2I_OF(type),type
*))__ASN1_dup)(i2d,d2i,x)
```

```
 void *ASN1_item_dup(const ASN1_ITEM *it, void *x);

 #ifndef OPENSSL_NO_FP_API void *ASN1_d2i_fp(void *(*xnew)(void), d2i_of_void
*d2i, FILE *in, void **x);
+static void * __attribute__((unused))
__ASN1_d2i_fp=openssl_fcast(ASN1_d2i_fp);
 #define ASN1_d2i_fp_of(type,xnew,d2i,in,x) \
-       ((type *(*)(type *(*)(void),D2I_OF(type),FILE *,type
**))openssl_fcast(ASN1_d2i_fp))(xnew,d2i,
in,x)
+       ((type *(*)(type *(*)(void),D2I_OF(type),FILE *,type
**))__ASN1_d2i_fp)(xnew,d2i,in,x)
 void *ASN1_item_d2i_fp(const ASN1_ITEM *it, FILE *in, void *x);
 int ASN1_i2d_fp(i2d_of_void *i2d,FILE *out,void *x);
+static void * __attribute__((unused))
__ASN1_i2d_fp=openssl_fcast(ASN1_i2d_fp);
 #define ASN1_i2d_fp_of(type,i2d,out,x) \
-       ((int (*)(I2D_OF(type),FILE *,type
*))openssl_fcast(ASN1_i2d_fp))(i2d,out,x)
+       ((int (*)(I2D_OF(type),FILE *,type *))__ASN1_i2d_fp)(i2d,out,x)
 #define ASN1_i2d_fp_of_const(type,i2d,out,x) \
-       ((int (*)(I2D_OF_const(type),FILE *,type
*))openssl_fcast(ASN1_i2d_fp))(i2d,out,x)
+       ((int (*)(I2D_OF_const(type),FILE *,type *))__ASN1_i2d_fp)(i2d,out,x)
 int ASN1_item_i2d_fp(const ASN1_ITEM *it, FILE *out, void *x);
 int ASN1_STRING_print_ex_fp(FILE *fp, ASN1_STRING *str, unsigned long flags);
 #endif
@@ -925,14 +928,16 @@

 #ifndef OPENSSL_NO_BIO void *ASN1_d2i_bio(void *(*xnew)(void), d2i_of_void
*d2i, BIO *in, void **x);
+static void * __attribute__((unused))
__ASN1_d2i_bio=openssl_fcast(ASN1_d2i_bio);
 #define ASN1_d2i_bio_of(type,xnew,d2i,in,x) \
-       ((type *(*)(type *(*)(void),D2I_OF(type),BIO *,type
**))openssl_fcast(ASN1_d2i_bio))(xnew,d2i,
in,x)
+       ((type *(*)(type *(*)(void),D2I_OF(type),BIO *,type
**))__ASN1_d2i_bio)(xnew,d2i,in,x)
 void *ASN1_item_d2i_bio(const ASN1_ITEM *it, BIO *in, void *x);
 int ASN1_i2d_bio(i2d_of_void *i2d,BIO *out, unsigned char *x);
+static void * __attribute__((unused))
__ASN1_i2d_bio=openssl_fcast(ASN1_i2d_bio);
 #define ASN1_i2d_bio_of(type,i2d,out,x) \
-       ((int (*)(I2D_OF(type),BIO *,type
*))openssl_fcast(ASN1_i2d_bio))(i2d,out,x)
+       ((int (*)(I2D_OF(type),BIO *,type *))__ASN1_i2d_bio)(i2d,out,x)
 #define ASN1_i2d_bio_of_const(type,i2d,out,x) \
-       ((int (*)(I2D_OF_const(type),BIO *,const type
*))openssl_fcast(ASN1_i2d_bio))(i2d,out,x)
+       ((int (*)(I2D_OF_const(type),BIO *,const type
*))__ASN1_i2d_bio)(i2d,out,x)
 int ASN1_item_i2d_bio(const ASN1_ITEM *it, BIO *out, void *x);
 int ASN1_UTCTIME_print(BIO *fp,ASN1_UTCTIME *a);
 int ASN1_GENERALIZEDTIME_print(BIO *fp,ASN1_GENERALIZEDTIME *a);
@@ -975,8 +980,9 @@
 void *ASN1_item_unpack(ASN1_STRING *oct, const ASN1_ITEM *it);
 ASN1_STRING *ASN1_pack_string(void *obj, i2d_of_void *i2d,
                               ASN1_OCTET_STRING **oct);
+static void * __attribute__((unused))
__ASN1_pack_string=openssl_fcast(ASN1_pack_string);
 #define ASN1_pack_string_of(type,obj,i2d,oct) \
```

```
-       ((ASN1_STRING *(*)(type *,I2D_OF(type),ASN1_OCTET_STRING
**))openssl_fcast(ASN1_pack_string))(obj,i2d,oct)
+       (ASN1_pack_string((obj), (i2d_of_void *)(i2d), (oct)))
 ASN1_STRING *ASN1_item_pack(void *obj, const ASN1_ITEM *it, ASN1_OCTET_STRING
**oct);

 void ASN1_STRING_set_default_mask(unsigned long mask);
```

14.1.111 /usr/src/common/openssl/crypto/ocsp/ocsp.h

- Forward fit of openssl fixes.

```
--- a/usr/src/common/openssl/crypto/ocsp/ocsp.h Tue Sep 02 11:22:56 2008 -0400
+++ b/usr/src/common/openssl/crypto/ocsp/ocsp.h Thu Sep 25 10:26:10 2008 -0400
@@ -468,8 +468,9 @@

 ASN1_STRING *ASN1_STRING_encode(ASN1_STRING *s, i2d_of_void *i2d,
                                 void *data, STACK_OF(ASN1_OBJECT) *sk);
+static void * __attribute__((unused))
__ASN1_STRING_encode=openssl_fcast(ASN1_STRING_encode);
 #define ASN1_STRING_encode_of(type,s,i2d,data,sk) \-((ASN1_STRING
*(*)(ASN1_STRING *,I2D_OF(type),type *,STACK_OF(ASN1_OBJECT)
*))openssl_fcast(ASN1_STRI
NG_encode))(s,i2d,data,sk)
+       (ASN1_STRING_encode((s), (i2d_of_void *)(i2d), (data),
(STACK_OF(ASN1_OBJECT) *)(sk)))
 X509_EXTENSION *OCSP_crlID_new(char *url, long *n, char *tim);
```

14.1.112 /usr/src/common/openssl/crypto/pem/pem.h

- Forward fit of openssl fixes.

```
--- a/usr/src/common/openssl/crypto/pem/pem.h   Tue Sep 02 11:22:56 2008 -
0400+++ b/usr/src/common/openssl/crypto/pem/pem.h   Thu Sep 25 10:26:10 2008 -
0400
@@ -220,19 +220,22 @@
 #define IMPLEMENT_PEM_read_fp(name, type, str, asn1) \ type
*PEM_read_##name(FILE *fp, type **x, pem_password_cb *cb, void *u)\
 { \
-return(((type *(*)(D2I_OF(type),char *,FILE *,type **,pem_password_cb *,void
*))openssl_fcast(PEM_ASN
1_read))(d2i_##asn1, str,fp,x,cb,u)); \
+static void *__PEM_ASN1_read=openssl_fcast(PEM_ASN1_read); \
+return(((type *(*)(D2I_OF(type),char *,FILE *,type **,pem_password_cb *,void
*))__PEM_ASN1_read)(d2i_
##asn1, str,fp,x,cb,u)); \ } \
 #define IMPLEMENT_PEM_write_fp(name, type, str, asn1) \
 int PEM_write_##name(FILE *fp, type *x) \ { \
-return(((int (*)(I2D_OF(type),const char *,FILE *,type *, const EVP_CIPHER
*,unsigned char *,int, pem
_password_cb *,void
*))openssl_fcast(PEM_ASN1_write))(i2d_##asn1,str,fp,x,NULL,NULL,0,NULL,NULL));
\
+static void *__PEM_ASN1_write=openssl_fcast(PEM_ASN1_write); \
+return(((int (*)(I2D_OF(type),const char *,FILE *,type *, const EVP_CIPHER
*,unsigned char *,int, pem
_password_cb *,void
*))__PEM_ASN1_write)(i2d_##asn1,str,fp,x,NULL,NULL,0,NULL,NULL)); \
 }
```

```
 #define IMPLEMENT_PEM_write_fp_const(name, type, str, asn1) \ int
PEM_write_##name(FILE *fp, const type *x) \ { \-return(((int
(*)(I2D_OF_const(type),const char *,FILE *, const type *, const EVP_CIPHER
*,unsigned ch
ar *,int, pem_password_cb *,void
*))openssl_fcast(PEM_ASN1_write))(i2d_##asn1,str,fp,x,NULL,NULL,0,NUL
L,NULL)); \+static void *__PEM_ASN1_write=openssl_fcast(PEM_ASN1_write); \
+return(((int (*)(I2D_OF_const(type),const char *,FILE *, const type *, const
EVP_CIPHER *,unsigned ch
ar *,int, pem_password_cb *,void
*))__PEM_ASN1_write)(i2d_##asn1,str,fp,x,NULL,NULL,0,NULL,NULL)); \
 }

 #define IMPLEMENT_PEM_write_cb_fp(name, type, str, asn1) \
@@ -240,7 +243,8 @@
             unsigned char *kstr, int klen, pem_password_cb *cb, \
                  void *u) \        { \-       return(((int
(*)(I2D_OF(type),const char *,FILE *,type *, const EVP_CIPHER *,unsigned char
*,i
nt, pem_password_cb *,void
*))openssl_fcast(PEM_ASN1_write))(i2d_##asn1,str,fp,x,enc,kstr,klen,cb,u));
 \+       static void *__PEM_ASN1_write=openssl_fcast(PEM_ASN1_write); \
+       return(((int (*)(I2D_OF(type),const char *,FILE *,type *, const
EVP_CIPHER *,unsigned char *,int, pem_password_cb *,void
*))__PEM_ASN1_write)(i2d_##asn1,str,fp,x,enc,kstr,klen,cb,u)); \
        }

 #define IMPLEMENT_PEM_write_cb_fp_const(name, type, str, asn1) \
@@ -248,7 +252,8 @@
             unsigned char *kstr, int klen, pem_password_cb *cb, \
                  void *u) \
        { \
-       return(((int (*)(I2D_OF_const(type),const char *,FILE *,type *, const
EVP_CIPHER *,unsigned char *,int, pem_password_cb *,void
*))openssl_fcast(PEM_ASN1_write))(i2d_##asn1,str,fp,x,enc,kstr,klen,cb,u)); \
+       static void *__PEM_ASN1_write=openssl_fcast(PEM_ASN1_write); \
+       return(((int (*)(I2D_OF_const(type),const char *,FILE *,type *, const
EVP_CIPHER *,unsigned char *,int, pem_password_cb *,void
*))__PEM_ASN1_write)(i2d_##asn1,str,fp,x,enc,kstr,klen,cb,u)); \
        }

 #endif
@@ -256,33 +261,38 @@
 #define IMPLEMENT_PEM_read_bio(name, type, str, asn1) \
 type *PEM_read_bio_##name(BIO *bp, type **x, pem_password_cb *cb, void *u)\
 { \
-return(((type *(*)(D2I_OF(type),const char *,BIO *,type **,pem_password_cb
*,void *))openssl_fcast(PEM_ASN1_read_bio))(d2i_##asn1, str,bp,x,cb,u)); \
+static void *__PEM_ASN1_read_bio=openssl_fcast(PEM_ASN1_read_bio); \
+return(((type *(*)(D2I_OF(type),const char *,BIO *,type **,pem_password_cb
*,void *))__PEM_ASN1_read_bio)(d2i_##asn1, str,bp,x,cb,u)); \
 }

 #define IMPLEMENT_PEM_write_bio(name, type, str, asn1) \
 int PEM_write_bio_##name(BIO *bp, type *x) \
 { \
-return(((int (*)(I2D_OF(type),const char *,BIO *,type *, const EVP_CIPHER
*,unsigned char *,int, pem_password_cb *,void
*))openssl_fcast(PEM_ASN1_write_bio))(i2d_##asn1,str,bp,x,NULL,NULL,0,NULL,NULL
)); \
+static void *__PEM_ASN1_write_bio=openssl_fcast(PEM_ASN1_write_bio); \
```

```
+return(((int (*)(I2D_OF(type),const char *,BIO *,type *, const EVP_CIPHER
*,unsigned char *,int, pem_password_cb *,void
*))__PEM_ASN1_write_bio)(i2d_##asn1,str,bp,x,NULL,NULL,0,NULL,NULL)); \
 }

#define IMPLEMENT_PEM_write_bio_const(name, type, str, asn1) \
 int PEM_write_bio_##name(BIO *bp, const type *x) \
 { \
-return(((int (*)(I2D_OF_const(type),const char *,BIO *,const type *, const
EVP_CIPHER *,unsigned char *,int, pem_password_cb *,void
*))openssl_fcast(PEM_ASN1_write_bio))(i2d_##asn1,str,bp,x,NULL,NULL,0,NULL,NULL
)); \
+static void *__PEM_ASN1_write_bio=openssl_fcast(PEM_ASN1_write_bio); \
+return(((int (*)(I2D_OF_const(type),const char *,BIO *,const type *, const
EVP_CIPHER *,unsigned char *,int, pem_password_cb *,void
*))__PEM_ASN1_write_bio)(i2d_##asn1,str,bp,x,NULL,NULL,0,NULL,NULL)); \
 }

 #define IMPLEMENT_PEM_write_cb_bio(name, type, str, asn1) \
 int PEM_write_bio_##name(BIO *bp, type *x, const EVP_CIPHER *enc, \
             unsigned char *kstr, int klen, pem_password_cb *cb, void *u) \
        { \
-       return(((int (*)(I2D_OF(type),const char *,BIO *,type *,const
EVP_CIPHER *,unsigned char *,int,pem_password_cb *,void
*))openssl_fcast(PEM_ASN1_write_bio))(i2d_##asn1,str,bp,x,enc,kstr,klen,cb,u));
\
+       static void *__PEM_ASN1_write_bio=openssl_fcast(PEM_ASN1_write_bio); \
+       return(((int (*)(I2D_OF(type),const char *,BIO *,type *,const
EVP_CIPHER *,unsigned char *,int,pem_password_cb *,void
*))__PEM_ASN1_write_bio)(i2d_##asn1,str,bp,x,enc,kstr,klen,cb,u)); \
        }

 #define IMPLEMENT_PEM_write_cb_bio_const(name, type, str, asn1) \
 int PEM_write_bio_##name(BIO *bp, type *x, const EVP_CIPHER *enc, \
             unsigned char *kstr, int klen, pem_password_cb *cb, void *u) \
        { \
-       return(((int (*)(I2D_OF_const(type),const char *,BIO *,type *,const
EVP_CIPHER *,unsigned char *,int,pem_password_cb *,void
*))openssl_fcast(PEM_ASN1_write_bio))(i2d_##asn1,str,bp,x,enc,kstr,klen,cb,u));
\
+       static void *__PEM_ASN1_write_bio=openssl_fcast(PEM_ASN1_write_bio); \
+       return(((int (*)(I2D_OF_const(type),const char *,BIO *,type *,const
EVP_CIPHER *,unsigned char *,int,pem_password_cb *,void
*))__PEM_ASN1_write_bio)(i2d_##asn1,str,bp,x,enc,kstr,klen,cb,u)); \
        }

 #define IMPLEMENT_PEM_write(name, type, str, asn1) \
@@ -545,13 +555,15 @@
             pem_password_cb *cb, void *u);
 void * PEM_ASN1_read_bio(d2i_of_void *d2i, const char *name, BIO *bp,
                          void **x, pem_password_cb *cb, void *u);
+static void * __attribute__((unused))
__PEM_ASN1_read_bio=openssl_fcast(PEM_ASN1_read_bio);
 #define PEM_ASN1_read_bio_of(type,d2i,name,bp,x,cb,u) \
-((type *(*)(D2I_OF(type),const char *,BIO *,type **,pem_password_cb *,void
*))openssl_fcast(PEM_ASN1_read_bio))(d2i,name,bp,x,cb,u)
+((type *(*)(D2I_OF(type),const char *,BIO *,type **,pem_password_cb *,void
*))__PEM_ASN1_read_bio)(d2i,name,bp,x,cb,u)
 int    PEM_ASN1_write_bio(i2d_of_void *i2d,const char *name,BIO *bp,char *x,
                           const EVP_CIPHER *enc,unsigned char *kstr,int klen,
                           pem_password_cb *cb, void *u);
+static void * __attribute__((unused))
__PEM_ASN1_write_bio=openssl_fcast(PEM_ASN1_write_bio);
```

```
 #define PEM_ASN1_write_bio_of(type,i2d,name,bp,x,enc,kstr,klen,cb,u) \
-       ((int (*)(I2D_OF(type),const char *,BIO *,type *, const EVP_CIPHER
*,unsigned char *,int, pem_password_cb *,void
*))openssl_fcast(PEM_ASN1_write_bio))(i2d,name,bp,x,enc,kstr,klen,cb,u)
+       ((int (*)(I2D_OF(type),const char *,BIO *,type *, const EVP_CIPHER
*,unsigned char *,int, pem_password_cb *,void
*))__PEM_ASN1_write_bio)(i2d,name,bp,x,enc,kstr,klen,cb,u)

 STACK_OF(X509_INFO) *  PEM_X509_INFO_read_bio(BIO *bp, STACK_OF(X509_INFO)
*sk, pem_password_cb *cb, void *u);
 int    PEM_X509_INFO_write_bio(BIO *bp,X509_INFO *xi, EVP_CIPHER *enc,
```

14.1.113 /usr/src/lib/gss_mechs/mech_krb5/mech/accept_sec_context.c

- Not all platforms default to `signed char` when `char` is declared. Ensure comparison with values less than 0 will work and not automatically evaluated as false.

```
--- a/usr/src/lib/gss_mechs/mech_krb5/mech/accept_sec_context.c Tue Sep 02
11:22:56 2008 -0400
+++ b/usr/src/lib/gss_mechs/mech_krb5/mech/accept_sec_context.c Thu Sep 25
10:26:10 2008 -0400
@@ -309,7 +309,7 @@
 {
    krb5_context context;
    unsigned char *ptr, *ptr2;
-   signed char *sptr;
+   char *sptr;
    long tmp;
    size_t md5len;
    int bigend;
@@ -422,7 +422,7 @@
        goto fail;
    }

-   sptr = (signed char *) ptr;
+   sptr = (char *) ptr;
    TREAD_STR(sptr, ap_req.data, ap_req.length);

    /*
```

14.1.114 /usr/src/common/util/memstr.c

- Correct syntax error

```
--- a/usr/src/common/util/memstr.c      Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/common/util/memstr.c      Tue Sep 02 11:15:49 2008 -0400
@@ -61,7 +61,7 @@
       }
       return (s);
 #else
-      extern void * _memset(void *, int size_t);
+      extern void * _memset(void *, int, size_t);

       return (_memset(s, c, n));
 #endif
```

14.1.115 /usr/src/head/arpa/nameser_compat.h

- Add support for System z platform.

```
--- a/usr/src/head/arpa/nameser_compat.h       Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/head/arpa/nameser_compat.h       Tue Sep 02 11:15:49 2008 -0400
@@ -99,7 +99,7 @@
     defined(apollo) || defined(__convex__) || defined(_CRAY) || \
     defined(__hppa) || defined(__hp9000) || \
     defined(__hp9000s300) || defined(__hp9000s700) || \
-    defined(__hp3000s900) || defined(MPE) || defined(__s390) \
+    defined(__hp3000s900) || defined(MPE) || defined(__s390) || \
     defined(BIT_ZERO_ON_LEFT) || defined(m68k) || \
       (defined(__Lynx__) && \
       (defined(__68k__) || defined(__sparc__) || defined(__powerpc__)))
```

14.1.116 /usr/src/head/inttypes.h

- Add support for System z platform (match `wchar_t` to what gcc uses internally)

```
--- a/usr/src/head/inttypes.h     Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/head/inttypes.h     Tue Sep 02 11:15:49 2008 -0400
@@ -68,7 +68,7 @@
 #if !defined(__cplusplus) || (__cplusplus < 199711L && !defined(__GNUG__))
 #ifndef _WCHAR_T
 #define      _WCHAR_T
-#if defined(_LP64)
+#if defined(_LP64) || defined(__s390__)
 typedef      int    wchar_t;
 #else
 typedef      long   wchar_t;
```

14.1.117 /usr/src/head/iso/stdlib_iso.h

- Add support for System z platform (match `wchar_t` to what gcc uses internally)

```
--- a/usr/src/head/iso/stdlib_iso.h      Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/head/iso/stdlib_iso.h      Tue Sep 02 11:15:49 2008 -0400
@@ -110,7 +110,7 @@
 #if !defined(__cplusplus) || (__cplusplus < 199711L && !defined(__GNUG__))
 #ifndef _WCHAR_T
 #define      _WCHAR_T
-#if defined(_LP64)
+#if defined(_LP64) || defined(__s390__)
 typedef      int    wchar_t;
 #else
 typedef long wchar_t;
```

14.1.118 /usr/src/head/regex.h

- Add support for System z platform (match `wchar_t` to what gcc uses internally)

```
--- a/usr/src/head/regex.h Tue Aug 19 11:19:58 2008 -0400
```

```
+++ b/usr/src/head/regex.h Tue Sep 02 11:15:49 2008 -0400
@@ -58,7 +58,7 @@
 #if !defined(__cplusplus) || (__cplusplus < 199711L && !defined(__GNUG__))
 #ifndef _WCHAR_T
 #define      _WCHAR_T
-#if defined(_LP64)
+#if defined(_LP64) || defined(__s390__)
 typedef int  wchar_t;
 #else
 typedef long    wchar_t;
```

14.1.119 /usr/src/head/stddef.h

- Add support for System z platform (match `wchar_t` to what gcc uses internally)

```
--- a/usr/src/head/stddef.h      Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/head/stddef.h      Tue Sep 02 11:15:49 2008 -0400
@@ -65,7 +65,7 @@
 #if !defined(__cplusplus) || (__cplusplus < 199711L && !defined(__GNUG__))
 #ifndef _WCHAR_T
 #define      _WCHAR_T
-#if defined(_LP64)
+#if defined(_LP64) || defined(__s390__)
 typedef int  wchar_t;
 #else
 typedef long    wchar_t;
```

14.1.120 /usr/src/lib/Makefile.lib

- GNU linker does not understand mapfiles used by the Solaris linker so we make them disappear
- GCC build requires `-zcombreloc` and `-zmuldefs` flags but not `-zdefs`, so use the `$($(MACH)_ZFLAGS)` to specify them on a per-platform basis
- Add System z platform support
- Don't search `/usr/include` when compiling (we need to look in `$ROOT/usr/include`)

```
diff -r 4f051ff1b998 usr/src/lib/Makefile.lib
--- a/usr/src/lib/Makefile.lib    Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/Makefile.lib    Tue Sep 02 11:15:49 2008 -0400
@@ -161,6 +161,29 @@
 LINTOUT=     lint.out
 ARFLAGS=     r
 SONAME=             $(DYNLIB)
+
+#
+# GNU ld doesn't understand mapfiles used by the Solaris linker so we have
+# to make them disappear
+#
+#s390_MAPFLAG=   $(MAPFILES:%=-Wl,--version-script %)
+# $(MAPFILE.PGA:%=-Wl,--version-script %) $(MAPFILE.NED:%=-Wl,--version-script
%)
+#s390x_MAPFLAG=  $(MAPFILES:%=-Wl,--version-script %)
+# $(MAPFILE.PGA:%=-Wl,--version-script %) $(MAPFILE.NED:%=-Wl,--version-script
%)
+sparcv9_MAPFLAG=   $(MAPFILES:%=-M%) $(MAPFILE.PGA:%=-M%) $(MAPFILE.NED:%=-M%)
```

```
+sparc_MAPFLAG=            $(MAPFILES:%=-M%) $(MAPFILE.PGA:%=-M%)
$(MAPFILE.NED:%=-M%)
+i386_MAPFLAG=             $(MAPFILES:%=-M%) $(MAPFILE.PGA:%=-M%)
$(MAPFILE.NED:%=-M%)
+amd64_MAPFLAG=            $(MAPFILES:%=-M%) $(MAPFILE.PGA:%=-M%)
$(MAPFILE.NED:%=-M%)
+MAPFLAGS= $($(MACH)_MAPFLAG)
+
+
+sparc_ZFLAGS=      $(ZTEXT) $(ZDEFS)
+i386_ZFLAGS=$(ZTEXT) $(ZDEFS)
+amd64_ZFLAGS=      $(ZTEXT) $(ZDEFS)
+s390_ZFLAGS=$(ZTEXT) $(ZCOMBRELOC) $(ZMULDEFS) -L$(ROOT)/lib
+s390x_ZFLAGS=      $(ZTEXT) $(ZCOMBRELOC) $(ZMULDEFS) -L$(ROOT)/lib/s390x
+ZFLAGS=            $($(MACH)_ZFLAGS)
+
 # For most libraries, we should be able to resolve all symbols at link time,
 # either within the library or as dependencies, all text should be pure, and
 # combining relocations into one relocation table reduces startup costs.
@@ -168,8 +191,7 @@


 HSONAME=     -h$(SONAME)
-DYNFLAGS=    $(HSONAME) $(ZTEXT) $(ZDEFS) $(BDIRECT) \
-             $(MAPFILES:%=-M%) $(MAPFILE.PGA:%=-M%) $(MAPFILE.NED:%=-M%)
+DYNFLAGS=    $(HSONAME) $(ZFLAGS) $(BDIRECT) $(MAPFLAGS)

 LDLIBS=            $(LDLIBS.lib)

@@ -222,10 +244,13 @@
 $(PICS)  := sparcv9_CFLAGS += -xregs=no%appl $(sparcv9_C_PICFLAGS)
 $(PICS)  := i386_CFLAGS += $(i386_C_PICFLAGS)
 $(PICS)  := amd64_CFLAGS += $(amd64_C_PICFLAGS)
+$(PICS)  := s390_CFLAGS += $(s390_C_PICFLAGS)
+$(PICS)  := s390x_CFLAGS += $(s390x_C_PICFLAGS)
 $(PICS)  := CCFLAGS += $(CC_PICFLAGS)
 $(PICS)  := CPPFLAGS += -DPIC -D_REENTRANT
 $(PICS)  := sparcv9_CCFLAGS += -xregs=no%appl $(sparcv9_CC_PICFLAGS)
 $(PICS)  := amd64_CCFLAGS += $(amd64_CC_PICFLAGS)
+$(PICS)  := s390x_CCFLAGS += $(s390x_CC_PICFLAGS)
 $(PICS)  := CFLAGS += $(CTF_FLAGS)
 $(PICS)       :=    CFLAGS64 += $(CTF_FLAGS)
 $(PICS)  := CTFCONVERT_POST = $(CTFCONVERT_O)
@@ -233,18 +258,21 @@

 $(LINTLIB):=LOG = -DLOGGING
 $(LIBRARY):=AROBJS = $(OBJS)
-$(LIBRARY):=DIR = objs
+#$(LIBRARY):=      DIR = objs
+$(LIBRARY):=DIR = pics
 $(DYNLIB):= DIR = pics
 $(DYNLIBCCC):=     DIR = pics
+
+s390_CFLAGS  += -_gcc=-nostdinc
+s390x_CFLAGS += -_gcc=-nostdinc

 SONAMECCC=   $(DYNLIBCCC)
 HSONAMECCC=  -h $(SONAMECCC)
 #
 # Keep in sync with the standard DYNFLAGS
 #
-$(DYNLIBCCC):=     DYNFLAGS = $(HSONAMECCC) $(ZTEXT) $(ZDEFS) \
-             $(MAPFILES:%=-M%) $(MAPFILE.PGA:%=-M%) $(MAPFILE.NED:%=-M%) \
```

```
-             $(BDIRECT) $(NORUNPATH)
+$(DYNLIBCCC):=      DYNFLAGS = $(HSONAMECCC) $(ZFLAGS) \
+             $(MAPFLAGS) $(BDIRECT) $(NORUNPATH)


 # build rule for "portable" source
```

14.1.121 /usr/src/lib/brand/native/zone/Makefile

- When cross-building `libxml2` may not be installed on the base system and may be placed in `$ROOT`
- Use `$($(MACH)_XLDLIBS)` mechanism to allow a platform to specify additional library flags

```
--- a/usr/src/lib/brand/native/zone/Makefile   Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/brand/native/zone/Makefile   Tue Sep 02 11:15:49 2008 -0400
@@ -38,8 +38,9 @@
 include $(SRC)/cmd/Makefile.cmd
 include ../../Makefile.brand

-CPPFLAGS += -I/usr/include/libxml2 -D_REENTRANT
-LDLIBS += -lzonecfg -luutil
+CPPFLAGS += -I$(ROOT)/usr/include/libxml2 -I/usr/include/libxml2 -D_REENTRANT
+s390_XLDLIBS +=    -_gcc="-Wl,-rpath-link=$(ROOT)/usr/lib"
+LDLIBS += -lzonecfg -luutil $($(MACH)_XLDLIBS)

 POFILES=    $(PROGS:%=%.po)
 POFILE=             native_zone.po
```

14.1.122 /usr/src/lib/fm/libfmd_snmp/Makefile.com

- Use `$($(MACH)_SNMPDIR)` to specify a platform specific library directory to search

```
--- a/usr/src/lib/fm/libfmd_snmp/Makefile.com  Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/fm/libfmd_snmp/Makefile.com  Tue Sep 02 11:15:50 2008 -0400
@@ -52,7 +52,10 @@
 CFLAGS64 += $(CCVERBOSE) $(C_BIGPICFLAGS)

 # No lint libraries are delivered for Net-SNMP yet
-SNMPLIBS = -L$(SFWLIBDIR) -lnetsnmp -lnetsnmphelpers -lnetsnmpagent
+sparc_SNMPDIR = -L$(SFWLIBDIR)
+i38l_SNMPDIR =     -L$(SFWLIBDIR)
+s390_SNMPDIR =
+SNMPLIBS = $($(MACH)_SNMPDIR) -lnetsnmp -lnetsnmphelpers -lnetsnmpagent
 lint := SNMPLIBS=

 LDLIBS += $(MACH_LDLIBS)
```

14.1.123 /usr/src/lib/fm/topo/libtopo/Makefile.com

- When cross-building `libxml2` may not be installed on the base system and may be placed in `$ROOT`

```
--- a/usr/src/lib/fm/topo/libtopo/Makefile.com Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/fm/topo/libtopo/Makefile.com Tue Sep 02 11:15:50 2008 -0400
@@ -75,7 +75,7 @@
```

```
 CLEANFILES += $(SRCDIR)/topo_error.c $(SRCDIR)/topo_tables.c

-CPPFLAGS += -I../common -I/usr/include/libxml2 -I.
+CPPFLAGS += -I../common -I$(ROOT)/usr/include/libxml2 -I/usr/include/libxml2 -
I.
 CFLAGS += $(CCVERBOSE) $(C_BIGPICFLAGS)
 CFLAGS += -D_POSIX_PTHREAD_SEMANTICS
 CFLAGS64 += $(CCVERBOSE) $(C_BIGPICFLAGS)
```

14.1.124 /usr/src/lib/fm/topo/modules/Makefile.plugin

- Use `$($(MACH)_APIMAP)` to specify a platform specific mapfile (or none at all in the case of the GNU linker)

```
--- a/usr/src/lib/fm/topo/modules/Makefile.plugin    Tue Aug 19 11:19:58 2008 -
0400
+++ b/usr/src/lib/fm/topo/modules/Makefile.plugin    Tue Sep 02 11:15:50 2008 -
0400
@@ -60,7 +60,10 @@
 LINTFLAGS = -msux
 LINTFILES = $(SRCS:%.c=%.ln)

-APIMAP = ../../../libtopo/common/topo_mod.map
+sparc_APIMAP =     -M../../../libtopo/common/topo_mod.map
+i386_APIMAP =      -M../../../libtopo/common/topo_mod.map
+s390_APIMAP =
+APIMAP = $($(MACH)_APIMAP)
 MAPFILES =         # use APIMAP instead

 CFLAGS += $(CTF_FLAGS) $(CCVERBOSE) $(XSTRCONST) $(CC_PICFLAGS)
@@ -68,7 +71,7 @@

 CPPFLAGS += -I.
 CPPFLAGS += -D_POSIX_PTHREAD_SEMANTICS -D_REENTRANT
-LDFLAGS += $(ZIGNORE) -M$(APIMAP)
+LDFLAGS += $(ZIGNORE) $(APIMAP)
 LDLIBS += -L$(ROOTLIBDIR)/fm -ltopo -lnvpair -lc
 DYNFLAGS += -R/usr/lib/fm
```

14.1.125 /usr/src/lib/gss_mechs/mech_krb5/crypto/des/string2key.c

- Make gcc happy by correct casting for pointer arithmetic.

```
--- a/usr/src/lib/gss_mechs/mech_krb5/crypto/des/string2key.c      Tue Aug 19
11:19:58 2008 -0400
+++ b/usr/src/lib/gss_mechs/mech_krb5/crypto/des/string2key.c      Tue Sep 02
11:15:50 2008 -0400
@@ -85,7 +85,7 @@
       at = strchr(afssalt.data, '@');
       if (at) {
           *at = 0;
-          afssalt.length = at - afssalt.data;
+          afssalt.length = (uintptr_t) at - (uintptr_t) afssalt.data;
       } else
           afssalt.length = strlen(afssalt.data);
       return mit_afs_string_to_key(context, keyblock, data, &afssalt);
```

14.1.126 /usr/src/lib/gss_mechs/mech_krb5/mech/accept_sec_context.c

- `option.data` is used in comparisons with values < 0. Not all platforms default to `signed char` for `char` declarations

```
--- a/usr/src/lib/gss_mechs/mech_krb5/mech/accept_sec_context.c    Tue Aug 19
11:19:58 2008 -0400
+++ b/usr/src/lib/gss_mechs/mech_krb5/mech/accept_sec_context.c    Tue Sep 02
11:15:50 2008 -0400
@@ -727,7 +727,7 @@
                macro uses ptr as both lvalue and rvalue */

              TREAD_STR(ptr, ptr2, option.length);
-             option.data = (char *) ptr2;
+             option.data = (signed char *) ptr2;

              i -= option.length;
```

14.1.127 /usr/src/lib/libast/common/obsolete/spawn.c

- Remove redundant "`extern`" from function .

```
diff -r 4f051ff1b998 usr/src/lib/libast/common/obsolete/spawn.c
--- a/usr/src/lib/libast/common/obsolete/spawn.c      Tue Aug 19 11:19:58 2008 -
0400
+++ b/usr/src/lib/libast/common/obsolete/spawn.c      Tue Sep 02 11:15:50 2008 -
0400
@@ -137,7 +137,7 @@

 #else

-extern pid_t
+pid_t
 spawnlp(const char* name, const char* arg, ...)
 {
       va_list       ap;
```

14.1.128 /usr/src/lib/libbrand/Makefile.com

- When cross-building `libxml2` may not be installed on the base system and may be placed in `$ROOT`

```
--- a/usr/src/lib/libbrand/Makefile.com Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libbrand/Makefile.com Tue Sep 02 11:15:50 2008 -0400
@@ -34,7 +34,7 @@
 LIBS=         $(DYNLIB) $(LINTLIB)
 LDLIBS +=     -lc
 $(LINTLIB) := SRCS=       $(SRCDIR)/$(LINTSRC)
-CPPFLAGS += -I/usr/include/libxml2 -I$(SRCDIR) -D_REENTRANT
+CPPFLAGS += -I$(ROOT)/usr/include/libxml2 -I/usr/include/libxml2 -I$(SRCDIR) -
D_REENTRANT
 $(DYNLIB) := LDLIBS += -lxml2

 SRCDIR=             ../common
```

14.1.129 /usr/src/lib/libbpm/common/au_tp.c

- Not all platforms default to `signed char` for `char` declarations.

```
--- a/usr/src/lib/libbsm/common/au_to.c Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libbsm/common/au_to.c Tue Sep 02 11:15:50 2008 -0400
@@ -321,7 +321,7 @@
        /*
         * Check validity of print type
         */
-       if (unit_print < AUP_BINARY || unit_print > AUP_STRING)
+       if ((signed char) unit_print < AUP_BINARY || unit_print > AUP_STRING)
               return (NULL);

        switch (unit_type) {
```

14.1.130 /usr/src/lib/libc/Makefile.targ

- System z defines several atomic functions in 'C' files so a rule needs to be added to get them built.

```
--- a/usr/src/lib/libc/Makefile.targ    Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libc/Makefile.targ    Tue Sep 02 11:15:50 2008 -0400
@@ -268,6 +268,10 @@
        $(BUILD.s)
        $(POST_PROCESS_O)

+pics/%.o: $(SRC)/common/atomic/$(TARGETMACH)/%.c
+       $(COMPILE.c) -o $@ $<
+       $(POST_PROCESS_O)
+
 $(COMOBJS:%=pics/%): $(SRC)/common/util/$$(@F:.o=.c)
        $(COMPILE.c) -o $@ $(SRC)/common/util/$(@F:.o=.c)
        $(POST_PROCESS_O)
```

14.1.131 /usr/src/libc/common/sys/access.s

- GNU as is a single pass assembler that can't cope with forward references of labels in `.equ` type statements.

```
--- a/usr/src/lib/libc/common/sys/access.s      Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libc/common/sys/access.s      Tue Sep 02 11:15:50 2008 -0400
@@ -48,7 +48,5 @@
        SET_SIZE(access)

 # ifdef __GNUC__
-# undef access
-
-       ANSI_PRAGMA_WEAK2(access,_access,function)
+       ANSI_PRAGMA_WEAK2(_access,access,function)
 # endif
```

14.1.132 /usr/src/libc/common/sys/acct.s

- GNU as is a single pass assembler that can't cope with forward references of labels in `.equ` type statements.

```
--- a/usr/src/lib/libc/common/sys/acct.s       Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libc/common/sys/acct.s       Tue Sep 02 11:15:50 2008 -0400
@@ -48,7 +48,5 @@
       SET_SIZE(acct)

 #ifdef __GNUC__
-# undef acct
-
-       ANSI_PRAGMA_WEAK2(acct,_acct, function)
+       ANSI_PRAGMA_WEAK2(_acct,acct, function)
 #endif
```

14.1.133 /usr/src/libc/common/sys/alarm.s

- GNU as is a single pass assembler that can't cope with forward references of labels in `.equ` type statements.

```
--- a/usr/src/lib/libc/common/sys/alarm.s      Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libc/common/sys/alarm.s      Tue Sep 02 11:15:50 2008 -0400
@@ -48,6 +48,5 @@
       SET_SIZE(alarm)

 #ifdef __GNUC__
-# undef alarm
-       ANSI_PRAGMA_WEAK2(alarm,_alarm,function)
+       ANSI_PRAGMA_WEAK2(_alarm,alarm,function)
 #endif
```

14.1.134 /usr/src/libc/common/sys/chdir.s

- GNU as is a single pass assembler that can't cope with forward references of labels in `.equ` type statements.

```
--- a/usr/src/lib/libc/common/sys/chdir.s      Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libc/common/sys/chdir.s      Tue Sep 02 11:15:50 2008 -0400
@@ -48,7 +48,5 @@
       SET_SIZE(chdir)

 #ifdef __GNUC__
-# undef chdir
-
-       ANSI_PRAGMA_WEAK2(chdir,_chdir,function)
+       ANSI_PRAGMA_WEAK2(_chdir,chdir,function)
 #endif
```

14.1.135 /usr/src/libc/common/sys/chmod.s

- GNU as is a single pass assembler that can't cope with forward references of labels in `.equ` type statements.

```
--- a/usr/src/lib/libc/common/sys/chmod.s      Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libc/common/sys/chmod.s      Tue Sep 02 11:15:50 2008 -0400
@@ -48,7 +48,5 @@
       SET_SIZE(chmod)

 #ifdef __GNUC__
-# undef chmod
-
```

```
-       ANSI_PRAGMA_WEAK2(chmod,_chmod,function)
+       ANSI_PRAGMA_WEAK2(_chmod,chmod,function)
 #endif
```

14.1.136 /usr/src/libc/common/sys/chown.s

- GNU as is a single pass assembler that can't cope with forward references of labels in `.equ` type statements.

```
--- a/usr/src/lib/libc/common/sys/chown.s       Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libc/common/sys/chown.s       Tue Sep 02 11:15:50 2008 -0400
@@ -48,7 +48,5 @@
       SET_SIZE(chown)

 #ifdef __GNUC__
-# undef chown
-
-       ANSI_PRAGMA_WEAK2(chown,_chown,function)
+       ANSI_PRAGMA_WEAK2(_chown,chown,function)
 #endif
```

14.1.137 /usr/src/libc/common/sys/chroot.s

- GNU as is a single pass assembler that can't cope with forward references of labels in `.equ` type statements.

```
--- a/usr/src/lib/libc/common/sys/chroot.s      Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libc/common/sys/chroot.s      Tue Sep 02 11:15:50 2008 -0400
@@ -48,7 +48,5 @@
       SET_SIZE(chroot)

 #ifdef __GNUC__
-# undef chroot
-
-       ANSI_PRAGMA_WEAK2(chroot,_chroot,function)
+       ANSI_PRAGMA_WEAK2(_chroot,chroot,function)
 #endif
```

14.1.138 /usr/src/libc/common/sys/dup.s

- GNU as is a single pass assembler that can't cope with forward references of labels in `.equ` type statements.

```
--- a/usr/src/lib/libc/common/sys/dup.s Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libc/common/sys/dup.s Tue Sep 02 11:15:50 2008 -0400
@@ -48,7 +48,5 @@
       SET_SIZE(dup)

 #ifdef __GNUC__
-# undef dup
-       ANSI_PRAGMA_WEAK2(dup,_dup,function)
-
+       ANSI_PRAGMA_WEAK2(_dup,dup,function)
 #endif
```

14.1.139 /usr/src/libc/common/sys/fchdir.s

- GNU as is a single pass assembler that can't cope with forward references of labels in `.equ` type statements.

```
--- a/usr/src/lib/libc/common/sys/fchdir.s      Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libc/common/sys/fchdir.s      Tue Sep 02 11:15:50 2008 -0400
@@ -48,7 +48,5 @@
       SET_SIZE(fchdir)

 #ifdef __GNUC__
-# undef fchdir
-
-       ANSI_PRAGMA_WEAK2(fchdir,_fchdir,function)
+       ANSI_PRAGMA_WEAK2(_fchdir,fchdir,function)
 #endif
```

14.1.140 /usr/src/libc/common/sys/fchmod.s

- GNU as is a single pass assembler that can't cope with forward references of labels in `.equ` type statements.

```
--- a/usr/src/lib/libc/common/sys/fchmod.s      Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libc/common/sys/fchmod.s      Tue Sep 02 11:15:50 2008 -0400
@@ -48,7 +48,5 @@
       SET_SIZE(fchmod)

 #ifdef __GNUC__
-# undef fchmod
-
-       ANSI_PRAGMA_WEAK2(fchmod,_fchmod,function)
+       ANSI_PRAGMA_WEAK2(_fchmod,fchmod,function)
 #endif
```

14.1.141 /usr/src/libc/common/sys/fcntl.s

- GNU as is a single pass assembler that can't cope with forward references of labels in `.equ` type statements.

```
--- a/usr/src/lib/libc/common/sys/fcntl.s       Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libc/common/sys/fcntl.s       Tue Sep 02 11:15:50 2008 -0400
@@ -39,6 +39,7 @@
       SET_SIZE(__fcntl_syscall)

 #ifdef __GNUC__
-# undef fcntl
-       ANSI_PRAGMA_WEAK2(fcntl,__fcntl,function)
+       ANSI_PRAGMA_WEAK2(__fcntl,__fcntl_syscall,function)
+       ANSI_PRAGMA_WEAK2(_fcntl,__fcntl_syscall,function)
+       ANSI_PRAGMA_WEAK2(fcntl,__fcntl_syscall,function)
 #endif
```

14.1.142 /usr/src/libc/common/sys/fpathconf.s

- GNU as is a single pass assembler that can't cope with forward references of labels in `.equ` type statements.

```
--- a/usr/src/lib/libc/common/sys/fpathconf.s  Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libc/common/sys/fpathconf.s  Tue Sep 02 11:15:50 2008 -0400
@@ -48,7 +48,5 @@
       SET_SIZE(fpathconf)

 #ifdef __GNUC__
-# undef fpathconf
-
-       ANSI_PRAGMA_WEAK2(fpathconf,_fpathconf,function)
+       ANSI_PRAGMA_WEAK2(_fpathconf,fpathconf,function)
 #endif
```

14.1.143 /usr/src/libc/common/sys/fstat.s

- GNU as is a single pass assembler that can't cope with forward references of labels in `.equ` type statements.

```
--- a/usr/src/lib/libc/common/sys/fstat.s      Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libc/common/sys/fstat.s      Tue Sep 02 11:15:50 2008 -0400
@@ -53,7 +53,7 @@
 # if defined(__GNUC__)
 # undef fstat

-       ANSI_PRAGMA_WEAK2(fstat,_fstat,function)
+       ANSI_PRAGMA_WEAK2(_fstat,fstat,function)
 # endif

 #else
@@ -66,9 +66,7 @@
       SET_SIZE(fstat64)

 # if defined(__GNUC__)
-# undef fstat64
-
-       ANSI_PRAGMA_WEAK2(fstat64,_fstat64,function)
+       ANSI_PRAGMA_WEAK2(_fstat64,fstat64,function)
 # endif

 #endif
```

14.1.144 /usr/src/libc/common/sys/fstatvfs.s

- GNU as is a single pass assembler that can't cope with forward references of labels in `.equ` type statements.

```
--- a/usr/src/lib/libc/common/sys/fstatvfs.s   Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libc/common/sys/fstatvfs.s   Tue Sep 02 11:15:50 2008 -0400
@@ -56,7 +56,7 @@
       #ifdef __GNUC__
       # undef fstatvfs

-             ANSI_PRAGMA_WEAK2(fstatvfs,_fstatvfs,function)
+             ANSI_PRAGMA_WEAK2(_fstatvfs,fstatvfs,function)
       #endif

 #else
@@ -69,9 +69,7 @@
       SET_SIZE(fstatvfs64)

       #ifdef __GNUC__
```

```
-       # undef fstatvfs64
-
-             ANSI_PRAGMA_WEAK2(fstatvfs64,_fstatvfs64,function)
+             ANSI_PRAGMA_WEAK2(_fstatvfs64,fstatvfs64,function)
        #endif

 #endif
```

14.1.145 /usr/src/libc/common/sys/getcpuid.s

- GNU as is a single pass assembler that can't cope with forward references of labels in `.equ` type statements.

```
--- a/usr/src/lib/libc/common/sys/getcpuid.s   Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libc/common/sys/getcpuid.s   Tue Sep 02 11:15:50 2008 -0400
@@ -51,7 +51,5 @@
       SET_SIZE(getcpuid)

 #ifdef __GNUC__
-# undef getcpuid
-
-       ANSI_PRAGMA_WEAK2(getcpuid,_getcpuid,function)
+       ANSI_PRAGMA_WEAK2(_getcpuid,getcpuid,function)
 #endif
```

14.1.146 /usr/src/libc/common/sys/getdtents.s

- GNU as is a single pass assembler that can't cope with forward references of labels in `.equ` type statements.

```
--- a/usr/src/lib/libc/common/sys/getdents.s   Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libc/common/sys/getdents.s   Tue Sep 02 11:15:50 2008 -0400
@@ -54,9 +54,7 @@
       SET_SIZE(getdents)

 #ifdef __GNUC__
-# undef getdents
-
-       ANSI_PRAGMA_WEAK2(getdents,_getdents,function)
+       ANSI_PRAGMA_WEAK2(_getdents,getdents,function)
 #endif

 #else
@@ -69,9 +67,7 @@
       SET_SIZE(getdents64)

 #ifdef __GNUC__
-# undef getdents64
-
-       ANSI_PRAGMA_WEAK2(getdents64,_getdents64,function)
+       ANSI_PRAGMA_WEAK2(_getdents64,getdents64,function)
 #endif

 #endif
```

14.1.147 /usr/src/libc/common/sys/getegid.s

- GNU as is a single pass assembler that can't cope with forward references of labels in `.equ` type statements.

```
--- a/usr/src/lib/libc/common/sys/getegid.s    Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libc/common/sys/getegid.s    Tue Sep 02 11:15:50 2008 -0400
@@ -51,6 +51,5 @@
       SET_SIZE(getegid)

 #ifdef __GNUC__
-#undef getegid
-       ANSI_PRAGMA_WEAK2(getegid,_getegid,function)
+       ANSI_PRAGMA_WEAK2(_getegid,getegid,function)
 #endif
```

14.1.148 /usr/src/libc/common/sys/geteuid.s

- GNU as is a single pass assembler that can't cope with forward references of labels in `.equ` type statements.

```
--- a/usr/src/lib/libc/common/sys/geteuid.s    Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libc/common/sys/geteuid.s    Tue Sep 02 11:15:50 2008 -0400
@@ -54,7 +54,6 @@
       SET_SIZE(geteuid)

 #ifdef __GNUC__
-# undef geteuid
-       ANSI_PRAGMA_WEAK2(geteuid,_geteuid,function)
-       ANSI_PRAGMA_WEAK2(_private_geteuid,_geteuid,function)
+       ANSI_PRAGMA_WEAK2(_geteuid,geteuid,function)
+       ANSI_PRAGMA_WEAK2(_private_geteuid,geteuid,function)
 #endif
```

14.1.149 /usr/src/libc/common/sys/getgroups.s

- GNU as is a single pass assembler that can't cope with forward references of labels in `.equ` type statements.

```
--- a/usr/src/lib/libc/common/sys/getgroups.s  Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libc/common/sys/getgroups.s  Tue Sep 02 11:15:50 2008 -0400
@@ -48,7 +48,5 @@
       SET_SIZE(getgroups)

 #ifdef __GNUC__
-# undef getgroups
-
-       ANSI_PRAGMA_WEAK2(getgroups,_getgroups,function)
+       ANSI_PRAGMA_WEAK2(_getgroups,getgroups,function)
 #endif
```

14.1.150 /usr/src/libc/common/sys/getitimer.s

- GNU as is a single pass assembler that can't cope with forward references of labels in `.equ` type statements.

```
--- a/usr/src/lib/libc/common/sys/getitimer.s  Tue Aug 19 11:19:58 2008 -0400
```

```
+++ b/usr/src/lib/libc/common/sys/getitimer.s  Tue Sep 02 11:15:50 2008 -0400
@@ -48,7 +48,5 @@
       SET_SIZE(getitimer)

 #ifdef __GNUC__
-# undef getitimer
-
-       ANSI_PRAGMA_WEAK2(getitimer,_getitimer,function)
+       ANSI_PRAGMA_WEAK2(_getitimer,getitimer,function)
 #endif
```

14.1.151 /usr/src/libc/common/sys/getppid.s

- GNU as is a single pass assembler that can't cope with forward references of labels in `.equ` type statements.

```
--- a/usr/src/lib/libc/common/sys/getppid.s    Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libc/common/sys/getppid.s    Tue Sep 02 11:15:50 2008 -0400
@@ -51,7 +51,5 @@
       SET_SIZE(getppid)

 #ifdef __GNUC__
-# undef getppid
-
-       ANSI_PRAGMA_WEAK2(getppid,_getppid,function)
+       ANSI_PRAGMA_WEAK2(_getppid,getppid,function)
 #endif
```

14.1.152 /usr/src/libc/common/sys/install_utrap.s

- GNU as is a single pass assembler that can't cope with forward references of labels in `.equ` type statements.

```
--- a/usr/src/lib/libc/common/sys/install_utrap.s     Tue Aug 19 11:19:58 2008 -
0400
+++ b/usr/src/lib/libc/common/sys/install_utrap.s     Tue Sep 02 11:15:50 2008 -
0400
@@ -45,7 +45,5 @@
       SET_SIZE(install_utrap)

 #ifdef __GNUC__
-# undef install_utrap
-
-       ANSI_PRAGMA_WEAK2(install_utrap,_install_utrap,function)
+       ANSI_PRAGMA_WEAK2(_install_utrap,install_utrap,function)
 #endif
```

14.1.153 /usr/src/libc/common/sys/ioctl.s

- GNU as is a single pass assembler that can't cope with forward references of labels in `.equ` type statements.

```
--- a/usr/src/lib/libc/common/sys/ioctl.s      Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libc/common/sys/ioctl.s      Tue Sep 02 11:15:50 2008 -0400
@@ -47,7 +47,5 @@
       SET_SIZE(ioctl)

 #ifdef __GNUC__
```

```
-# undef ioctl
-
-       ANSI_PRAGMA_WEAK2(ioctl,_ioctl,function)
+       ANSI_PRAGMA_WEAK2(_ioctl,ioctl,function)
 #endif
```

14.1.154 /usr/src/libc/common/sys/kill.s

- GNU as is a single pass assembler that can't cope with forward references of labels in `.equ` type statements.

```
--- a/usr/src/lib/libc/common/sys/kill.s        Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libc/common/sys/kill.s        Tue Sep 02 11:15:50 2008 -0400
@@ -48,6 +48,5 @@
        SET_SIZE(kill)

 #ifdef __GNUC__
-# undef kill
-       ANSI_PRAGMA_WEAK2(kill,_kill,function)
+       ANSI_PRAGMA_WEAK2(_kill,kill,function)
 #endif
```

14.1.155 /usr/src/libc/common/sys/lchown.s

- GNU as is a single pass assembler that can't cope with forward references of labels in `.equ` type statements.

```
--- a/usr/src/lib/libc/common/sys/lchown.s      Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libc/common/sys/lchown.s      Tue Sep 02 11:15:50 2008 -0400
@@ -48,7 +48,5 @@
        SET_SIZE(lchown)

 #ifdef __GNUC__
-# undef lchown
-
-       ANSI_PRAGMA_WEAK2(lchown,_lchown,function)
+       ANSI_PRAGMA_WEAK2(_lchown,lchown,function)
 #endif
```

14.1.156 /usr/src/libc/common/sys/llseek.s

- GNU as is a single pass assembler that can't cope with forward references of labels in `.equ` type statements.

```
--- a/usr/src/lib/libc/common/sys/llseek.s      Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libc/common/sys/llseek.s      Tue Sep 02 11:15:50 2008 -0400
@@ -57,7 +57,5 @@
        SET_SIZE(llseek)

 #ifdef __GNUC__
-# undef llseek
-
-       ANSI_PRAGMA_WEAK2(llseek,_llseek,function)
+       ANSI_PRAGMA_WEAK2(_llseek,llseek,function)
 #endif
```

14.1.157 /usr/src/libc/common/sys/lseek.s

- GNU as is a single pass assembler that can't cope with forward references of labels in `.equ` type statements.

```
--- a/usr/src/lib/libc/common/sys/lseek.s      Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libc/common/sys/lseek.s      Tue Sep 02 11:15:50 2008 -0400
@@ -54,9 +54,7 @@
       SET_SIZE(lseek)

 # ifdef __GNUC__
-# undef lseek
-
-      ANSI_PRAGMA_WEAK2(lseek,_lseek,function)
+      ANSI_PRAGMA_WEAK2(_lseek,lseek,function)
 # endif

 #else
@@ -71,9 +69,7 @@
       SET_SIZE(lseek64)

 # ifdef __GNUC__
-# undef lseek64
-
-      ANSI_PRAGMA_WEAK2(lseek64,_lseek64,function)
+      ANSI_PRAGMA_WEAK2(_lseek64,lseek64,function)
 # endif

 #endif
```

14.1.158 /usr/src/libc/common/sys/lstat.s

- GNU as is a single pass assembler that can't cope with forward references of labels in `.equ` type statements.

```
--- a/usr/src/lib/libc/common/sys/lstat.s      Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libc/common/sys/lstat.s      Tue Sep 02 11:15:50 2008 -0400
@@ -54,9 +54,7 @@
       SET_SIZE(lstat)

 # ifdef __GNUC__
-# undef lstat
-
-      ANSI_PRAGMA_WEAK2(lstat,_lstat,function)
+      ANSI_PRAGMA_WEAK2(_lstat,lstat,function)
 # endif

 #else
@@ -69,9 +67,7 @@
       SET_SIZE(lstat64)

 # ifdef __GNUC__
-# undef lstat64
-
-      ANSI_PRAGMA_WEAK2(lstat64,_lstat64,function)
+      ANSI_PRAGMA_WEAK2(_lstat64,lstat64,function)
 # endif

 #endif
```

14.1.159 /usr/src/libc/common/sys/memcntl.s

- GNU as is a single pass assembler that can't cope with forward references of labels in `.equ` type statements.

```
--- a/usr/src/lib/libc/common/sys/memcntl.s     Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libc/common/sys/memcntl.s     Tue Sep 02 11:15:50 2008 -0400
@@ -49,7 +49,5 @@
       SET_SIZE(memcntl)

 #ifdef __GNUC__
-# undef memcntl
-
-       ANSI_PRAGMA_WEAK2(memcntl,_memcntl,function)
+       ANSI_PRAGMA_WEAK2(_memcntl,memcntl,function)
 #endif
```

14.1.160 /usr/src/libc/common/sys/mkdir.s

- GNU as is a single pass assembler that can't cope with forward references of labels in `.equ` type statements.

```
--- a/usr/src/lib/libc/common/sys/mkdir.s       Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libc/common/sys/mkdir.s       Tue Sep 02 11:15:50 2008 -0400
@@ -48,7 +48,5 @@
       SET_SIZE(mkdir)

 #ifdef __GNUC__
-# undef mkdir
-
-       ANSI_PRAGMA_WEAK2(mkdir,_mkdir,function)
+       ANSI_PRAGMA_WEAK2(_mkdir,mkdir,function)
 #endif
```

14.1.161 /usr/src/libc/common/sys/mknod.s

- GNU as is a single pass assembler that can't cope with forward references of labels in `.equ` type statements.

```
--- a/usr/src/lib/libc/common/sys/mknod.s       Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libc/common/sys/mknod.s       Tue Sep 02 11:15:50 2008 -0400
@@ -48,7 +48,5 @@
       SET_SIZE(mknod)

 #ifdef __GNUC__
-# undef mknod
-
-       ANSI_PRAGMA_WEAK2(mknod,_mknod,function)
+       ANSI_PRAGMA_WEAK2(_mknod,mknod,function)
 #endif
```

14.1.162 /usr/src/libc/common/sys/mmap.s

- GNU as is a single pass assembler that can't cope with forward references of labels in `.equ` type statements.

```
--- a/usr/src/lib/libc/common/sys/mmap.s        Tue Aug 19 11:19:58 2008 -0400
```

```
+++ b/usr/src/lib/libc/common/sys/mmap.s       Tue Sep 02 11:15:50 2008 -0400
@@ -64,9 +64,7 @@
       SET_SIZE(mmap)

 # ifdef __GNUC__
-# undef mmap
-
-       ANSI_PRAGMA_WEAK2(mmap,_mmap,function)
+       ANSI_PRAGMA_WEAK2(_mmap,mmap,function)
 # endif

 #else
@@ -89,9 +87,7 @@
       SET_SIZE(mmap64)

 # ifdef __GNUC__
-# undef mmap64
-
-       ANSI_PRAGMA_WEAK2(mmap64,_mmap64,function)
+       ANSI_PRAGMA_WEAK2(_mmap64,mmap64,function)
 # endif

 #endif
```

14.1.163 /usr/src/libc/common/sys/mount.s

- GNU as is a single pass assembler that can't cope with forward references of labels in `.equ` type statements.

```
--- a/usr/src/lib/libc/common/sys/mount.s      Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libc/common/sys/mount.s      Tue Sep 02 11:15:50 2008 -0400
@@ -49,7 +49,5 @@
       SET_SIZE(mount)

 #ifdef __GNUC__
-# undef mount
-
-       ANSI_PRAGMA_WEAK2(mount,_mount,function)
+       ANSI_PRAGMA_WEAK2(_mount,mount,function)
 #endif
```

14.1.164 /usr/src/libc/common/sys/nice.s

- GNU as is a single pass assembler that can't cope with forward references of labels in `.equ` type statements.

```
--- a/usr/src/lib/libc/common/sys/nice.s       Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libc/common/sys/nice.s       Tue Sep 02 11:15:50 2008 -0400
@@ -48,7 +48,5 @@
       SET_SIZE(nice)

 #ifdef __GNUC__
-# undef nice
-
-       ANSI_PRAGMA_WEAK2(nice,_nice,function)
+       ANSI_PRAGMA_WEAK2(_nice,nice,function)
 #endif
```

14.1.165 /usr/src/libc/common/sys/ntp_adjtime.s

- GNU as is a single pass assembler that can't cope with forward references of labels in `.equ` type statements.

```
--- a/usr/src/lib/libc/common/sys/ntp_adjtime.s      Tue Aug 19 11:19:58 2008 -
0400
+++ b/usr/src/lib/libc/common/sys/ntp_adjtime.s      Tue Sep 02 11:15:50 2008 -
0400
@@ -44,7 +44,5 @@
       SET_SIZE(ntp_adjtime)

 #ifdef __GNUC__
-# undef ntp_adjtime
-
-       ANSI_PRAGMA_WEAK2(ntp_adjtime,_ntp_adjtime,function)
+       ANSI_PRAGMA_WEAK2(_ntp_adjtime,ntp_adjtime,function)
 #endif
```

14.1.166 /usr/src/libc/common/sys/ntp_gettime.s

- GNU as is a single pass assembler that can't cope with forward references of labels in `.equ` type statements.

```
--- a/usr/src/lib/libc/common/sys/ntp_gettime.s      Tue Aug 19 11:19:58 2008 -
0400
+++ b/usr/src/lib/libc/common/sys/ntp_gettime.s      Tue Sep 02 11:15:50 2008 -
0400
@@ -44,7 +44,5 @@
       SET_SIZE(ntp_gettime)

 #ifdef __GNUC__
-# undef ntp_gettime
-
-       ANSI_PRAGMA_WEAK2(ntp_gettime,_ntp_gettime,function)
+       ANSI_PRAGMA_WEAK2(_ntp_gettime,ntp_gettime,function)
 #endif
```

14.1.167 /usr/src/libc/common/sys/pathconf.s

- GNU as is a single pass assembler that can't cope with forward references of labels in `.equ` type statements.

```
--- a/usr/src/lib/libc/common/sys/pathconf.s   Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libc/common/sys/pathconf.s   Tue Sep 02 11:15:50 2008 -0400
@@ -48,7 +48,5 @@
       SET_SIZE(pathconf)

 #ifdef __GNUC__
-# undef pathconf
-
-       ANSI_PRAGMA_WEAK2(pathconf,_pathconf,function)
+       ANSI_PRAGMA_WEAK2(_pathconf,pathconf,function)
 #endif
```

14.1.168 /usr/src/libc/common/sys/pcsample.s

- GNU as is a single pass assembler that can't cope with forward references of labels in `.equ` type statements.

```
--- a/usr/src/lib/libc/common/sys/pcsample.s   Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libc/common/sys/pcsample.s   Tue Sep 02 11:15:50 2008 -0400
@@ -44,7 +44,5 @@
       SET_SIZE(pcsample)

 #ifdef __GNUC__
-# undef pcsample
-
-       ANSI_PRAGMA_WEAK2(pcsample,_pcsample,function)
+       ANSI_PRAGMA_WEAK2(_pcsample,pcsample,function)
 #endif
```

14.1.169 /usr/src/libc/common/sys/readlink.s

- GNU as is a single pass assembler that can't cope with forward references of labels in `.equ` type statements.

```
--- a/usr/src/lib/libc/common/sys/readlink.s   Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libc/common/sys/readlink.s   Tue Sep 02 11:15:50 2008 -0400
@@ -50,7 +50,5 @@
       SET_SIZE(readlink)

 #ifdef __GNUC__
-# undef readlink
-
-       ANSI_PRAGMA_WEAK2(readlink,_readlink,function)
+       ANSI_PRAGMA_WEAK2(_readlink,readlink,function)
 #endif
```

14.1.170 /usr/src/libc/common/sys/resolvepath.s

- GNU as is a single pass assembler that can't cope with forward references of labels in `.equ` type statements.

```
--- a/usr/src/lib/libc/common/sys/resolvepath.s      Tue Aug 19 11:19:58 2008 -
0400
+++ b/usr/src/lib/libc/common/sys/resolvepath.s      Tue Sep 02 11:15:50 2008 -
0400
@@ -44,7 +44,5 @@
       SET_SIZE(resolvepath)

 #ifdef __GNUC__
-# undef resolvepath
-
-       ANSI_PRAGMA_WEAK2(resolvepath,_resolvepath,function)
+       ANSI_PRAGMA_WEAK2(_resolvepath,resolvepath,function)
 #endif
```

14.1.171 /usr/src/libc/common/sys/rmdir.s

- GNU as is a single pass assembler that can't cope with forward references of labels in `.equ` type statements.

```
--- a/usr/src/lib/libc/common/sys/rmdir.s      Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libc/common/sys/rmdir.s      Tue Sep 02 11:15:50 2008 -0400
@@ -48,7 +48,5 @@
       SET_SIZE(rmdir)

 #ifdef __GNUC__
-# undef rmdir
-
-       ANSI_PRAGMA_WEAK2(rmdir,_rmdir,function)
+       ANSI_PRAGMA_WEAK2(_rmdir,rmdir,function)
 #endif
```

14.1.172 /usr/src/libc/common/sys/setgroups.s

- GNU as is a single pass assembler that can't cope with forward references of labels in `.equ` type statements.

```
--- a/usr/src/lib/libc/common/sys/setgroups.s  Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libc/common/sys/setgroups.s  Tue Sep 02 11:15:50 2008 -0400
@@ -48,7 +48,5 @@
       SET_SIZE(setgroups)

 #ifdef __GNUC__
-# undef setgroups
-
-       ANSI_PRAGMA_WEAK2(setgroups,_setgroups,function)
+       ANSI_PRAGMA_WEAK2(_setgroups,setgroups,function)
 #endif
```

14.1.173 /usr/src/libc/common/sys/setitimer.s

- GNU as is a single pass assembler that can't cope with forward references of labels in `.equ` type statements.

```
--- a/usr/src/lib/libc/common/sys/setitimer.s  Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libc/common/sys/setitimer.s  Tue Sep 02 11:15:50 2008 -0400
@@ -48,7 +48,5 @@
       SET_SIZE(setitimer)

 #ifdef __GNUC__
-# undef setitimer
-
-       ANSI_PRAGMA_WEAK2(setitimer,_setitimer,function)
+       ANSI_PRAGMA_WEAK2(_setitimer,setitimer,function)
 #endif
```

14.1.174 /usr/src/libc/common/sys/setreid.s

- GNU as is a single pass assembler that can't cope with forward references of labels in `.equ` type statements.

```
--- a/usr/src/lib/libc/common/sys/setreid.s    Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libc/common/sys/setreid.s    Tue Sep 02 11:15:50 2008 -0400
@@ -46,9 +46,6 @@
       SET_SIZE(setregid)

 #ifdef __GNUC__
-# undef setreuid
```

```
-# undef setregid
-
-       ANSI_PRAGMA_WEAK2(setreuid,_setreuid,function)
-       ANSI_PRAGMA_WEAK2(setregid,_setregid,function)
+       ANSI_PRAGMA_WEAK2(_setreuid,setreuid,function)
+       ANSI_PRAGMA_WEAK2(_setregid,setregid,function)
 #endif
```

14.1.175 /usr/src/libc/common/sys/setrlimit.s

- GNU as is a single pass assembler that can't cope with forward references of labels in `.equ` type statements.

```
--- a/usr/src/lib/libc/common/sys/setrlimit.s  Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libc/common/sys/setrlimit.s  Tue Sep 02 11:15:50 2008 -0400
@@ -56,7 +56,7 @@
        # ifdef __GNUC__
        # undef setrlimit

-              ANSI_PRAGMA_WEAK2(setrlimit,_setrlimit,function)
+              ANSI_PRAGMA_WEAK2(_setrlimit,setrlimit,function)
        # endif

 #else
@@ -66,9 +66,7 @@
        SET_SIZE(setrlimit64)

        # ifdef __GNUC__
-       # undef setrlimit64
-
-              ANSI_PRAGMA_WEAK2(setrlimit64,_setrlimit64,function)
+              ANSI_PRAGMA_WEAK2(_setrlimit64,setrlimit64,function)
        # endif

 #endif
```

14.1.176 /usr/src/libc/common/sys/sigaltstk.s

- GNU as is a single pass assembler that can't cope with forward references of labels in `.equ` type statements.

```
--- a/usr/src/lib/libc/common/sys/sigaltstk.s  Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libc/common/sys/sigaltstk.s  Tue Sep 02 11:15:50 2008 -0400
@@ -48,6 +48,5 @@
       SET_SIZE(sigaltstack)

 #ifdef __GNUC__
-# undef sigaltstack
-       ANSI_PRAGMA_WEAK2(sigaltstack,_sigaltstack,function)
+       ANSI_PRAGMA_WEAK2(_sigaltstack,sigaltstack,function)
 #endif
```

14.1.177 /usr/src/libc/common/sys/sigsendset.s

- GNU as is a single pass assembler that can't cope with forward references of labels in `.equ` type statements.

```
--- a/usr/src/lib/libc/common/sys/sigsendset.s Tue Aug 19 11:19:58 2008 -0400
```

```
+++ b/usr/src/lib/libc/common/sys/sigsendset.s Tue Sep 02 11:15:50 2008 -0400
@@ -48,7 +48,5 @@
       SET_SIZE(sigsendset)

 #ifdef __GNUC__
-# undef sigsendset
-
-       ANSI_PRAGMA_WEAK2(sigsendset,_sigsendset,function)
+       ANSI_PRAGMA_WEAK2(_sigsendset,sigsendset,function)
 #endif
```

14.1.178 /usr/src/libc/common/sys/stat.s

- GNU as is a single pass assembler that can't cope with forward references of labels in `.equ` type statements.

```
--- a/usr/src/lib/libc/common/sys/stat.s       Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libc/common/sys/stat.s       Tue Sep 02 11:15:50 2008 -0400
@@ -55,7 +55,7 @@
       # ifdef __GNUC__
       # undef stat

-             ANSI_PRAGMA_WEAK2(stat,_stat,function)
+             ANSI_PRAGMA_WEAK2(_stat,stat,function)
       # endif

 #else
@@ -68,9 +68,7 @@
       SET_SIZE(stat64)

       # ifdef __GNUC__
-      # undef stat64
-
-             ANSI_PRAGMA_WEAK2(stat64,_stat64,function)
+             ANSI_PRAGMA_WEAK2(_stat64,stat64,function)
       # endif

 #endif
```

14.1.179 /usr/src/libc/common/sys/statvfs.s

- GNU as is a single pass assembler that can't cope with forward references of labels in `.equ` type statements.

```
--- a/usr/src/lib/libc/common/sys/statvfs.s    Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libc/common/sys/statvfs.s    Tue Sep 02 11:15:50 2008 -0400
@@ -54,9 +54,7 @@
       SET_SIZE(statvfs)

 # ifdef __GNUC__
-#  undef statvfs
-
-       ANSI_PRAGMA_WEAK2(statvfs,_statvfs,function)
+       ANSI_PRAGMA_WEAK2(_statvfs,statvfs,function)
 # endif

 #else
@@ -69,9 +67,7 @@
       SET_SIZE(statvfs64)
```

```
 # ifdef __GNUC__
-#  undef statvfs64
-
-      ANSI_PRAGMA_WEAK2(statvfs64,_statvfs64,function)
+      ANSI_PRAGMA_WEAK2(_statvfs64,statvfs64,function)
 # endif

 #endif
```

14.1.180 /usr/src/libc/common/sys/symlink.s

- GNU as is a single pass assembler that can't cope with forward references of labels in `.equ` type statements.

```
--- a/usr/src/lib/libc/common/sys/symlink.s    Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libc/common/sys/symlink.s    Tue Sep 02 11:15:51 2008 -0400
@@ -48,7 +48,5 @@
       SET_SIZE(symlink)

 #ifdef __GNUC__
-# undef symlink
-
-      ANSI_PRAGMA_WEAK2(symlink,_symlink,function)
+      ANSI_PRAGMA_WEAK2(_symlink,symlink,function)
 #endif
```

14.1.181 /usr/src/libc/common/sys/sync.s

- GNU as is a single pass assembler that can't cope with forward references of labels in `.equ` type statements.

```
--- a/usr/src/lib/libc/common/sys/sync.s       Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libc/common/sys/sync.s       Tue Sep 02 11:15:51 2008 -0400
@@ -48,7 +48,5 @@
       SET_SIZE(sync)

 #ifdef __GNUC__
-# undef sync
-
-      ANSI_PRAGMA_WEAK2(sync,_sync,function)
+      ANSI_PRAGMA_WEAK2(_sync,sync,function)
 #endif
```

14.1.182 /usr/src/libc/common/sys/sysfs.s

- GNU as is a single pass assembler that can't cope with forward references of labels in `.equ` type statements.

```
--- a/usr/src/lib/libc/common/sys/sysfs.s      Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libc/common/sys/sysfs.s      Tue Sep 02 11:15:51 2008 -0400
@@ -50,7 +50,5 @@
       SET_SIZE(sysfs)

 #ifdef __GNUC__
-# undef sysfs
-
-      ANSI_PRAGMA_WEAK2(sysfs,_sysfs,function)
+      ANSI_PRAGMA_WEAK2(_sysfs,sysfs,function)
```

```
 #endif
```

14.1.183 /usr/src/libc/common/sys/sysinfo.s

- GNU as is a single pass assembler that can't cope with forward references of labels in `.equ` type statements.

```
--- a/usr/src/lib/libc/common/sys/sysinfo.s     Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libc/common/sys/sysinfo.s     Tue Sep 02 11:15:51 2008 -0400
@@ -48,7 +48,5 @@
       SET_SIZE(sysinfo)

 #ifdef __GNUC__
-# undef sysinfo
-
-       ANSI_PRAGMA_WEAK2(sysinfo,_sysinfo,function)
+       ANSI_PRAGMA_WEAK2(_sysinfo,sysinfo,function)
 #endif
```

14.1.184 /usr/src/libc/common/sys/times.s

- GNU as is a single pass assembler that can't cope with forward references of labels in `.equ` type statements.

```
--- a/usr/src/lib/libc/common/sys/times.s       Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libc/common/sys/times.s       Tue Sep 02 11:15:51 2008 -0400
@@ -48,7 +48,5 @@
       SET_SIZE(times)

 #ifdef __GNUC__
-# undef times
-
-       ANSI_PRAGMA_WEAK2(times,_times,function)
+       ANSI_PRAGMA_WEAK2(_times,times,function)
 #endif
```

14.1.185 /usr/src/libc/common/sys/ulimit.s

- GNU as is a single pass assembler that can't cope with forward references of labels in `.equ` type statements.

```
--- a/usr/src/lib/libc/common/sys/ulimit.s      Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libc/common/sys/ulimit.s      Tue Sep 02 11:15:51 2008 -0400
@@ -48,7 +48,5 @@
       SET_SIZE(ulimit)

 #ifdef __GNUC__
-# undef ulimit
-
-       ANSI_PRAGMA_WEAK2(ulimit,_ulimit,function)
+       ANSI_PRAGMA_WEAK2(_ulimit,ulimit,function)
 #endif
```

14.1.186 /usr/src/libc/common/sys/umask.s

- GNU as is a single pass assembler that can't cope with forward references of labels in `.equ` type statements.

```
--- a/usr/src/lib/libc/common/sys/umask.s      Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libc/common/sys/umask.s      Tue Sep 02 11:15:51 2008 -0400
@@ -48,7 +48,5 @@
       SET_SIZE(umask)

 #ifdef __GNUC__
-# undef umask
-
-       ANSI_PRAGMA_WEAK2(umask,_umask,function)
+       ANSI_PRAGMA_WEAK2(_umask,umask,function)
 #endif
```

14.1.187 /usr/src/libc/common/sys/umount2.s

- GNU as is a single pass assembler that can't cope with forward references of labels in `.equ` type statements.

```
--- a/usr/src/lib/libc/common/sys/umount2.s    Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libc/common/sys/umount2.s    Tue Sep 02 11:15:51 2008 -0400
@@ -44,7 +44,5 @@
       SET_SIZE(umount2)

 #ifdef __GNUC__
-# undef umount2
-
-       ANSI_PRAGMA_WEAK2(umount2,_umount2,function)
+       ANSI_PRAGMA_WEAK2(_umount2,umount2,function)
 #endif
```

14.1.188 /usr/src/libc/common/sys/unlink.s

- GNU as is a single pass assembler that can't cope with forward references of labels in `.equ` type statements.

```
--- a/usr/src/lib/libc/common/sys/unlink.s     Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libc/common/sys/unlink.s     Tue Sep 02 11:15:51 2008 -0400
@@ -48,7 +48,5 @@
       SET_SIZE(unlink)

 #ifdef __GNUC__
-# undef unlink
-
-       ANSI_PRAGMA_WEAK2(unlink,_unlink,function)
+       ANSI_PRAGMA_WEAK2(_unlink,unlink,function)
 #endif
```

14.1.189 /usr/src/libc/common/sys/utime.s

- GNU as is a single pass assembler that can't cope with forward references of labels in `.equ` type statements.

```
--- a/usr/src/lib/libc/common/sys/utime.s      Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libc/common/sys/utime.s      Tue Sep 02 11:15:51 2008 -0400
@@ -48,7 +48,5 @@
       SET_SIZE(utime)

 #ifdef __GNUC__
-# undef utime
```

```
-
-       ANSI_PRAGMA_WEAK2(utime,_utime,function)
+       ANSI_PRAGMA_WEAK2(_utime,utime,function)
 #endif
```

14.1.190 /usr/src/lib/libc/inc/thr_uberdata.h

- Add System z as a supported platform.

```
--- a/usr/src/lib/libc/inc/thr_uberdata.h       Fri Aug 01 19:14:04 2008 -0700
+++ b/usr/src/lib/libc/inc/thr_uberdata.h       Wed Aug 06 14:18:46 2008 -0400
@@ -136,8 +136,15 @@
 #define        WAITERMASK64    0x00ff000000000000ULL
 #define        SPINNERMASK64   0x0000ff0000000000ULL

+#elif defined(__s390)
+/* lock.lock64.pad[x]     4 5 6 7 */
+#define        LOCKMASK        0xff000000
+#define        WAITERMASK      0x000000ff
+#define        WAITER          0x00000001
+#define        LOCKSET         0xff
+#define        LOCKCLEAR       0
 #else
-#error "neither __sparc nor __x86 is defined"
+#error "neither __sparc, __x86, or __s390 is defined"
 #endif

 /*
@@ -646,7 +653,11 @@ typedef struct ulwp {
        sigset_t        ul_tmpmask;     /* signal mask for sigsuspend/pollsys
*/
        siginfo_t       ul_siginfo;     /* deferred siginfo */
        mutex_t         ul_spinlock;    /* used when suspending/continuing */
+#ifndef __s390
        fpuenv_t        ul_fpuenv;      /* floating point state */
+#else
+       uint32_t        ul_fpc;         /* Floating point control register */
+#endif
        uintptr_t       ul_sp;          /* stack pointer when blocked */
        void            *ul_ex_unwind;  /* address of _ex_unwind() or -1 */
 #if defined(sparc)

--- a/usr/src/lib/libc/inc/thr_uberdata.h      Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libc/inc/thr_uberdata.h      Tue Sep 02 11:15:51 2008 -0400
@@ -140,9 +140,18 @@
 /* lock.lock64.pad[x]        4 5 6 7 */
 #define     LOCKMASK     0xff000000
 #define     WAITERMASK   0x000000ff
+#define     SPINNERMASK  0x00ff0000
+#define     SPINNERSHIFT 16
 #define     WAITER       0x00000001
 #define     LOCKSET             0xff
 #define     LOCKCLEAR    0
+
+#define     PIDSHIFT     32
+#define     LOCKMASK64   0xffffffffff000000ULL
+#define     LOCKBYTE64   0x00000000ff000000ULL
+#define     WAITERMASK64 0x00000000000000ffULL
+#define     SPINNERMASK64 0x0000000000ff0000ULL
+
 #else
 #error "neither __sparc, __x86, or __s390 is defined"
```

```
 #endif
@@ -552,7 +561,7 @@
       lwpid_t             ul_lwpid;     /* thread id, aka the lwp id */
       pri_t         ul_pri;             /* scheduling priority */
       pri_t         ul_epri;      /* real-time ceiling priority */
-      char          ul_policy;    /* scheduling policy */
+      int8_t        ul_policy;    /* scheduling policy */
       char          ul_cid;             /* scheduling class id */
       union {
             struct {
@@ -660,7 +669,7 @@
 #endif
       uintptr_t     ul_sp;        /* stack pointer when blocked */
       void          *ul_ex_unwind;      /* address of _ex_unwind() or -1 */
-#if defined(sparc)
+#if defined(sparc) || defined(__s390)
       void          *ul_unwind_ret;     /* used only by _ex_clnup_handler() */
 #endif
 } ulwp_t;
@@ -1061,7 +1070,9 @@
       sigset32_t    ul_tmpmask;   /* signal mask for sigsuspend/pollsys */
       siginfo32_t   ul_siginfo;   /* deferred siginfo */
       mutex_t             ul_spinlock; /* used when suspending/continuing */
+#ifndef __s390
       fpuenv32_t    ul_fpuenv;    /* floating point state */
+#endif
       caddr32_t     ul_sp;        /* stack pointer when blocked */
 #if defined(sparc)
       caddr32_t     ul_unwind_ret;      /* used only by _ex_clnup_handler() */
```

14.1.191 /usr/src/lib/libc/port/fp/floatdidf.c

- Use 1L as the argument to shift so that compiler doesn't complain about being wider than data type.

```
--- a/usr/src/lib/libc/port/fp/floatdidf.c      Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libc/port/fp/floatdidf.c      Tue Sep 02 11:15:51 2008 -0400
@@ -41,8 +41,8 @@

       aa.q = a;
       d = aa.sl[H];
-      d *= (1 << HALF_BITS);
-      d *= (1 << HALF_BITS);
+      d *= (1L << HALF_BITS);
+      d *= (1L << HALF_BITS);
       d += aa.ul[L];

       return (d);
```

14.1.192 /usr/src/lib/libc/port/fp/floatdisf.c

- Use 1L as the argument to shift so that compiler doesn't complain about being wider than data type.

```
--- a/usr/src/lib/libc/port/fp/floatdisf.c      Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libc/port/fp/floatdisf.c      Tue Sep 02 11:15:51 2008 -0400
@@ -41,8 +41,8 @@

       aa.q = a;
       d = aa.sl[H];
```

```
-       d *= (1 << HALF_BITS);
-       d *= (1 << HALF_BITS);
+       d *= (1L << HALF_BITS);
+       d *= (1L << HALF_BITS);
        d += aa.ul[L];

        return ((float)d);
```

14.1.193 /usr/src/lib/libc/port/fp/gconvert.c

- Add System z as a supported platform.

```
--- a/usr/src/lib/libc/port/fp/gconvert.c       Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libc/port/fp/gconvert.c       Tue Sep 02 11:15:51 2008 -0400
@@ -158,7 +158,7 @@
        decimal_record  dr;
        fp_exception_field_type fef;

-#if defined(__sparc) || defined(__sparc)
+#if defined(__sparc) || defined(__s390)
        dm.rd = _QgetRD();
 #elif defined(__i386) || defined(__amd64)
        dm.rd = __xgetRD();
```

14.1.194 /usr/src/lib/libc/port/fp/qdivrem.c

- Use 1L as the argument to shift so that compiler doesn't complain about being wider than data type.

```
--- a/usr/src/lib/libc/port/fp/qdivrem.c        Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libc/port/fp/qdivrem.c        Tue Sep 02 11:15:51 2008 -0400
@@ -44,7 +44,7 @@

 #include "quadint.h"

-#define      B       (1 << HALF_BITS)     /* digit base */
+#define      B       (1L << HALF_BITS)    /* digit base */

 /* Combine two `digits' to make a single two-digit number. */
 #define      COMBINE(a, b) (((ulong_t)(a) << HALF_BITS) | (b))
```

14.1.195 /usr/src/lib/libc/port/fp/quadint.h

- Use 1L as the argument to shift so that compiler doesn't complain about being wider than data type.

```
--- a/usr/src/lib/libc/port/fp/quadint.h        Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libc/port/fp/quadint.h        Tue Sep 02 11:15:51 2008 -0400
@@ -114,7 +114,7 @@
  * (sizeof(long)*CHAR_BIT/2).
  */
 #define      HHALF(x)      ((x) >> HALF_BITS)
-#define      LHALF(x)      ((x) & ((1 << HALF_BITS) - 1))
+#define      LHALF(x)      ((x) & ((1L << HALF_BITS) - 1))
 #define      LHUP(x)              ((x) << HALF_BITS)

 longlong_t   ___divdi3(longlong_t a, longlong_t b);
```

14.1.196 /usr/src/lib/libc/port/gen/gettxt.c

- `cur_cat` is defined as `char []` which means it will never be `NULL` so test is redundant and gcc will issue warning.

```
--- a/usr/src/lib/libc/port/gen/gettxt.c        Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libc/port/gen/gettxt.c        Tue Sep 02 11:15:51 2008 -0400
@@ -106,7 +106,7 @@
                msgfile[name_len] = '\0';
        } else {
                lrw_rdlock(&_rw_cur_cat);
-               if (cur_cat == NULL || *cur_cat == '\0') {
+               if (*cur_cat == '\0') {
                        lrw_unlock(&_rw_cur_cat);
                        return (handle_return(dflt_str));
                }
```

14.1.197 /usr/src/lib/libc/port/gen/gtxt.c

- `cur_cat` is defined as `char []` which means it will never be `NULL` so test is redundant and gcc will issue warning.

```
--- a/usr/src/lib/libc/port/gen/gtxt.c  Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libc/port/gen/gtxt.c  Tue Sep 02 11:15:51 2008 -0400
@@ -273,7 +273,7 @@
         */
        if (!catname || !*catname) {
                lrw_rdlock(&_rw_cur_cat);
-               if (cur_cat == NULL || !*cur_cat) {
+               if (!*cur_cat) {
                        lrw_unlock(&_rw_cur_cat);
                        return (not_found);
                }
```

14.1.198 /usr/src/lib/libc/port/gen/nss_dbdefs.c

- Mask to `hval &` operation is one order of magnitude too large.

```
--- a/usr/src/lib/libc/port/gen/nss_dbdefs.c    Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libc/port/gen/nss_dbdefs.c    Tue Sep 02 11:15:51 2008 -0400
@@ -310,7 +310,7 @@
                        hval = 0;
                        while (*cp) {
                                hval = (hval << 4) + *cp++;
-                               if ((g = (hval & 0xf00000000)) != 0)
+                               if ((g = (hval & 0xf0000000)) != 0)
                                        hval ^= g >> 24;
                                hval &= ~g;
                        }
@@ -333,7 +333,7 @@
        hval = 0;
        while (*cp) {
                hval = (hval << 4) + *cp++;
-               if ((g = (hval & 0xf00000000)) != 0)
+               if ((g = (hval & 0xf0000000)) != 0)
                        hval ^= g >> 24;
                hval &= ~g;
        }
```

14.1.199 /usr/src/lib/libc/port/gen/walkstack.c

- Add System z as a supported platform.

```
--- a/usr/src/lib/libc/port/gen/walkstack.c    Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libc/port/gen/walkstack.c    Tue Sep 02 11:15:51 2008 -0400
@@ -149,7 +149,8 @@
 #define      CHECK_FOR_SIGFRAME(fp, oldctx) ((((fp) + sizeof (struct frame)) +
\
       3 * sizeof (int) == (oldctx)) && \
       (((struct frame *)fp)->fr_savpc == (greg_t)-1))
-#elif defined(__s390x) || defined(__s390)
+
+#elif defined(__s390)
 #define      FRAME_PTR_REGISTER R_SP
 #define      CHECK_FOR_SIGFRAME(fp, oldctx) ((fp) + SA(sizeof (struct frame)) \
       == (oldctx))
@@ -289,6 +290,10 @@
                     */
                    sig = signo; /* already read - see below */
 #endif
+
+#ifdef __s390
+                   sig = 0;      /* S390 FIXME */
+#endif
                    /*
                     * this is the special signal frame, so cons up
                     * the saved fp & pc to pass to user's function
@@ -298,7 +303,11 @@
                        ((uintptr_t)oldctx->
                        uc_mcontext.gregs[FRAME_PTR_REGISTER] +
                        STACK_BIAS);
+#ifndef __s390
                    savepc = oldctx->uc_mcontext.gregs[PC_REGISTER];
+#else
+                   savepc = oldctx->uc_mcontext.psw.pc;
+#endif

                    oldctx = oldctx->uc_link; /* handle nested signals */
               }
```

14.1.200 /usr/src/lib/libc/port/stdio/getpass.c

- Move static function prototypes from within routine to global area: gcc doesn't cope with prototypes declared this way.

```
--- a/usr/src/lib/libc/port/stdio/getpass.c    Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libc/port/stdio/getpass.c    Tue Sep 02 11:15:51 2008 -0400
@@ -49,6 +49,7 @@

 static int intrupt;
 static char *__getpass(const char *, int);
+static void catch(int);

 #define      MAXPASSWD     256     /* max significant characters in password */
 #define      SMLPASSWD     8       /* unix standard  characters in password */
@@ -76,7 +77,6 @@
       FILE   *fi;
       char *pbuf = tsdalloc(_T_GETPASS, MAXPASSWD + 1, NULL);
```

```
        struct sigaction act, osigint, osigtstp;
-       static void catch(int);

        if (pbuf == NULL ||
            (fi = fopen("/dev/tty", "r+F")) == NULL)
```

14.1.201 /usr/src/lib/libcpc/common/libcpc.h

- Add System z support.

```
--- a/usr/src/lib/libcpc/common/libcpc.h        Tue Sep 02 11:22:56 2008 -0400
+++ b/usr/src/lib/libcpc/common/libcpc.h        Thu Sep 25 10:26:10 2008 -0400
@@ -162,7 +162,7 @@
 extern int cpc_enable(cpc_t *cpc);
 extern int cpc_disable(cpc_t *cpc);

-#if defined(__sparc) || defined(__i386)
+#if defined(__sparc) || defined(__i386) || defined(__s390)

 /*
  * Obsolete libcpc interfaces.
```

14.1.202 /usr/src/lib/libcpc/common/libcpc_impl.h

- Add System z support.

```
--- a/usr/src/lib/libcpc/common/libcpc_impl.h   Tue Sep 02 11:22:56 2008 -0400
+++ b/usr/src/lib/libcpc/common/libcpc_impl.h   Thu Sep 25 10:26:10 2008 -0400
@@ -148,7 +148,7 @@
 #define        CPUDRV                          "/devices/pseudo/cpc@0"
 #define        CPUDRV_SHARED                   CPUDRV":shared"

-#if defined(__sparc) || defined(__i386)
+#if defined(__sparc) || defined(__i386) || defined(__s390)
 /*
  * These two are only used for backwards compatibility to the Obsolete CPCv1.
  */
@@ -173,7 +173,10 @@
 #define        CPC_SPARC64_III         3000
 #define        CPC_SPARC64_V           3002

-#endif /* __sparc || __i386 */
+#define CPC_SYSTEM_Z9          4000
+#define CPC_SYSTEM_Z10         4000
+
+#endif /* __sparc || __i386 || __s390 */

 #if defined(__i386) || defined(__amd64)
 /*
@@ -285,7 +288,31 @@
 #define        CPC_P6_PES_PIC0_MASK    (0xffu)
 #define        CPC_P6_PES_PIC1_MASK    (0xffu)

-#endif /* __i386 */
+#elif defined(__s390)
+
+/*
+ * System z processors
+ *
+ * The performance counters on these processors allow up to two 32-bit
```

```
+ * performance events to be captured simultaneously from a selection
+ * of metrics.   The metrics are selected by writing to the performance
+ * control register, and subsequent values collected by reading from the
+ * performance instrumentation counter registers.  Both registers are
+ * priviliged by default, and implemented as ASRs.
+ */
+
+struct _cpc_event {
+       int ce_cpuver;
+       hrtime_t ce_hrt;        /* gethrtime() */
+       uint64_t ce_tick;       /* virtualized %tick */
+       uint64_t ce_pic[2];     /* virtualized %pic */
+       uint64_t ce_pcr;        /* %pcr */
+};
+
+#define        CPC_TICKREG(ev)         ((ev)->ce_tick)
+#define        CPC_TICKREG_NAME        "stck"
+
+#endif /* __s390 */

 #ifdef __cplusplus
 }
```

14.1.203 /usr/src/lib/libcpc/common/obsoleted.c

- Add System z support.

```
--- a/usr/src/lib/libcpc/common/obsoleted.c     Tue Sep 02 11:22:56 2008 -0400
+++ b/usr/src/lib/libcpc/common/obsoleted.c     Thu Sep 25 10:26:10 2008 -0400
@@ -101,6 +101,8 @@
        __cpc_v1_cpuver = this->ce_cpuver;
 #ifdef __sparc
        __cpc_v1_pcr = this->ce_pcr;
+#elif __s390
+       ;
 #else
        __cpc_v1_pes[0] = this->ce_pes[0];
        __cpc_v1_pes[1] = this->ce_pes[1];
@@ -133,6 +135,8 @@
        this->ce_cpuver = __cpc_v1_cpuver;
 #ifdef __sparc
        this->ce_pcr = __cpc_v1_pcr;
+#elif __s390
+       ;
 #else
        this->ce_pes[0] = __cpc_v1_pes[0];
        this->ce_pes[1] = __cpc_v1_pes[1];
```

14.1.204 /usr/src/lib/libcurses/screen/tgetch.c

- Move static function prototypes from within routine to global area: gcc doesn't cope with prototypes declared this way.

```
--- a/usr/src/lib/libcurses/screen/tgetch.c    Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libcurses/screen/tgetch.c    Tue Sep 02 11:15:52 2008 -0400
@@ -59,6 +59,8 @@
 static      int _getkey(int, chtype *);
 static      int _fpk(void);
 static      int _pk(void);
+static      int get_xterm_mouse(int, int *);
```

```
+static       void _map_button(chtype *);

 chtype
 tgetch(int interpret)
@@ -266,7 +268,6 @@
                     if (kp[key]->_keyval == KEY_MOUSE) {
                            MOUSE_STATUS old_mouse;
                            int rc;
-                           static int get_xterm_mouse(int, int *);

                            old_mouse = Mouse_status;

@@ -339,7 +340,6 @@
                          (MOUSE_Y_POS == LINES) &&
                          (SP->slk != (SLK_MAP *) NULL) &&
                          (SP->_map_mbe_to_key  != 0)) {
-                           static void _map_button(chtype *);
                            _map_button(inp);
                     }
```

14.1.205 /usr/src/lib/libdll/Makefile.com

- Need to search `$(ROOT)/usr/include` and not `/usr/include`.

```
--- a/usr/src/lib/libdll/Makefile.com   Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libdll/Makefile.com   Tue Sep 02 11:15:52 2008 -0400
@@ -67,6 +67,7 @@
       $(DLLPLATFORMCPPFLAGS) \
       -I. \
       -I$(ROOT)/usr/include/ast \
+      -I$(ROOT)/usr/include      \
       '-DCONF_LIBSUFFIX=".so"' \
       '-DCONF_LIBPREFIX="lib"' \
       -D_BLD_dll \
```

14.1.206 /usr/src/lib/libidmap/common/namemaps.c

- Need to test value just obtained by `malloc()` not the array address (which will never be `NULL` because it's declared as `char []`).

```
--- a/usr/src/lib/libidmap/common/namemaps.c   Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libidmap/common/namemaps.c   Tue Sep 02 11:15:52 2008 -0400
@@ -784,7 +784,7 @@


       attrs[0] = (ns_ldap_attr_t *)malloc(sizeof (ns_ldap_attr_t));
-      if (attrs == NULL)
+      if (attrs[0] == NULL)
              return (IDMAP_ERR_MEMORY);

       attrs[0]->attrname = attr;
```

14.1.207 /usr/src/lib/libiscsitgt/Makefile.com

- When cross-building `libxml2` may not be installed on the base system and may be placed in `$ROOT`

```
--- a/usr/src/lib/libiscsitgt/Makefile.com     Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libiscsitgt/Makefile.com     Tue Sep 02 11:15:52 2008 -0400
@@ -42,7 +42,7 @@
 $(LINTLIB) :=      SRCS = $(SRCDIR)/$(LINTSRC)

 CFLAGS +=    $(CCVERBOSE)
-CPPFLAGS +=  -I/usr/include/libxml2 -I$(SRCDIR) \
+CPPFLAGS +=  -I$(ROOT)/usr/include/libxml2 -I/usr/include/libxml2 -I$(SRCDIR) \
     -I../../../cmd/iscsi/iscsitgtd

 .KEEP_STATE:
```

14.1.208 /usr/src/lib/libkmf/ber_der/Makefile.com

- When cross-building `libxml2` may not be installed on the base system and may be placed in `$ROOT`

```
--- a/usr/src/lib/libkmf/ber_der/Makefile.com  Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libkmf/ber_der/Makefile.com  Tue Sep 02 11:15:52 2008 -0400
@@ -44,7 +44,9 @@

 CFLAGS              +=     $(CCVERBOSE) -xCC
 CFLAGS64     +=     $(CCVERBOSE) -xCC
-CPPFLAGS     +=     -I/usr/include/libxml2 -I$(INCDIR) -I../../include
+CPPFLAGS     +=     -I$(ROOT)/usr/include/libxml2 \
+                    -I/usr/include/libxml2 \
+                    -I$(INCDIR) -I../../include
 CPPFLAGS64   +=     -I/usr/include/libxml2 -I$(INCDIR) -I../../include
 LDLIBS              +=     -lc
```

14.1.209 /usr/src/lib/libkmf/libkmf/Makefile.com

- When cross-building `libxml2` may not be installed on the base system and may be placed in `$ROOT`

```
--- a/usr/src/lib/libkmf/libkmf/Makefile.com   Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libkmf/libkmf/Makefile.com   Tue Sep 02 11:15:52 2008 -0400
@@ -65,7 +65,8 @@
 $(DYNLIB) :=    LDLIBS += -lxml2
 $(DYNLIB64) :=  LDLIBS64 += -lxml2

-CPPFLAGS     +=     -I$(INCDIR) -I/usr/include/libxml2 -I../../ber_der/inc -
I$(SRCDIR)
+CPPFLAGS     +=     -I$(INCDIR) -I$(ROOT)/usr/include/libxml2 -
I/usr/include/libxml2 \
+                    -I../../ber_der/inc -I$(SRCDIR)

 .KEEP_STATE:
```

14.1.210 /usr/src/lib/libkmf/plugins/kmf_nss/Makefile.com

- When cross-building `libxml2` may not be installed on the base system and may be placed in `$ROOT`

```
--- a/usr/src/lib/libkmf/plugins/kmf_nss/Makefile.com Tue Aug 19 11:19:58 2008 -
0400
```

```
+++ b/usr/src/lib/libkmf/plugins/kmf_nss/Makefile.com Tue Sep 02 11:15:52 2008 -
0400
@@ -48,7 +48,8 @@

 CFLAGS              +=      $(CCVERBOSE)
 CPPFLAGS     +=     -D_REENTRANT $(KMFINC) $(NSSINC)   \
-             -I$(SFWDIR)/include -I$(INCDIR) -I/usr/include/libxml2
+             -I$(SFWDIR)/include -I$(INCDIR) -I$(ROOT)/usr/include/libxml2
       \
+             -I/usr/include/libxml2

 PICS= $(OBJECTS:%=pics/%)
```

14.1.211 /usr/src/lib/libkmf/kmf_openssl/Makefile.com

- When cross-building `libxml2` may not be installed on the base system and may be placed in `$ROOT`

```
--- a/usr/src/lib/libkmf/plugins/kmf_openssl/Makefile.com   Tue Aug 19 11:19:58
2008 -0400
+++ b/usr/src/lib/libkmf/plugins/kmf_openssl/Makefile.com   Tue Sep 02 11:15:53
2008 -0400
@@ -51,7 +51,8 @@

 CFLAGS              +=      $(CCVERBOSE)
 CPPFLAGS     +=     -D_REENTRANT $(KMFINC) $(OPENSSL_CPPFLAGS) \
-                    -I$(INCDIR) -I/usr/include/libxml2
+                    -I$(INCDIR) -I$(ROOT)/usr/include/libxml2 \
+                    -I/usr/include/libxml2

 PICS= $(OBJECTS:%=pics/%)
 SONAME=      $(DYNLIB)
```

14.1.212 /usr/src/lib/libkmf/kmf_pkcs11/Makefile.com

- When cross-building `libxml2` may not be installed on the base system and may be placed in `$ROOT`

```
--- a/usr/src/lib/libkmf/plugins/kmf_pkcs11/Makefile.com    Tue Aug 19 11:19:58
2008 -0400
+++ b/usr/src/lib/libkmf/plugins/kmf_pkcs11/Makefile.com    Tue Sep 02 11:15:53
2008 -0400
@@ -52,7 +52,8 @@


 CFLAGS              +=      $(CCVERBOSE)
-CPPFLAGS     +=     -D_REENTRANT $(KMFINC) -I$(INCDIR)  -I/usr/include/libxml2
-I$(BIGNUMDIR)
+CPPFLAGS     +=     -D_REENTRANT $(KMFINC) -I$(INCDIR) -
I$(ROOT)/usr/include/libxml2 \
+                    -I/usr/include/libxml2 -I$(BIGNUMDIR)
 LINTFLAGS64  +=     -errchk=longptr64

 PICS= $(OBJECTS:%=pics/%)
```

14.1.213 /usr/src/lib/libkmf/kmf_openssl/Makefile.com

- When cross-building `mps` may not be installed on the base system and may be placed in `$ROOT`

```
--- a/usr/src/lib/libldap5/Makefile.com Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libldap5/Makefile.com Tue Sep 02 11:15:53 2008 -0400
@@ -59,7 +59,7 @@
 include ../../Makefile.lib

 NSS_LIBS=    -lnspr4 -lplc4 -lnss3 -lssl3
-NSS_HDRS=    /usr/include/mps
+NSS_HDRS=    -I$(ROOT)/usr/include/mps -I/usr/include/mps
 NSS_LDPATH=  /usr/lib/mps
 NSS_LDPATH64=      $(NSS_LDPATH)/64

@@ -74,7 +74,7 @@

 # Include directories for all files
 COM_INC=     -I$(SRC)/lib/libldap5/include/ldap \
-             -I$(NSS_HDRS)
+             $(NSS_HDRS)

 SRCS=        $(BEROBJS:%.o=../sources/ldap/ber/%.c) \
              $(LDAPOBJS:%.o=../sources/ldap/common/%.c) \
```

14.1.214 /usr/lib/libnisdb/db_headers.h

- gcc requires that `verbose` and `dbenv` be defined inside `extern "C" {}`.

```
--- a/usr/src/lib/libnisdb/db_headers.h Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libnisdb/db_headers.h Tue Sep 02 11:15:53 2008 -0400
@@ -36,8 +36,14 @@
 #include <stdlib.h>
 #include <setjmp.h>

+#ifdef       __cplusplus
+extern "C" {
+#endif
 extern int verbose;
 extern jmp_buf dbenv;
+#ifdef       __cplusplus
+}
+#endif

 #define      FATAL(msg, fcode) \
       { \
```

14.1.215 /usr/src/lib/libnsl/dial/callers.c

- Move static function prototypes from within routine to global area: gcc doesn't cope with prototypes declared this way.

```
--- a/usr/src/lib/libnsl/dial/callers.c Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libnsl/dial/callers.c Tue Sep 02 11:15:53 2008 -0400
@@ -37,6 +37,7 @@
 static char *fdig(char *);
 #ifndef SMALL
```

```
 static char *strecpy(char *, char *, char *);
+static char *currdial(void);
 #endif
 static int interface(const char *);
 static int fd_mklock(int);
@@ -48,6 +49,11 @@
 static int  Modemctrl;
 static unsigned connecttime;
 static int (*Setup)();
+static int pop_push(int);
+static void setdevcfg(char *, char *);
+static void ttygenbrk(int);
+static void dialreset(void);
+static struct netbuf      *stoa(char *, struct netbuf *);

 /*
  *    to add a new caller:
@@ -178,12 +184,9 @@
       struct caller *ca;
       char *args[D_MAX+1], dcname[20];
       char **sdev;
-      static int pop_push(int);
-      static void setdevcfg(char *, char *);
       int nullfd;
       char *phonecl;                    /* clear phone string */
       char phoneex[2*(MAXPH+2)]; /* expanded phone string */
-      static void ttygenbrk(int);
       struct termio tty_orig;
       int ret_orig = -1;

@@ -409,10 +412,6 @@
 {
       static char *info;  /* dynamically allocated MAXLINE */
       int na;
-      static void dialreset(void);
-#ifndef SMALL
-      static char *currdial(void);
-#endif

       DEBUG(2, "gdial(%s) called\n", type);
       if (info == NULL) {
@@ -474,8 +473,6 @@
       struct t_bind *bind_ret = 0;
       struct t_info tinfo;
       struct t_call *sndcall = 0, *rcvcall = 0;
-
-      static struct netbuf       *stoa(char *, struct netbuf *);

       if (dev[D_LINE][0] != '/') {
             /*    dev holds device name relative to /dev  */
```

14.1.216 /usr/src/lib/libnsl/dial/conn.c

- Move static function prototypes from within routine to global area: gcc doesn't cope with prototypes declared this way.

```
--- a/usr/src/lib/libnsl/dial/conn.c    Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libnsl/dial/conn.c    Tue Sep 02 11:15:53 2008 -0400
@@ -67,6 +67,8 @@
 static int notin(char *, char *);
 static int ifdate(char *);
```

```
 static int classmatch(char *[], char *[]);
+static void sysreset(void);
+static void devreset(void);

 static char *Myline = CNULL;     /* to force which line will be used */
 static char *Mytype = CNULL;     /* to force selection of specific device type
*/
@@ -85,7 +87,6 @@
 {
       int nf, fn = FAIL;
       char *flds[F_MAX+1];
-      static void sysreset(void);

       CDEBUG(4, "conn(%s)\n", system);
       Uerror = 0;
@@ -137,7 +138,6 @@
       int dcf = -1;
       int reread = 0;
       int tries = 0;       /* count of call attempts - for limit purposes */
-      static void devreset(void);

       CDEBUG(1, "Device Type %s wanted\n", flds[F_TYPE]);
       Uerror = 0;
```

14.1.217 /usr/src/lib/libnsl/dial/callers.c

- `mapbuf` is declared as `char` which on some platforms is `signed` and on others `unsigned`. To ensure the comparison with `EOF` (-1) works cast the variable to a `signed char`.

```
--- a/usr/src/lib/libnsl/nss/parse.c    Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libnsl/nss/parse.c    Tue Sep 02 11:15:53 2008 -0400
@@ -192,7 +192,7 @@
             }
             /* Buffer overflow -- eat rest of line and loop again */
             while (mapbuf[*lastlen] != '\n') {
-                  if (mapbuf[*lastlen] == EOF) {
+                  if ((int8_t) mapbuf[*lastlen] == EOF) {
                         return (-1);
                   }
                   (*lastlen)++;
```

14.1.218 /usr/src/lib/libnsl/rpc/svc_vc.c

- Move static function prototypes from within routine to global area: gcc doesn't cope with prototypes declared this way.

```
--- a/usr/src/lib/libnsl/rpc/svc_vc.c   Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libnsl/rpc/svc_vc.c   Tue Sep 02 11:15:53 2008 -0400
@@ -91,6 +91,7 @@
 static SVCXPRT           *makefd_xprt(int, uint_t, uint_t, t_scalar_t, char
*);
 static bool_t            fd_is_dead(int);
 static void        update_nonblock_timestamps(SVCXPRT *);
+static void        do_accept();

 struct cf_rendezvous { /* kept in xprt->xp_p1 for rendezvouser */
       uint_t sendsize;
@@ -679,7 +680,6 @@
```

```
        struct cf_rendezvous *r;
        char *tpname = NULL;
        char devbuf[256];
-       static void do_accept();

 /* LINTED pointer alignment */
        r = (struct cf_rendezvous *)xprt->xp_p1;
```

14.1.219 /usr/src/lib/libpicltree/llib-lpicltree

- Have the prototypes used my lint match the definition used elsewhere.

```
--- a/usr/src/lib/libpicltree/llib-lpicltree   Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libpicltree/llib-lpicltree   Tue Sep 02 11:15:53 2008 -0400
@@ -50,12 +50,12 @@
 int     ptree_add_row_to_table(picl_prophdl_t tbl, int nprops,
                         const picl_prophdl_t *props);
 int     ptree_update_propval_by_name(picl_nodehdl_t nodeh, const char *name,
-                        const void *vbuf, unsigned int sz);
+                        const void *vbuf, size_t sz);
 int     ptree_update_propval(picl_prophdl_t proph, const void *buf,
-                        unsigned int sz);
-int     ptree_get_propval(picl_prophdl_t proph, void *buf, unsigned int sz);
+                        size_t sz);
+int     ptree_get_propval(picl_prophdl_t proph, void *buf, size_t sz);
 int     ptree_get_propval_by_name(picl_nodehdl_t nodeh, const char *name,
-                        void *buf, unsigned int sz);
+                        void *buf, size_t sz);
 int     ptree_get_propinfo(picl_prophdl_t proph, ptree_propinfo_t *pi);
 int     ptree_get_first_prop(picl_nodehdl_t nodeh, picl_prophdl_t *proph);
 int     ptree_get_next_prop(picl_prophdl_t thish, picl_prophdl_t *proph);
```

14.1.220 /usr/src/lib/libpool/Makefile.com

- When cross-building `libxml2` may not be installed on the base system and may be placed in `$ROOT`

```
--- a/usr/src/lib/libpool/Makefile.com  Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libpool/Makefile.com  Tue Sep 02 11:15:53 2008 -0400
@@ -50,7 +50,8 @@
 $(LINTLIB) :=      SRCS = $(SRCDIR)/$(LINTSRC)

 CFLAGS +=    $(CCVERBOSE)
-CPPFLAGS +=  -D_REENTRANT -D_FILE_OFFSET_BITS=64 -I/usr/include/libxml2
+CPPFLAGS +=  -D_REENTRANT -D_FILE_OFFSET_BITS=64 -I$(ROOT)/usr/include/libxml2
\
+             -I/usr/include/libxml2


 .KEEP_STATE:
```

14.1.221 /usr/src/lib/libdll/Makefile.com

- Need to search `$(ROOT)/usr/include` and not `/usr/include`.

```
--- a/usr/src/lib/libpp/Makefile.com    Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libpp/Makefile.com    Tue Sep 02 11:15:53 2008 -0400
@@ -82,6 +82,7 @@
        $(DTEXTDOM) $(DTS_ERRNO) \
```

```
       -I. \
       -I$(ROOT)/usr/include/ast \
+      -I$(ROOT)/usr/include      \
       -D_PACKAGE_ast \
       '-DUSAGE_LICENSE=\
             "[-author?Glenn Fowler <gsf@research.att.com>]"\
```

14.1.222 /usr/src/lib/libresolv/res_gethost.c

- Move static function prototypes from within routine to global area: gcc doesn't cope with prototypes declared this way.

```
--- a/usr/src/lib/libresolv/res_gethost.c      Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libresolv/res_gethost.c      Tue Sep 02 11:15:53 2008 -0400
@@ -29,6 +29,10 @@
 #include <arpa/nameser.h>
 #include <resolv.h>
 #include <syslog.h>
+
+struct hostent *hp, *gethostdomain();
+static struct hostent *_gethtbyname();
+static struct hostent *_gethtbyaddr();

 /*
  * When the name service switch calls libresolv, it doesn't want fallback
@@ -105,7 +109,7 @@
       cp = answer->buf + sizeof (HEADER);
       if (qdcount) {
             if (iquery) {
-                  if ((n = dn_expand((char *)answer->buf, eom,
+                  if ((n = dn_expand((unsigned char *)answer->buf, eom,
                                      cp, bp, buflen)) < 0) {
                         h_errno = NO_RECOVERY;
                         return ((struct hostent *) NULL);
@@ -134,7 +138,7 @@
 #endif
       haveanswer = 0;
       while (--ancount >= 0 && cp < eom && haveanswer < MAXADDRS) {
-            if ((n = dn_expand((char *)answer->buf, eom,
+            if ((n = dn_expand((unsigned char *)answer->buf, eom,
                                      cp, bp, buflen)) < 0)
                   break;
             cp += n;
@@ -155,7 +159,7 @@
                         continue;
             }
             if (iquery && type == T_PTR) {
-                  if ((n = dn_expand((char *)answer->buf, eom,
+                  if ((n = dn_expand((unsigned char *)answer->buf, eom,
                                      cp, bp, buflen)) < 0) {
                         cp += n;
                         continue;
@@ -231,8 +235,6 @@
       querybuf buf;
       register char *cp;
       int n;
-      struct hostent *hp, *gethostdomain();
-      static struct hostent *_gethtbyname();

       /*
        * disallow names consisting only of digits/dots, unless
@@ -272,7 +274,6 @@
```

```
        querybuf buf;
        register struct hostent *hp;
        char qbuf[MAXDNAME];
-       static struct hostent *_gethtbyaddr();

        if (type != AF_INET)
                return ((struct hostent *) NULL);
@@ -281,7 +282,7 @@
                ((unsigned)addr[2] & 0xff),
                ((unsigned)addr[1] & 0xff),
                ((unsigned)addr[0] & 0xff));
-       n = res_query(qbuf, C_IN, T_PTR, (char *)&buf, sizeof (buf));
+       n = res_query(qbuf, C_IN, T_PTR, (unsigned char *)&buf, sizeof (buf));
        if (n < 0) {
 #ifdef DEBUG
                if (_res.options & RES_DEBUG)
```

14.1.223 /usr/src/lib/libdll/Makefile.com

- Need to search `$(ROOT)/usr/include` and not `/usr/include`.

```
--- a/usr/src/lib/libresolv2/include/Makefile  Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libresolv2/include/Makefile  Tue Sep 02 11:15:53 2008 -0400
@@ -55,6 +55,6 @@
        ./make_os_version

 port_ipv6.h: probe_ipv6 FRC
-       CC="$(CC)" ./probe_ipv6
+       CC="$(CC) -I$(ROOT)/usr/include" ./probe_ipv6

 FRC:
```

14.1.224 /usr/src/lib/libdll/Makefile.com

- Need to search `$(ROOT)/usr/include` and not `/usr/include`.
- Search for `libgcc_s` when building with gcc

```
--- a/usr/src/lib/libscf/Makefile.com   Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libscf/Makefile.com   Tue Sep 02 11:15:53 2008 -0400
@@ -58,10 +58,14 @@
 # of libuutil.
 #
 LIBUUTIL =   $(SRC)/lib/libuutil
+s390_NCPPFL =      -_gcc=-nostdinc -I$(ROOT)/usr/include
 MY_NATIVE_CPPFLAGS =\
                -DNATIVE_BUILD $(DTEXTDOM) \
+              $($(MACH)_NCPPFL)    \
                -I../inc -I$(COMDIR) -I$(LIBUUTIL)/common
-MY_NATIVE_LDLIBS = -L$(LIBUUTIL)/native -R$(LIBUUTIL)/native -luutil -ldoor -
lc \
+s390_NLDLIBS =     -L$(ROOT)/usr/lib
+MY_NATIVE_LDLIBS = $($(MACH)_NLDLIBS)   \
+              -L$(LIBUUTIL)/native -R$(LIBUUTIL)/native -luutil -ldoor -lc \
                -lgen

 .KEEP_STATE:
```

14.1.225 /usr/src/lib/libshare/Makefile.com

- When cross-building `libxml2` may not be installed on the base system and may be placed in `$ROOT`

```
--- a/usr/src/lib/libshare/Makefile.com Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libshare/Makefile.com Tue Sep 02 11:15:53 2008 -0400
@@ -46,7 +46,8 @@

 #add nfs/lib directory as part of the include path
 CFLAGS +=    $(CCVERBOSE)
-CPPFLAGS +=  -D_REENTRANT -I$(NFSLIB_DIR) -I/usr/include/libxml2
+CPPFLAGS +=  -D_REENTRANT -I$(NFSLIB_DIR) -I$(ROOT)/usr/include/libxml2 \
+             -I/usr/include/libxml2

 .KEEP_STATE:
```

14.1.226 /usr/src/lib/libshare/nfs/Makefile.com

- When cross-building `libxml2` may not be installed on the base system and may be placed in `$ROOT`

```
--- a/usr/src/lib/libshare/nfs/Makefile.com    Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libshare/nfs/Makefile.com    Tue Sep 02 11:15:53 2008 -0400
@@ -48,8 +48,9 @@

 #add nfs/lib directory as part of the include path
 CFLAGS +=    $(CCVERBOSE)
-CPPFLAGS +=  -D_REENTRANT -I$(NFSLIB_DIR) -I/usr/include/libxml2 \
-                   -I$(SRCDIR)/../common
+CPPFLAGS +=  -D_REENTRANT -I$(NFSLIB_DIR) -I$(ROOT)/usr/include/libxml2  \
+             -I/usr/include/libxml2 \
+             -I$(SRCDIR)/../common

 .KEEP_STATE:
```

14.1.227 /usr/src/lib/libshare/smb/Makefile.com

- When cross-building `libxml2` may not be installed on the base system and may be placed in `$ROOT`

```
--- a/usr/src/lib/libshare/smb/Makefile.com    Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libshare/smb/Makefile.com    Tue Sep 02 11:15:53 2008 -0400
@@ -50,7 +50,8 @@
 all install := LDLIBS += -lxml2

 CFLAGS +=    $(CCVERBOSE)
-CPPFLAGS +=  -D_REENTRANT -I/usr/include/libxml2 \
+CPPFLAGS +=  -D_REENTRANT -I$(ROOT)/usr/include/libxml2 \
+             -I/usr/include/libxml2 \
              -I$(SRCDIR)/../common

 .KEEP_STATE:
```

14.1.228 /usr/src/lib/libshare/smbfs/Makefile.com

- When cross-building `libxml2` may not be installed on the base system and may be placed in `$ROOT`

```
--- a/usr/src/lib/libshare/smbfs/Makefile.com  Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libshare/smbfs/Makefile.com  Tue Sep 02 11:15:53 2008 -0400
@@ -43,7 +43,8 @@
 LDLIBS +=    -lshare -lscf -lumem -luuid -lc -lxml2 -lsmbfs

 CFLAGS +=    $(CCVERBOSE)
-CPPFLAGS +=  -D_REENTRANT -I/usr/include/libxml2 -I$(SRCDIR)/../common \
+CPPFLAGS +=  -D_REENTRANT -I$(ROOT)/usr/include/libxml2 \
+             -I/usr/include/libxml2 -I$(SRCDIR)/../common \
              -I$(SRC)/lib/libsmbfs -I$(SRC)/uts/common

 .KEEP_STATE:
```

14.1.229 /usr/src/lib/libsqlite/Makefile.com

- Need to search `$(ROOT)/usr/include` and not `/usr/include`.
- Use `$($(MACH)_MAPFILES)` mechanism to support platform specific map file processing.
- Kludge (to be fixed): for cross-builds we need to build lemon using the compiler of the cross-building platform.

```
--- a/usr/src/lib/libsqlite/Makefile.com       Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libsqlite/Makefile.com       Tue Sep 02 11:15:53 2008 -0400
@@ -90,10 +90,14 @@
       $(SRCDIR)/where.c    \
       $(SRCDIR)/trigger.c

-MYCPPFLAGS = -D_REENTRANT -DTHREADSAFE=1 -DHAVE_USLEEP=1 -I. -I.. -I$(SRCDIR)
+MYCPPFLAGS =       -D_REENTRANT -DTHREADSAFE=1 -DHAVE_USLEEP=1 -I. -I.. -
I$(SRCDIR)    \
+             -I$(ROOT)/usr/include
 CPPFLAGS += $(MYCPPFLAGS)

-MAPFILES = ../mapfile-sqlite
+sparc_MAPFILES = -M../mapfile-sqlite
+i386_MAPFILES = -M../mapfile-sqlite
+s390_MAPFILES =
+MAPFILES = $($(MACH)_MAPFILES)

 # Header files used by all library source files.
 #
@@ -213,11 +217,11 @@
 native: $(NATIVERELOC)

 $(RELOC): objs .WAIT $(OBJS)
-      $(LD) -r $(MAPFILES:%=-M%) -o $(RELOC) $(OBJS)
+      $(LD) -r $(MAPFILES) -o $(RELOC) $(OBJS)
       $(CTFMERGE) -t -f -L VERSION -o $(RELOC) $(OBJS)

 $(NATIVERELOC):    objs .WAIT $(OBJS:%.o=%-native.o)
-      $(LD) -r $(MAPFILES:%=-M%) -o $(NATIVERELOC) $(OBJS:%.o=%-native.o)
+      $(LD) -r $(MAPFILES) -o $(NATIVERELOC) $(OBJS:%.o=%-native.o)
```

```
 opcodes.h: $(SRCDIR)/vdbe.c
       @echo "Generating $@"; \
@@ -244,7 +248,12 @@
       $(MAKE) lemon-build

 lemon-build: lemon.o $(TOOLDIR)/lempar.c
-      $(LINK.c) -o lemon lemon.o
+      @if [ $(MACH) = "s390" ]; then                          \
+            $(sparc_CC) -c -o lemon.o $(TOOLDIR)/lemon.c;     \
+            $(sparc_CC) -o lemon lemon.o;                     \
+      else                                                    \
+            $(LINK.c) -o lemon lemon.o;                       \
+      fi
       $(RM) lempar.c
       $(LN) -s $(TOOLDIR)/lempar.c lempar.c
       $(RM) lemon-build
```

14.1.230 /usr/src/lib/libtsol/common/labeld.h

- Correct cast of COOKIE (only used in comparison of `door_ptr_t` variable).

```
--- a/usr/src/lib/libtsol/common/labeld.h      Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libtsol/common/labeld.h      Tue Sep 02 11:15:53 2008 -0400
@@ -45,7 +45,7 @@

 #define      DOOR_PATH     "/var/tsol/doors/"
 #define      DOOR_NAME     "labeld"
-#define      COOKIE        (void *)0x6c616264  /* "labd" */
+#define      COOKIE        (door_ptr_t)0x6c616264     /* "labd" */

 /*    Op codes */
```

14.1.231 /usr/src/lib/libumem/Makefile.com

- Use `$($(MACH)_SAPFLAGS)` mechanism to support platform specific standalone linker flags.

```
--- a/usr/src/lib/libumem/Makefile.com  Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libumem/Makefile.com  Tue Sep 02 11:15:53 2008 -0400
@@ -117,8 +117,10 @@

 LDLIBS +=    -lc

-LDFLAGS_standalone = $(ZNOVERSION) $(BREDUCE) -M../common/mapfile-vers \
-      -M$(MAPFILE_SUPPLEMENTAL) -dy -r
+sparc_SAFLAGS= -M../common/mapfile-vers -M$(MAPFILE_SUPPLEMENTAL)
+i386_SAFLAGS= -M../common/mapfile-vers -M$(MAPFILE_SUPPLEMENTAL)
+s390_SAFLAGS=
+LDFLAGS_standalone = $(ZNOVERSION) $(BREDUCE) -dy -r $($(MACH)_SAFLAGS)
 LDFLAGS = $(LDFLAGS_$(CURTYPE))

 ASFLAGS_standalone = -DUMEM_STANDALONE
```

14.1.232 /usr/src/lib/libuutil/Makefile.com

- Use `$($(MACH)_NCPPFLG)` mechanism to support platform specific C flags.

```
--- a/usr/src/lib/libuutil/Makefile.com Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libuutil/Makefile.com Tue Sep 02 11:15:53 2008 -0400
@@ -74,8 +74,10 @@
 LINTFLAGS += -erroff=E_GLOBAL_COULD_BE_STATIC2
 LINTFLAGS64 +=     -erroff=E_GLOBAL_COULD_BE_STATIC2

-MY_NATIVE_CPPFLAGS = -DNATIVE_BUILD -I$(SRCDIR)
-MY_NATIVE_LDLIBS = -lc
+s390_NCPPFLG = -_gcc=-nostdinc -I$(ROOT)/usr/include
+MY_NATIVE_CPPFLAGS = $($(MACH)_NCPPFLG) -DNATIVE_BUILD -I$(SRCDIR)
+s390_NLDLIBS = -L$(ROOT)/usr/lib
+MY_NATIVE_LDLIBS = $($(MACH)_NLDLIBS) -lc

 $(NOT_RELEASE_BUILD)CPPFLAGS += -DDEBUG
```

14.1.233 /usr/src/lib/libvolmgt/common/volmgt_on_private.c

- Move static function prototypes from within routine to global area: gcc doesn't cope with prototypes declared this way.

```
--- a/usr/src/lib/libvolmgt/common/volmgt_on_private.c      Tue Aug 19 11:19:58
2008 -0400
+++ b/usr/src/lib/libvolmgt/common/volmgt_on_private.c      Tue Sep 02 11:15:53
2008 -0400
@@ -55,7 +55,11 @@

 #define     NULL_PATH           "/dev/null"

-
+static int   vol_getmntdev(FILE *, struct mnttab *, dev_t,
+                     struct dk_cinfo *);
+static int   call_unmount_prog(int, int, char *, int, char *,
+                     char *);
+static int   get_media_info(char *, char **, int *, char **);

 /*
  * This is an ON Consolidation Private interface.
@@ -72,8 +76,6 @@
 int
 _dev_mounted(char *path)
 {
-      static int    vol_getmntdev(FILE *, struct mnttab *, dev_t,
-                        struct dk_cinfo *);
       int           fd = -1;
       struct dk_cinfo     info;
       static FILE   *fp = NULL;         /* mnttab file pointer */
@@ -133,9 +135,6 @@
 int
 _dev_unmount(char *path)
 {
-      static int    call_unmount_prog(int, int, char *, int, char *,
-                        char *);
-      static int    get_media_info(char *, char **, int *, char **);
       char          *bn = NULL;         /* block name */
       char          *mtype = NULL;            /* media type */
       char          *spcl = NULL;       /* special dev. path */
@@ -409,8 +408,6 @@
 static int
 get_media_info(char *path, char **mtypep, int *mnump, char **spclp)
 {
-      static int    vol_getmntdev(FILE *, struct mnttab *, dev_t,
```

```
-                       struct dk_cinfo *);
        FILE            *fp = NULL;
        int             fd = -1;
        char            *cn = NULL;         /* char spcl pathname */
```

14.1.234 /usr/src/lib/libzonecfg/Makefile.com

- When cross-building `libxml2` may not be installed on the base system and may be placed in `$ROOT`

```
--- a/usr/src/lib/libzonecfg/Makefile.com       Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/libzonecfg/Makefile.com       Tue Sep 02 11:15:53 2008 -0400
@@ -38,7 +38,7 @@
 $(DYNLIB) := LDLIBS += -lxml2

 SRCDIR =    ../common
-CPPFLAGS += -I/usr/include/libxml2 -I$(SRCDIR) -D_REENTRANT
+CPPFLAGS += -I$(ROOT)/usr/include/libxml2 -I/usr/include/libxml2 -I$(SRCDIR) -
D_REENTRANT
 $(LINTLIB) := SRCS=       $(SRCDIR)/$(LINTSRC)

 .KEEP_STATE:
```

14.1.235 /usr/src/lib/pam_modules/sample/sample_acct_mgmt.c

- Move static function prototypes from within routine to global area: gcc doesn't cope with prototypes declared this way.

```
--- a/usr/src/lib/pam_modules/sample/sample_acct_mgmt.c     Tue Aug 19 11:19:58
2008 -0400
+++ b/usr/src/lib/pam_modules/sample/sample_acct_mgmt.c     Tue Sep 02 11:15:53
2008 -0400
@@ -35,6 +35,7 @@
 #include <libintl.h>

 static int parse_allow_name(char *, char *);
+static char *getname();

 /*
  * pam_sm_acct_mgmt       main account managment routine.
@@ -110,7 +111,6 @@
 parse_allow_name(char *who, char *cp)
 {
        char name[256];
-       static char *getname();

        /* catch "allow=" */
        if (*cp == '\0')
```

14.1.236 /usr/src/lib/policykit/Makefile.policykit

- When cross-building `dbus` and `glib` may not be installed on the base system and may be placed in `$ROOT`

```
--- a/usr/src/lib/policykit/Makefile.policykit Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/policykit/Makefile.policykit Tue Sep 02 11:15:54 2008 -0400
@@ -29,8 +29,10 @@
```

```
 POLICYKIT_VERSION = 0.2

-POLICYKIT_DBUS_CPPFLAGS = -DDBUS_API_SUBJECT_TO_CHANGE -I/usr/include/dbus-1.0
-I/usr/lib/dbus-1.0/include
-POLICYKIT_GLIB_CPPFLAGS =       -I/usr/include/glib-2.0 -I/usr/lib/glib-
2.0/include
+POLICYKIT_DBUS_CPPFLAGS = -DDBUS_API_SUBJECT_TO_CHANGE -
I$(ROOT)/usr/include/dbus-1.0 \
+                          -I/usr/include/dbus-1.0 -I/usr/lib/dbus-1.0/include
+POLICYKIT_GLIB_CPPFLAGS =       -I$(ROOT)/usr/include/glib-2.0    \
+                          -I/usr/include/glib-2.0 -I/usr/lib/glib-2.0/include

 POLICYKIT_DBUS_LDLIBS =   -ldbus-1
 POLICYKIT_GLIB_LDLIBS =   -lglib-2.0
```

14.1.237 /usr/src/lib/scsi/plugins/ses/Makefile.lib

- Use `$($(MACH)_APIMAPFLAG)` mechanism to support platform specific map linker flags.

```
--- a/usr/src/lib/scsi/plugins/ses/Makefile.lib      Tue Aug 19 11:19:58 2008 -
0400
+++ b/usr/src/lib/scsi/plugins/ses/Makefile.lib      Tue Sep 02 11:15:54 2008 -
0400
@@ -69,6 +69,10 @@
 DMODLINTFILES = $(DMOD_SRCS:%.c=%.ln)

 APIMAP = ../../../../libses/libses_api.map
+sparc_APIMAPFLAG = -M
+i386_APIMAPFLAG = -M
+s390_APIMAPFLAG = -_gcc="--version-script="
+APIMAPFLAG = $($(MACH)_APIMAPFLAG)

 C99MODE = $(C99_ENABLE)
 CFLAGS += $(CTF_FLAGS) $(CCVERBOSE) $(XSTRCONST) $(CC_PICFLAGS)
@@ -79,7 +83,7 @@
 $(NOT_RELEASE_BUILD)CPPFLAGS += -DDEBUG
 LDFLAGS += $(ZTEXT) $(ZIGNORE)

-$(PROG) := LDFLAGS += $(ZDEFS) -M$(APIMAP)
+$(PROG) := LDFLAGS += $(ZDEFS) $(APIMAPFLAG)$(APIMAP)
 $(PROG) := LDLIBS += -lc -lnvpair

 $(DMODPROG) := LDFLAGS += $(ZNODEFS)
```

14.1.238 /usr/src/lib/smbsrv/Makefile.targ

- Enclose `$(CC)` in double-quotes so that `ndrgen` doesn't interpret any parameters within `$(CC)` as its own arguments.

```
--- a/usr/src/lib/smbsrv/Makefile.targ  Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/lib/smbsrv/Makefile.targ  Tue Sep 02 11:15:54 2008 -0400
@@ -29,7 +29,7 @@
 #

 %_ndr.c: $(NDLDIR)/%.ndl
-       $(NDRGEN) -Y $(CC) $<
+       $(NDRGEN) -Y "$(CC)" $<
```

```
pics/%.o:    $(SRC)/common/smbsrv/%.c
      $(COMPILE.c) -o $@ $<
```

14.1.239 /usr/src/lib/smbsrv/libsmbns/common/smbns_netbios_cache.c

- Give hint to the compiler that we're only after the last byte of the `name->attributes` variable.

```
--- a/usr/src/lib/smbsrv/libsmbns/common/smbns_netbios_cache.c     Tue Aug 19
11:19:58 2008 -0400
+++ b/usr/src/lib/smbsrv/libsmbns/common/smbns_netbios_cache.c     Tue Sep 02
11:15:54 2008 -0400
@@ -428,7 +428,7 @@
                    bcopy(name->name, scan, NETBIOS_NAME_SZ);
                    scan += NETBIOS_NAME_SZ;
                    *scan++ = PUBLIC_BITS(name->attributes) >> 8;
-                   *scan++ = PUBLIC_BITS(name->attributes);
+                   *scan++ = (PUBLIC_BITS(name->attributes)) & 0xff;
                    (*numnames)++;
             }
```

14.1.240 /usr/src/tools/Makefile

- Add support for System z.

```
--- a/usr/src/tools/Makefile      Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/tools/Makefile      Tue Sep 02 11:15:54 2008 -0400
@@ -68,7 +68,7 @@
       elfextract    \
       mbh_patch

-s390x_SUBDIRS=             \
+s390_SUBDIRS=              \
       aw             \
       kipl_cvt       \
       stabs
```

14.1.241 /usr/src/tools/ctf/cvt/fixup_tdescs.c

- gcc doesn't support the initialization of `match` in the way Sun Studio does so split it out into two operations.

```
--- a/usr/src/tools/ctf/cvt/fixup_tdescs.c     Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/tools/ctf/cvt/fixup_tdescs.c     Tue Sep 02 11:15:54 2008 -0400
@@ -178,7 +178,8 @@
 static tdesc_t *
 lookup_tdesc(tdata_t *td, const char *name)
 {
-      struct match match = { NULL, name };
+      struct match match = { NULL, NULL };
+      match.m_name = name;
       iter_iidescs_by_name(td, name, (int (*)())matching_iidesc, &match);
       return (match.m_ret);
 }
```

14.1.242 /usr/src/tools/cw/cw.c

- Add System z specific flags
- Add flags to turn off warnings in gcc by default
- For cross-builds cw will be built as an executable on the platform on which the cross-build is being performed. However, it will need to understand and support the System z flags.

```
--- a/usr/src/tools/cw/cw.c      Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/tools/cw/cw.c      Tue Sep 02 11:15:54 2008 -0400
@@ -388,7 +388,7 @@
       { "amd64",    (SS11|M64), { "-m64", "-mtune=opteron" } },
       { "386",      SS11,  { "-march=i386" } },
       { "pentium_pro", SS11,     { "-march=pentiumpro" } },
-#elif defined(__sparc)
+#elif defined(__sparc) && !defined(__s390)
       { "generic",  (SS11|M32), { "-m32", "-mcpu=v8" } },
       { "generic64",      (SS11|M64), { "-m64", "-mcpu=v9" } },
       { "v8",             (SS11|M32), { "-m32", "-mcpu=v8", "-mno-v8plus" } },
@@ -531,6 +531,8 @@
             return;

       newae(h, "-Wall");
+      newae(h, "-Wno-address");
+      newae(h, "-Wno-pointer-sign");
       newae(h, "-Wno-unknown-pragmas");
       newae(h, "-Wno-missing-braces");
       newae(h, "-Wno-sign-compare");
@@ -541,6 +543,8 @@
       newae(h, "-Wno-trigraphs");
       newae(h, "-Wno-char-subscripts");
       newae(h, "-Wno-switch");
+      newae(h, "-Wno-int-to-pointer-cast");
+      newae(h, "-Wno-pointer-to-int-cast");
 }

 static void
@@ -661,7 +665,7 @@
       newae(ctx->i_ae, "-fno-asm");
       newae(ctx->i_ae, "-nodefaultlibs");

-#if defined(__sparc)
+#if defined(__sparc) && !defined(__s390)
       /*
        * The SPARC ldd and std instructions require 8-byte alignment of
        * their address operand.  gcc correctly uses them only when the
@@ -1446,7 +1450,7 @@
       case 0:
             /* FALLTHROUGH */
       case M32:
-#if defined(__sparc)
+#if defined(__sparc) && !defined(__s390)
             /*
              * Only -m32 is defined and so put in the missing xarch
              * translation.
@@ -1456,7 +1460,7 @@
 #endif
             break;
       case M64:
-#if defined(__sparc)
```

```
+#if defined(__sparc) && !defined(__s390)
              /*
               * Only -m64 is defined and so put in the missing xarch
               * translation.
@@ -1477,7 +1481,7 @@
       case (SS11|M64):
              break;
       case (SS12|M32):
-#if defined(__sparc)
+#if defined(__sparc) && !defined(__s390)
              /*
               * Need to add in further 32 bit options because with SS12
               * the xarch=sparcvis option can be applied to 32 or 64
```

14.1.243 /usr/src/tools/elfsign/Makefile

```
--- a/usr/src/tools/elfsign/Makefile    Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/tools/elfsign/Makefile    Tue Sep 02 11:15:54 2008 -0400
@@ -51,7 +51,7 @@
 CPPFLAGS += -I$(SRC)/uts/common
 CPPFLAGS += -I$(SRC)/lib/libkmf/include
 CPPFLAGS += -I$(SRC)/lib/libcryptoutil/common
-LDFLAGS +=  -lmd -lelf -lkmf -lcryptoutil -lc
+LDFLAGS +=  -L/lib -lmd -lelf -lkmf -lcryptoutil -lc

 #
 # While the gate builds a libelfsign.so linked staticly against
```

14.1.244 /usr/src/uts/Makefile.uts

- Add `s390x` and `zSeries` as directories to be processed

```
--- a/usr/src/uts/Makefile.uts    Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/uts/Makefile.uts    Tue Sep 02 11:15:54 2008 -0400
@@ -340,7 +340,7 @@
 #
 CTFMERGE_GUDIR_sparc      = sun4u
 CTFMERGE_GUDIR_i386       = intel
-CTFMERGE_GUDIR_s390       = zSeries
+CTFMERGE_GUDIR_s390       = s390x
 CTFMERGE_GUDIR            = $(CTFMERGE_GUDIR_$(MACH))

 CTFMERGE_GENUNIX    = \
--- a/usr/src/uts/Makefile.uts  Tue Sep 02 11:22:56 2008 -0400
+++ b/usr/src/uts/Makefile.uts  Thu Sep 25 10:26:11 2008 -0400
@@ -431,6 +431,9 @@
 #      For now, 64b modules install into a subdirectory
 #      of their 32b brethren.
 #
+#      However, there is no 32b version of s390 so it will
+#      not have a s390x subdirectory
+#
 SUBDIR64_sparc          = sparcv9
 SUBDIR64_i386           = amd64
 SUBDIR64_s390           = s390x
```

14.1.245 /usr/src/uts/common/fs/nfs/nfs_server.c

- This appears to be a redundant definition. For non-System z platforms the `-N` flag in the Makefile contains this information. For System z a nfs_depends.s module is created with the same information within it.

```
--- a/usr/src/uts/common/fs/nfs/nfs_server.c   Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/uts/common/fs/nfs/nfs_server.c   Tue Sep 02 11:15:54 2008 -0400
@@ -102,7 +102,9 @@
       MODREV_1, (void *)&modlmisc, NULL
 };

+#ifndef __s390x
 char _depends_on[] = "misc/klmmod";
+#endif

 int
 _init(void)
```

14.1.246 /usr/src/uts/common/fs/smbclnt/smbfs/smbfs.h

- Include `sys/vfs_opreg.h` so that `struct fs_operation_def` is defined before it's used.

```
--- a/usr/src/uts/common/fs/smbclnt/smbfs/smbfs.h       Tue Sep 02 11:22:56
2008 -0400
+++ b/usr/src/uts/common/fs/smbclnt/smbfs/smbfs.h       Thu Sep 25 10:26:11
2008 -0400
@@ -53,7 +53,7 @@
 #include <sys/list.h>
 #include <sys/vfs.h>
 #include <sys/fs/smbfs_mount.h>
-
+#include <sys/vfs_opreg.h>

 /*
  * SM_MAX_STATFSTIME is the maximum time to cache statvfs data. Since this
```

14.1.247 /usr/src/uts/common/fs/sockfs/n17clogd.c

- Cast variable so we can do pointer arithmetic without gcc raising any warnings.

```
--- a/usr/src/uts/common/fs/sockfs/nl7clogd.c  Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/uts/common/fs/sockfs/nl7clogd.c  Tue Sep 02 11:15:54 2008 -0400
@@ -686,7 +686,7 @@
        * is allocated, and start all over.
        */
       pep = &((char *)log)[log->size];
-      wp = (log->buffer + log->cur_pos);
+      wp = ((char *)log->buffer + log->cur_pos);
       wp = NCA_LOG_ALIGN(wp);
       req = (nca_request_log_t *)wp;
       if ((wp + sizeof (*req)) >= pep) goto full;
@@ -714,7 +714,7 @@
       if (nl7c_http_log(quri, suri, req, &wp, &pep, &off)) goto full;

       /* Update logbuf */
```

```
-       log->cur_pos = (wp - log->buffer);
+       log->cur_pos = (wp - (char *)log->buffer);

        req->response_status = HS_OK;
```

14.1.248 /usr/src/uts/common/fs/udfs/udf_vnops.c

- Not sure these changes are required now we're using the `-Wno-pointer-sign` flag.

```
--- a/usr/src/uts/common/fs/udfs/udf_vnops.c   Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/uts/common/fs/udfs/udf_vnops.c   Tue Sep 02 11:15:54 2008 -0400
@@ -1250,7 +1250,8 @@
        struct ud_inode *ip, *dip = VTOI(dvp);

        struct path_comp *pc;
-       int8_t *dname = NULL, *uname = NULL, *sp;
+       int8_t *dname = NULL, *uname = NULL;
+       char *sp;

        ud_printf("udf_symlink\n");

@@ -1377,7 +1378,7 @@
        caller_context_t *ct)
 {
        int32_t error = 0, off, id_len, size, len;
-       int8_t *dname = NULL, *uname = NULL;
+       uint8_t *dname = NULL, *uname = NULL;
        struct ud_inode *ip;
        struct fbuf *fbp = NULL;
        struct path_comp *pc;
```

14.1.249 /usr/src/uts/common/fs/vnode.c

- The VOPCHK_PAGE() and VOPSET_PAGE() macros are required for System z to check change bits of page, it is a no-op on other platforms.

```
--- a/usr/src/uts/common/fs/vnode.c     Tue Sep 02 11:22:56 2008 -0400
+++ b/usr/src/uts/common/fs/vnode.c     Thu Sep 25 10:26:11 2008 -0400
@@ -3868,6 +3868,8 @@

         err = (*(vp)->v_op->vop_getpage)
             (vp, off, len, protp, plarr, plsz, seg, addr, rw, cr, ct);
+
+       VOPCHK_PAGE(vp, off, len, plarr);
         VOPSTATS_UPDATE(vp, getpage);
         return (err);
 }
@@ -3885,6 +3887,7 @@

         VOPXID_MAP_CR(vp, cr);

+       VOPSET_PAGE(vp, off, len);
         err = (*(vp)->v_op->vop_putpage)(vp, off, len, flags, cr, ct);
         VOPSTATS_UPDATE(vp, putpage);
         return (err);
```

14.1.250 /usr/src/uts/common/inet/ipf/netinet/ip_lookup.h

- Variable `ilik_unit` is used in signed comparisons but on some platforms `char` defaults to `signed` and on others to `unsigned`.

```
--- a/usr/src/uts/common/inet/ipf/netinet/ip_lookup.h Tue Aug 19 11:19:58 2008 -
0400
+++ b/usr/src/uts/common/inet/ipf/netinet/ip_lookup.h Tue Sep 02 11:15:54 2008 -
0400
@@ -70,7 +70,7 @@
              char   ilik_ival;
              u_char ilik_type;    /* IPLT_* */
              u_char ilik_otype;
-             char   ilik_unit;    /* IPL_LOG* */
+             int8_t ilik_unit;    /* IPL_LOG* */
       } ilik_unstr;
       u_32_t ilik_key;
 } iplookupiterkey_t;
```

14.1.251 /usr/src/uts/common/io/1394/s1394_dev_disc.c

- Not all platforms default to `signed char` when `char` is declared. Ensure comparison with values less than 0 will work and not automatically evaluated as false.

```
--- a/usr/src/uts/common/io/1394/s1394_dev_disc.c    Tue Sep 02 11:22:56 2008 -
0400
+++ b/usr/src/uts/common/io/1394/s1394_dev_disc.c    Thu Sep 25 10:26:11 2008 -
0400
@@ -3381,7 +3381,7 @@
                                /*
                                 * all done with cur dir; pop it off the stack
                                 */
-                               if (node->dir_stack_top >= 0) {
+                               if ((signed char) node->dir_stack_top >= 0) {
                                        TNF_PROBE_3_DEBUG(
                                            s1394_calc_next_quad_exit,
                                            S1394_TNF_SL_HOTPLUG_STACK, "",
```

14.1.252 /usr/src/uts/common/io/avintr.c

- System z uses `xc_one` rather than `xc_call`.

```
--- a/usr/src/uts/common/io/avintr.c    Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/uts/common/io/avintr.c    Tue Sep 02 11:15:54 2008 -0400
@@ -638,7 +638,11 @@
                    return;
       }

+#ifndef __s390x
       xc_call(0, 0, 0, X_CALL_MEDPRI, poke, (xc_func_t)siron_poke_intr);
+#else
+      xc_one(poke, siron_poke_intr, 0, 0);
+#endif
 }

 /*
```

14.1.253 /usr/src/uts/common/io/gld.c

- Correct double-quote delimiting.

```
--- a/usr/src/uts/common/io/gld.c       Tue Sep 02 11:22:56 2008 -0400
+++ b/usr/src/uts/common/io/gld.c       Thu Sep 25 10:26:11 2008 -0400
@@ -2767,7 +2767,7 @@
         for (gld = vlan->gldv_str_next; gld != (gld_t *)&vlan->gldv_str_next;
             gld = gld->gld_next) {
 #ifdef GLD_VERBOSE_DEBUG
-               cmn_err(CE_NOTE, "gld_sendup: SAP: %4x QPTR: %p "QSTATE: %s",
+               cmn_err(CE_NOTE, "gld_sendup: SAP: %4x QPTR: %p QSTATE: %s",
                    gld->gld_sap, (void *)gld->gld_qptr,
                    gld->gld_state == DL_IDLE ? "IDLE": "NOT IDLE");
 #endif
```

14.1.254 /usr/src/uts/common/io/strplumb.c

- Move static function prototypes from within routine to global area: gcc doesn't cope with prototypes declared this way.

```
--- a/usr/src/uts/common/io/strplumb.c  Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/uts/common/io/strplumb.c  Tue Sep 02 11:15:54 2008 -0400
@@ -109,6 +109,8 @@
       &modlmisc,
       NULL
 };
+
+char *strplumb_get_netdev_path(void);

 int
 _init(void)
```

14.1.255 /usr/src/uts/common/nfs/nfs.h

- Include `sys/vfs_opreg.h` so that `struct fs_operation_def` is defined before it's used.

```
--- a/usr/src/uts/common/nfs/nfs.h      Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/uts/common/nfs/nfs.h      Tue Sep 02 11:15:54 2008 -0400
@@ -50,6 +50,7 @@
 #include <vm/page.h>
 #include <rpc/rpc_sztypes.h>
 #include <sys/sysmacros.h>
+#include <sys/vfs_opreg.h>

 #ifdef      __cplusplus
 extern "C" {
```

14.1.256 /usr/src/uts/common/nfs/nfs4.h

- Include `sys/vfs_opreg.h` so that `struct fs_operation_def` is defined before it's used.

```
--- a/usr/src/uts/common/nfs/nfs4.h     Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/uts/common/nfs/nfs4.h     Tue Sep 02 11:15:55 2008 -0400
@@ -42,6 +42,7 @@
```

```
 #endif
 #include <nfs/nfs4_attr.h>
 #include <sys/acl.h>
+#include <sys/vfs_opreg.h>

 #ifdef      __cplusplus
 extern "C" {
```

14.1.257 /usr/src/uts/common/os/brand.c

- Add support for System z

```
--- a/usr/src/uts/common/os/brand.c     Fri Aug 01 19:14:04 2008 -0700
+++ b/usr/src/uts/common/os/brand.c     Wed Aug 06 14:18:46 2008 -0400
@@ -45,7 +45,10 @@ struct brand_mach_ops native_mach_ops  =
 struct brand_mach_ops native_mach_ops  = {
                NULL, NULL
 };
-#else /* !__sparcv9 */
+#elif defined(__s390)
+struct brand_mach_ops native_mach_ops  = {
+               NULL
+#else /* !__s390 */
 struct brand_mach_ops native_mach_ops  = {
                NULL, NULL, NULL, NULL, NULL, NULL
 };

--- a/usr/src/uts/common/os/brand.c     Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/uts/common/os/brand.c     Tue Sep 02 11:15:55 2008 -0400
@@ -48,6 +48,7 @@
 #elif defined(__s390)
 struct brand_mach_ops native_mach_ops  = {
             NULL
+};
 #else /* !__s390 */
 struct brand_mach_ops native_mach_ops  = {
             NULL, NULL, NULL, NULL, NULL, NULL
```

14.1.258 /usr/src/uts/common/os/clock.c

- For System z `lbolt` and `lbolt64` aren't variables but a call to a function that reads the clock and converts it into a timer value.

```
--- a/usr/src/uts/common/os/clock.c     Fri Aug 01 19:14:04 2008 -0700
+++ b/usr/src/uts/common/os/clock.c     Wed Aug 06 14:18:46 2008 -0400
@@ -544,8 +544,10 @@ clock(void)
         * We rely on there being only one clock thread and hence
         * don't need a lock to protect lbolt.
         */
+#ifndef __s390
        lbolt++;
        atomic_add_64((uint64_t *)&lbolt64, (int64_t)1);
+#endif

        /*
         * Check for a callout that needs be called from the clock
@@ -1747,8 +1749,10 @@ deadman(void)
                * timeouts.  Note that we rely on deadman() being invoked once
                * per second, and credit lbolt and lbolt64 with hz ticks each.
                */
```

```
+#ifndef __s390
                lbolt += hz;
                lbolt64 += hz;
+#endif

                if (!deadman_panic_timers)
                        return; /* allow all timers to be manually disabled */-
-- a/usr/src/uts/common/os/clock.c      Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/uts/common/os/clock.c     Tue Sep 02 11:15:55 2008 -0400
@@ -238,9 +238,14 @@
 int32_t pps_errcnt = 0;          /* calibration errors */
 int32_t pps_stbcnt = 0;          /* stability limit exceeded */

+/*
+ * For s390x lbolt/lbolt64 are direct reads of the clock converted to Hz
+ */
+#ifndef __s390x
 /* The following variables require no explicit locking */
 volatile clock_t lbolt;          /* time in Hz since last boot */
 volatile int64_t lbolt64; /* lbolt64 won't wrap for 2.9 billion yrs */
+#endif

 kcondvar_t lbolt_cv;
 int one_sec = 1; /* turned on once every second */
```

14.1.259 /usr/src/uts/common/os/kmem.c

- Move static function prototypes from within routine to global area: gcc doesn't cope with prototypes declared this way.

```
--- a/usr/src/uts/common/os/kmem.c      Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/uts/common/os/kmem.c      Tue Sep 02 11:15:55 2008 -0400
@@ -1140,6 +1140,7 @@
        kmem_bufctl_t *kmp_bufctl;  /* bufctl */
 } kmem_panic_info;

+static void kmem_update(void *);

 static void
 copy_pattern(uint64_t pattern, void *buf_arg, size_t size)
@@ -2895,8 +2896,6 @@
 static void
 kmem_update_timeout(void *dummy)
 {
-       static void kmem_update(void *);
-
        (void) timeout(kmem_update, dummy, kmem_reap_interval);
 }
```

14.1.260 /usr/src/uts/common/os/kstat_fr.c

- Rename `lbolt` member of structure to lboltv so that it doesn't conflict with the `lbolt` timer value.

```
--- a/usr/src/uts/common/os/kstat_fr.c  Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/uts/common/os/kstat_fr.c  Tue Sep 02 11:15:55 2008 -0400
@@ -154,7 +154,7 @@

 struct {
```

```
        kstat_named_t ncpus;
-       kstat_named_t lbolt;
+       kstat_named_t lboltv;
        kstat_named_t deficit;
        kstat_named_t clk_intr;
        kstat_named_t vac;
@@ -842,7 +842,7 @@
        }

        system_misc_kstat.ncpus.value.ui32              = (uint32_t)myncpus;
-       system_misc_kstat.lbolt.value.ui32              = (uint32_t)lbolt;
+       system_misc_kstat.lboltv.value.ui32             = (uint32_t)lbolt;
        system_misc_kstat.deficit.value.ui32            = (uint32_t)deficit;
        system_misc_kstat.clk_intr.value.ui32           = (uint32_t)lbolt;
        system_misc_kstat.vac.value.ui32        = (uint32_t)vac;
```

14.1.261 /usr/src/uts/common/os/sunpm.c

- Move static function prototypes from within routine to global area: gcc doesn't cope with prototypes declared this way.

```
--- a/usr/src/uts/common/os/sunpm.c     Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/uts/common/os/sunpm.c     Tue Sep 02 11:15:55 2008 -0400
@@ -326,6 +326,20 @@
 static int pm_all_to_normal_nexus(dev_info_t *, pm_canblock_t);
 static void e_pm_set_max_power(dev_info_t *, int, int);
 static int e_pm_get_max_power(dev_info_t *, int);
+static int pm_reset_timestamps(dev_info_t *, void *);
+static void pm_dep_thread(void);
+static int pm_start(dev_info_t *dip);
+static int cur_threshold(dev_info_t *, int);
+static int pm_next_lower_power(pm_component_t *, int);
+static int pm_phc_impl(dev_info_t *, int, int, int);
+static void bring_pmdep_up(dev_info_t *, int);
+static void pm_enqueue_pscc(pscc_t *, pscc_t **);
+static psce_t *pm_psc_find_clone(int, pscc_t **, krwlock_t *);
+static major_t i_path_to_major(char *, char *);
+static void i_pm_driver_removed(major_t major);
+static void adjust_ancestors(char *, int);
+static int pm_is_noinvol_ancestor(pm_noinvol_t *);
+static void pm_noinvol_process_ancestors(char *);

 /*
  * Dependency Processing is done thru a seperate thread.
@@ -591,7 +605,6 @@
        _NOTE(ARGUNUSED(arg))
        static int auto_save;
        static pm_cpupm_t cpupm_save;
-       static int pm_reset_timestamps(dev_info_t *, void *);

        switch (code) {
        case CB_CODE_CPR_CHKPT:
@@ -668,7 +681,6 @@
        PMD_FUNC(pmf, "pm_init")
        char **mod;
        extern pri_t minclsyspri;
-       static void pm_dep_thread(void);

        pm_comps_notlowest = 0;
        pm_system_idle_threshold = pm_default_idle_threshold;
@@ -848,7 +860,6 @@
 e_pm_valid_info(dev_info_t *dip, pm_info_t **infop)
```

```
 {
       pm_info_t *info;
-      static int pm_start(dev_info_t *dip);

       /*
        * Check if the device is power managed if not.
@@ -1187,8 +1198,6 @@
       pm_component_t       *cp;
       dev_info_t   *pdip = ddi_get_parent(dip);
       int          circ;
-      static int   cur_threshold(dev_info_t *, int);
-      static int   pm_next_lower_power(pm_component_t *, int);
       clock_t             min_scan = pm_default_min_scan;

       /*
@@ -2461,7 +2470,6 @@
       struct pm_component *cp = PM_CP(dip, comp);
       int retval;
       pm_info_t *info = PM_GET_PM_INFO(dip);
-      static int pm_phc_impl(dev_info_t *, int, int, int);

       PMD(PMD_KIDSUP, ("%s: %s@%s(%s#%d), comp=%d, level=%d\n", pmf,
           PM_DEVICE(dip), comp, level))
@@ -2928,7 +2936,6 @@
       dev_info_t *pdip = ddi_get_parent(dip);
       struct pm_component *cp;
       int blocked, circ, pcirc, old_level;
-      static int pm_phc_impl(dev_info_t *, int, int, int);

       if (level < 0) {
             PMD(PMD_FAIL, ("%s: %s@%s(%s#%d): bad level=%d\n", pmf,
@@ -3758,7 +3765,6 @@
       pm_info_t *wku_info;
       char *kept_path;
       dev_info_t *kept;
-      static void bring_pmdep_up(dev_info_t *, int);

       if (panicstr) {
             return;
@@ -5563,7 +5569,6 @@
 {
       pscc_t *p;
       psce_t *psce;
-      static void pm_enqueue_pscc(pscc_t *, pscc_t **);

       /*
        * We definitely need a control struct, then we have to search to see
@@ -5781,7 +5786,6 @@
 psce_t *
 pm_psc_clone_to_direct(int clone)
 {
-      static psce_t *pm_psc_find_clone(int, pscc_t **, krwlock_t *);
       return (pm_psc_find_clone(clone, &pm_pscc_direct,
           &pm_pscc_direct_rwlock));
 }
@@ -5792,7 +5796,6 @@
 psce_t *
 pm_psc_clone_to_interest(int clone)
 {
-      static psce_t *pm_psc_find_clone(int, pscc_t **, krwlock_t *);
       return (pm_psc_find_clone(clone, &pm_pscc_interest,
           &pm_pscc_interest_rwlock));
 }
```

```
@@ -7861,7 +7864,6 @@
       char *np, *ap, *bp;
       major_t ret;
       size_t len;
-      static major_t i_path_to_major(char *, char *);

       PMD(PMD_NOINVOL, ("%s: %s\n", pmf, path))

@@ -8091,7 +8093,6 @@
 void
 pm_driver_removed(major_t major)
 {
-      static void i_pm_driver_removed(major_t major);

       /*
        * Serialize removal of drivers. This is to keep ancestors of
@@ -8110,9 +8111,6 @@
 i_pm_driver_removed(major_t major)
 {
       PMD_FUNC(pmf, "driver_removed")
-      static void adjust_ancestors(char *, int);
-      static int pm_is_noinvol_ancestor(pm_noinvol_t *);
-      static void pm_noinvol_process_ancestors(char *);
       pm_noinvol_t *ip, *pp = NULL;
       int wasvolpmd;
       ASSERT(major != DDI_MAJOR_T_NONE);
```

14.1.262 /usr/src/uts/common/os/vers.c

- Add some build date/time information that we can access in the kernel.

```
--- a/usr/src/uts/common/os/vers.c      Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/uts/common/os/vers.c      Tue Sep 02 11:15:55 2008 -0400
@@ -36,3 +36,5 @@
 struct utsname utsname = {
       "SunOS", "", UTS_RELEASE, UTS_VERSION, UTS_PLATFORM
 };
+
+const char *version_data = __DATE__ " " __TIME__;
```

14.1.263 /usr/src/uts/common/rpc/rpcmod.c

- Move static function prototypes from within routine to global area: gcc doesn't cope with prototypes declared this way.

```
--- a/usr/src/uts/common/rpc/rpcmod.c   Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/uts/common/rpc/rpcmod.c   Tue Sep 02 11:15:55 2008 -0400
@@ -220,6 +220,8 @@
 static      void   mir_wput(queue_t *q, mblk_t *mp);
 static      void   mir_wsrv(queue_t *q);

+static void rpcmod_release(queue_t *, mblk_t *);
+
 static struct module_info rpcmod_info =
       {RPCMOD_ID, "rpcmod", 0, INFPSZ, 256*1024, 1024};

@@ -570,7 +572,6 @@
       struct rpcm *rmp;

       extern void (*rpc_rele)(queue_t *, mblk_t *);
```

```
-       static void rpcmod_release(queue_t *, mblk_t *);

        TRACE_0(TR_FAC_KRPC, TR_RPCMODOPEN_START, "rpcmodopen_start:");
```

14.1.264 /usr/src/uts/common/sys/Makefile.syshdrs

- Add System z as a supported platform.

```
--- a/usr/src/uts/common/sys/Makefile.syshdrs   Fri Aug 01 19:14:04 2008 -0700
+++ b/usr/src/uts/common/sys/Makefile.syshdrs   Wed Aug 06 14:18:46 2008 -0400
@@ -149,6 +149,8 @@ sparc_ROOTDIRS=     $(ROOTDKTPDIR) $(ROOTDIR

 i386_ROOTDIRS= $(ROOTDKTPDIR) $(ROOTDIR)/scsi/adapters $(ROOTDIR)/scsi/targets
\
                $(ROOTDIR)/agp $(ROOTDIR)/sata
+
+s390x_ROOTDIRS=        $(ROOTDKTPDIR) $(ROOTDIR)/scsi/adapters
$(ROOTDIR)/scsi/targets

 ROOTDIRS=                      \
        $(ROOTDIR)              \
@@ -279,6 +281,11 @@ i386_ROOTHDRS=     $(ROOTDKTPHDRS) $(ROOTPCH
                $(ROOTPCMCIAHDRS) $(ROOTHOTPLUGHDRS) \
                $(ROOTHOTPLUGPCIHDRS) $(ROOTSATAGENHDRS)

+s390x_ROOTHDRS= $(ROOTDKTPHDRS) $(ROOTPCHDRS) $(ROOTSCSITARGETSHDRS)    \
+               $(ROOTSCSIVHCIHDRS) $(ROOTFCHDRS)                        \
+               $(ROOTPCMCIAHDRS) $(ROOTHOTPLUGHDRS)                     \
+               $(ROOTHOTPLUGPCIHDRS)
+
 # install rules
 $(ROOTDIR)/%: %
        $(INS.file)
--- a/usr/src/uts/common/sys/Makefile.syshdrs  Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/uts/common/sys/Makefile.syshdrs  Tue Sep 02 11:15:55 2008 -0400
@@ -150,7 +150,7 @@
 i386_ROOTDIRS=     $(ROOTDKTPDIR) $(ROOTDIR)/scsi/adapters
$(ROOTDIR)/scsi/targets \
             $(ROOTDIR)/agp $(ROOTDIR)/sata

-s390x_ROOTDIRS=    $(ROOTDKTPDIR) $(ROOTDIR)/scsi/adapters
$(ROOTDIR)/scsi/targets
+s390_ROOTDIRS=     $(ROOTDKTPDIR) $(ROOTDIR)/scsi/adapters
$(ROOTDIR)/scsi/targets

 ROOTDIRS=                 \
       $(ROOTDIR)          \
@@ -176,9 +176,11 @@
       $(ROOTDIR)/lvm              \
       $(ROOTDIR)/pcmcia   \
       $(ROOTDIR)/scsi             \
+      $(ROOTDIR)/scsi/adapters    \
       $(ROOTDIR)/scsi/conf        \
       $(ROOTDIR)/scsi/generic     \
       $(ROOTDIR)/scsi/impl        \
+      $(ROOTDIR)/scsi/targets \
       $(ROOTDIR)/fc4              \
       $(ROOTDIR)/sysevent     \
       $(ROOTDIR)/contract     \
@@ -281,7 +283,7 @@
             $(ROOTPCMCIAHDRS) $(ROOTHOTPLUGHDRS) \
```

```
              $(ROOTHOTPLUGPCIHDRS) $(ROOTSATAGENHDRS)

-s390x_ROOTHDRS= $(ROOTDKTPHDRS) $(ROOTPCHDRS) $(ROOTSCSITARGETSHDRS)     \
+s390_ROOTHDRS=  $(ROOTSDKTPHDRS) $(ROOTPCHDRS) $(ROOTSCSITARGETSHDRS)    \
              $(ROOTSCSIVHCIHDRS) $(ROOTFCHDRS)              \
              $(ROOTPCMCIAHDRS) $(ROOTHOTPLUGHDRS)                  \
              $(ROOTHOTPLUGPCIHDRS)
```

14.1.265 /usr/src/uts/common/sys/exec.h

- Move `execsw[]` to a pointer after `struct execsw` is defined.

```
--- a/usr/src/uts/common/sys/exec.h     Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/uts/common/sys/exec.h     Tue Sep 02 11:15:55 2008 -0400
@@ -71,7 +71,6 @@
 #define      LOADED_EXEC(e)             ((e)->exec_func)

 extern int nexectype;            /* number of elements in execsw */
-extern struct execsw execsw[];
 extern kmutex_t execsw_lock;

 /*
@@ -186,6 +185,7 @@

 extern short elfmagic;
 extern short intpmagic;
+extern struct execsw execsw[];
 extern short javamagic;
 #if defined(__sparc)
 extern short aout_zmagic;
```

14.1.266 /usr/src/uts/common/sys/fs/autofs.h

- Include `sys/vfs_opreg.h` so that `struct fs_operation_def` is defined before it's used.

```
--- a/usr/src/uts/common/sys/fs/autofs.h       Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/uts/common/sys/fs/autofs.h       Tue Sep 02 11:15:55 2008 -0400
@@ -42,6 +42,7 @@
 #include <sys/zone.h>
 #include <sys/door.h>
 #include <rpcsvc/autofs_prot.h>
+#include <sys/vfs_opreg.h>

 #ifdef      __cplusplus
 extern "C" {
```

14.1.267 /usr/src/uts/common/sys/fs/dv_node.h

- Include `sys/vfs_opreg.h` so that `struct fs_operation_def` is defined before it's used.

```
--- a/usr/src/uts/common/sys/fs/dv_node.h      Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/uts/common/sys/fs/dv_node.h      Tue Sep 02 11:15:55 2008 -0400
@@ -45,6 +45,7 @@
 #include <sys/fs/sdev_node.h>
 #include <sys/devpolicy.h>
 #include <sys/avl.h>
+#include <sys/vfs_opreg.h>
```

```
 #ifdef __cplusplus
 extern "C" {
```

14.1.268 /usr/src/uts/common/sys/fs/fifonode.h

- Include `sys/vfs_opreg.h` so that `struct fs_operation_def` is defined before it's used.

```
--- a/usr/src/uts/common/sys/fs/fifonode.h     Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/uts/common/sys/fs/fifonode.h     Tue Sep 02 11:15:55 2008 -0400
@@ -36,6 +36,7 @@
 extern "C" {
 #endif

+#include <sys/vfs_opreg.h>

 /*
  * Each FIFOFS object is identified by a struct fifonode/vnode pair.
```

14.1.269 /usr/src/uts/common/sys/fs/hsfs_impl.h

- Include `sys/vfs_opreg.h` so that `struct fs_operation_def` is defined before it's used.

```
--- a/usr/src/uts/common/sys/fs/hsfs_impl.h    Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/uts/common/sys/fs/hsfs_impl.h    Tue Sep 02 11:15:55 2008 -0400
@@ -34,6 +34,8 @@
 #ifdef      __cplusplus
 extern "C" {
 #endif
+
+#include <sys/vfs_opreg.h>

 /*
  * global routines.
```

14.1.270 /usr/src/uts/common/sys/fs/lofs_info.h

- Include `sys/vfs_opreg.h` so that `struct fs_operation_def` is defined before it's used.

```
--- a/usr/src/uts/common/sys/fs/lofs_info.h    Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/uts/common/sys/fs/lofs_info.h    Tue Sep 02 11:15:55 2008 -0400
@@ -35,6 +35,8 @@
 #ifdef      __cplusplus
 extern "C" {
 #endif
+
+#include <sys/vfs_opreg.h>

 struct lnode;
```

14.1.271 /usr/src/uts/common/sys/fs/namenode.h

- Include `sys/vfs_opreg.h` so that `struct fs_operation_def` is defined before it's used.

```
--- a/usr/src/uts/common/sys/fs/namenode.h      Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/uts/common/sys/fs/namenode.h      Tue Sep 02 11:15:55 2008 -0400
@@ -35,6 +35,8 @@
 #if defined(_KERNEL)
 #include <sys/vnode.h>
 #endif
+
+#include <sys/vfs_opreg.h>

 #ifdef      __cplusplus
 extern "C" {
```

14.1.272 /usr/src/uts/common/sys/fs/pc_node.h

- Include `sys/vfs_opreg.h` so that `struct fs_operation_def` is defined before it's used.

```
--- a/usr/src/uts/common/sys/fs/pc_node.h       Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/uts/common/sys/fs/pc_node.h       Tue Sep 02 11:15:55 2008 -0400
@@ -35,6 +35,7 @@
 #include <vm/page.h>
 #include <sys/buf.h>
 #include <sys/vnode.h>
+#include <sys/vfs_opreg.h>

 /*
  * This overlays the fid structure (see vfs.h)
```

14.1.273 /usr/src/uts/common/sys/fs/tmpnode.h

- Include `sys/vfs_opreg.h` so that `struct fs_operation_def` is defined before it's used.

```
--- a/usr/src/uts/common/sys/fs/tmpnode.h       Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/uts/common/sys/fs/tmpnode.h       Tue Sep 02 11:15:55 2008 -0400
@@ -32,6 +32,7 @@
 #include <sys/t_lock.h>
 #include <vm/seg.h>
 #include <vm/seg_vn.h>
+#include <sys/vfs_opreg.h>

 #ifdef      __cplusplus
 extern "C" {
```

14.1.274 /usr/src/uts/common/sys/fs/ufs_inode.h

- Include `sys/vfs_opreg.h` so that `struct fs_operation_def` is defined before it's used.

```
--- a/usr/src/uts/common/sys/fs/ufs_inode.h     Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/uts/common/sys/fs/ufs_inode.h     Tue Sep 02 11:15:55 2008 -0400
@@ -58,6 +58,7 @@
```

```
 #include <sys/fs/ufs_acl.h>
 #include <sys/fs/ufs_panic.h>
 #include <sys/dnlc.h>
+#include <sys/vfs_opreg.h>

 #ifdef      __cplusplus
 extern "C" {
```

14.1.275 /usr/src/uts/common/sys/mode.h

- Include `sys/stat.h` and `sys/vnode.h` so that structures and typedefs are defined before they are used.

```
--- a/usr/src/uts/common/sys/mode.h     Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/uts/common/sys/mode.h     Tue Sep 02 11:15:55 2008 -0400
@@ -41,6 +41,8 @@
  * REQUIRES sys/stat.h
  * REQUIRES sys/vnode.h
  */
+#include <sys/stat.h>
+#include <sys/vnode.h>

 /*
  * Conversion between vnode types/modes and encoded type/mode as
```

14.1.276 /usr/src/uts/common/uts/prsystm.h

- Get definition of external function of `prgetprfpgregs32()` to match what was defined elsewhere.

```
--- a/usr/src/uts/common/sys/prsystm.h  Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/uts/common/sys/prsystm.h  Tue Sep 02 11:15:55 2008 -0400
@@ -34,6 +34,7 @@
 #pragma ident      "%Z%%M%      %I%    %E% SMI"

 #include <sys/isa_defs.h>
+#include <sys/procfs_isa.h>
 #include <sys/zone.h>

 #ifdef      __cplusplus
@@ -121,7 +122,7 @@
 #endif      /* __x86 */

 #ifdef _SYSCALL32_IMPL
-struct prfpregset32;
+// struct prfpregset32;
 struct pstatus32;
 struct lwpstatus32;
 struct psinfo32;
@@ -130,7 +131,7 @@
 extern void prgetlwpstatus32(kthread_t *, struct lwpstatus32 *, zone_t *);
 extern void prgetpsinfo32(proc_t *, struct psinfo32 *);
 extern void prgetlwpsinfo32(kthread_t *, struct lwpsinfo32 *);
-extern void prgetprfpregs32(klwp_t *, struct prfpregset32 *);
+extern void prgetprfpregs32(klwp_t *, prfpregset32_t *);
 #if defined(__sparc)
 struct gwindows32;
 void        prgetwindows32(klwp_t *, struct gwindows32 *);
```

14.1.277 /usr/src/uts/common/sys/socketvar.h

- Include `sys/vfs_opreg.h` so that `struct fs_operation_def` is defined before it's used.

```
--- a/usr/src/uts/common/sys/socketvar.h       Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/uts/common/sys/socketvar.h       Tue Sep 02 11:15:55 2008 -0400
@@ -52,6 +52,7 @@
 #include <sys/zone.h>
 #include <sys/sodirect.h>
 #include <inet/kssl/ksslapi.h>
+#include <sys/vfs_opreg.h>

 #ifdef       __cplusplus
 extern "C" {
```

14.1.278 /usr/src/uts/common/sys/syscall.h

- Add `cpcmd` syscall for System z

```
--- a/usr/src/uts/common/sys/syscall.h  Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/uts/common/sys/syscall.h  Tue Sep 02 11:15:56 2008 -0400
@@ -447,6 +447,7 @@
 #define      SYS_schedctl         206
 #define      SYS_pset             207
 #define      SYS_sparc_utrap_install    208
+#define SYS_cpcmd         208
 #define      SYS_resolvepath            209
 #define      SYS_lwp_mutex_timedlock    210
 #define      SYS_lwp_sema_timedwait     211
```

14.1.279 /usr/src/uts/common/sys/systm.h

- For System z `lbolt` and `lbolt64` aren't variables but a call to a function that reads the clock and converts it into a timer value.

```
--- a/usr/src/uts/common/sys/systm.h    Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/uts/common/sys/systm.h    Tue Sep 02 11:15:56 2008 -0400
@@ -67,8 +67,9 @@
 extern volatile clock_t lbolt;   /* time in HZ since last boot */
 extern volatile int64_t lbolt64; /* lbolt computed as 64-bit value */
 #else
-# define lbolt            tod2ticks()
-# define lbolt64   tod2ticks()
+extern hrtime_t tod2ticks();
+# define lbolt            ((clock_t) tod2ticks())
+# define lbolt64   ((int64_t) tod2ticks())
 #endif
 extern int interrupts_unleashed; /* set after the spl0() in main() */
```

14.1.280 /usr/src/uts/common/sys/user.h

- Add fields to `user_t` that are used by System z.

```
--- a/usr/src/uts/common/sys/user.h     Fri Aug 01 19:14:04 2008 -0700
+++ b/usr/src/uts/common/sys/user.h     Wed Aug 06 14:18:46 2008 -0400
```

```
@@ -187,6 +187,8 @@ typedef struct {            /* kernel syscall set
 #define         __KERN_NAUXV_IMPL 19
 #elif defined(__i386) || defined(__amd64)
 #define         __KERN_NAUXV_IMPL 21
+#elif defined (__s390)
+#define __KERN_NAUXV_IMPL 21
 #endif

 struct execsw;
@@ -285,6 +287,9 @@ typedef     struct  user {
        void    (*u_signal[MAXSIG])();  /* Disposition of signals */
        int             u_code;         /* fault code on trap */
        caddr_t         u_addr;         /* fault PC on trap */
+#ifdef __s390
+       psgw_t          u_psw;          /* user's saved PSW */
+#endif
 } user_t;


--- a/usr/src/uts/common/sys/user.h     Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/uts/common/sys/user.h     Tue Sep 02 11:15:56 2008 -0400
@@ -288,7 +288,7 @@
       int           u_code;             /* fault code on trap */
       caddr_t             u_addr;             /* fault PC on trap */
 #ifdef __s390
-      psgw_t        u_psw;        /* user's saved PSW */
+      pswg_t        u_psw;        /* user's saved PSW */
 #endif
 } user_t;
```

14.1.281 /usr/src/uts/common/sys/va_impl.h

- Add support for gcc version 4.

```
--- a/usr/src/uts/common/sys/va_impl.h  Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/uts/common/sys/va_impl.h  Tue Sep 02 11:15:56 2008 -0400
@@ -147,13 +147,20 @@
 #define     __va_copy(to, from) __va_void(((to) = (from)))
 #define     __va_end(list)              __va_void(0)

-#elif defined(__GNUC__)   && ((__GNUC__ == 2 && __GNUC_MINOR >= 96) || \
-      (__GNUC__ >= 3))           /* ------------------------ protocol */
+#elif defined(__GNUC__) && ((__GNUC__ == 2 && __GNUC_MINOR >= 96) || \
+        (__GNUC__ >= 3) && __GNUC__ < 4) /* ------------------------ protocol
*/
+
+#define __va_start(list, name)   __builtin_stdarg_start(list, name)
+#define __va_arg(list, type)     __builtin_va_arg(list, type)
+#define __va_end(list)           __builtin_va_end(list)
+#define __va_copy(to, from)      __builtin_va_copy(to, from)

-#define     __va_start(list, name)     __builtin_stdarg_start(list, name)
-#define     __va_arg(list, type)       __builtin_va_arg(list, type)
-#define     __va_end(list)             __builtin_va_end(list)
-#define     __va_copy(to, from) __builtin_va_copy(to, from)
+#elif defined(__GNUC__) && (__GNUC__ >= 4)
+
+#define __va_start(list, name)   __builtin_va_start(list, name)
+#define __va_arg(list, type)     __builtin_va_arg(list, type)
+#define __va_end(list)           __builtin_va_end(list)
+#define __va_copy(to, from)      __builtin_va_copy(to, from)
```

```
 #else                          /* ---------------------- protocol */
```

14.1.282 /usr/src/uts/common/syscall/lwp_sobj.c

- Move static function prototypes from within routine to global area: gcc doesn't cope with prototypes declared this way.

```
--- a/usr/src/uts/common/syscall/lwp_sobj.c    Tue Aug 19 11:19:58 2008 -0400
+++ b/usr/src/uts/common/syscall/lwp_sobj.c    Tue Sep 02 11:15:56 2008 -0400
@@ -68,6 +68,7 @@
 static void lwp_unsleep(kthread_t *t);
 static void lwp_change_pri(kthread_t *t, pri_t pri, pri_t *t_prip);
 static void lwp_mutex_cleanup(lwpchan_entry_t *ent, uint16_t lockflg);
+static int iswanted();

 extern int lwp_cond_signal(lwp_cond_t *cv);

@@ -1133,7 +1134,6 @@
       volatile uint8_t type = 0;
       lwpchan_t lwpchan;
       sleepq_head_t *sqh;
-      static int iswanted();
       uint16_t flag;
       int imm_timeout = 0;
```

14.1.283 /usr/src/uts/modDepends.ksh

- Script to create *modname*`_depends.s` files containing dependency information usually provided by the `-N` flag of the Solaris linker.

```
--- /dev/null Thu Jan 01 00:00:00 1970 +0000
+++ b/usr/src/uts/modDepends.ksh Tue Sep 02 11:15:56 2008 -0400
@@ -0,0 +1,29 @@
+#!/bin/ksh
+found=0
+while getopts N:d:r name
+do
+      case $name in
+      N)       if [ found -eq 1 ]
+             then
+                    depend="$depend $OPTARG"
+             else
+                    depend="$OPTARG"
+                    found=1
+             fi
+             ;;
+      *)     true
+             ;;
+      esac
+done
+shift $(($OPTIND-1))
+if [ $# -eq 1 ]
+then
+      echo "\t.align 4" >$1
+      echo "\t.global _depends_on" >>$1
+      echo "_depends_on:" >>$1
+      echo "\t.asciz \"$depend\"" >>$1
```

```
+       echo "\t.size _depends_on, .-_depends_on" >>$1
+       exit 0
+else
+       exit 1
+fi
```

14.2 Kludges

14.2.1 Various

- Manually create symlink for .so to .so.1 after install of certain libraries
- Manually added demangle.h

14.2.2 $SRC/cmd/agents/snmp

- mkdir –p $ROOT/var/snmp/mib

15. System z Source Files

This section contains description of new files added to implement OpenSolaris on System z. Only 'C', header, and assembler files are listed. There are hundreds of Makefiles in the various trees that are not described here.

15.1 Kernel

These files live in the `uts` directory under the s390x and zSeries subdirectories.

15.1.1 usr/src/uts/s390x/conf/confunix.c

Definitions for rootfs and swap.

15.1.2 usr/src/uts/s390x/cpu/cpu_module.c

Various clock related routines and a utility routine from the VAX days (scancc).

todhrestime	
flush_instr_mem	
syncfpu	
getthrestime	
gethrestime_sec	
gethrestime_lasttick	
gethrtime	
gethrtime_unscaled	
gethrtime_waitfree	
dtrace_gethrtime	
get_hrestime	
scalehrtime	
hres_tick	
drv_usecwait	
cpu_init_private	
scancc	

15.1.3 usr/src/uts/s390x/io/cbe.c

Platform-specific cyclic timer implementations for System z.

15.1.4 usr/src/uts/s390x/io/ccw.c

CCW common I/O framework.

15.1.5 usr/src/uts/s390x/io/ccwnex.c

CCW bus nexus driver.

15.1.6 usr/src/uts/s390x/io/cpunex.c

CPU Nexus driver.

15.1.7 usr/src/uts/s390x/io/hardclk.c

Time of day (TOD) clock support functions.

15.1.8 usr/src/uts/s390x/io/rootnex.c

rootnexus driver for s390x.

15.1.9 usr/src/uts/s390x/kipl/kipl.s

Low-core definitions and boot entry point.

15.1.10 usr/src/uts/s390x/kipl/kipl_kmem.c

Free storage handler for boot.

15.1.11 usr/src/uts/s390x/kipl/kipl_mem.c

Provide very basic alloc() and free() support during the boot stage.

15.1.12 usr/src/uts/s390x/kipl/kipl_mem.h

Header file for boot memory management.

15.1.13 usr/src/uts/s390x/kipl/kipl_prop.c

Routines to handle propositions.

15.1.14 usr/src/uts/s390x/kipl/kipl_setup.c

OpenSolaris on System z bootstrapper.

15.1.15 usr/src/uts/s390x/ml/ddi_asm.s

DDI layer get and put routines.

15.1.16 usr/src/uts/s390x/ml/genconst.c

Utility routines for creating assembler `#define` statements from C structures.

15.1.17 usr/src/uts/s390x/ml/interrupt.s

First level interrupt handler for SVC, program, I/O, external, machine check and restart interruptions.

15.1.18 usr/src/uts/s390x/ml/lock_prim.s

Lock primitives (e.g. mutex_enter, mutex_exit).

15.1.19 usr/src/uts/s390x/ml/mach_copy.s

Cross-address space copy operations.

15.1.20 usr/src/uts/s390x/ml/mach_locore.s

Post-bootstrap low-core defintions and control register setup.

15.1.21 usr/src/uts/s390x/ml/mach_subr_asm.s

Various LPW/RTT subroutines.

15.1.22 usr/src/uts/s390x/ml/s390x_subr.s

spl8	
spl7	
splzs	
splhi	
splhigh	
spl6	
i_ddi_splhigh	
splsclp	
spl0	
splx	

i_ddi_splx	
splr	
setspl	
on_fault	
no_fault	
on_trap_trampoline	
on_trap	
setjmp	
longjmp	
getfp	
getpil	
setpil	
_insque	
_remque	
cpu_wait	
caller	
scanc	
ftrace_interrupt_disable	
ftrace_interrupt_enable	
strlen	
membar_ldld	
membar_stld	
membar_ldst	
membar_stst	
membar_ldld_stld	
membar_stld_ldld	
membar_ldld_ldst	
membar_ldst_ldld	
membar_ldld_stst	
membar_stst_ldld	
membar_ldst_stld	

membar_stld_ldst	
membar_stld_stst	
membar_stst_stld	
membar_ldst_stst	
membar_stst_ldst	
membar_lookaside	
membar_memissue	
membar_sync	
dtrace_interrupt_disable	
dtrace_interrupt_enable	
dtrace_membar_return	
dtrace_membar_producer	
dtrace_membar_consumer	
panic_trigger	
dtrace_panic_trigger	
vpanic	
dtrace_vpanic	
fuword8_noerr	
fuword16_noerr	
fuword32_noerr	
fuword64_noerr	
suword8_noerr	
suword16_noerr	
suword32_noerr	
suword64_noerr	
prefetch_smap_w	
prefetch_page_r	
kmdb_enter	
sigsoftint	
sti	
cli	

intr_restore	
intr_clear	
intr_enable	
switch_sp_and_call	
frogr	
flogr	
flogr	
atomic_btr32	
threadp	
ticks2tod	
nano2tod	
tod2nano	
tod2ticks	
gettick_counter	

15.1.23 usr/src/uts/s390x/ml/swtch.s

Thread switching and resumtion subroutines.

15.1.24 usr/src/uts/s390x/ml/syscall_trap.s

Syscall processor.

15.1.25 usr/src/uts/s390x/ml/thunk.s

32-bit syscall to 64-bit syscall parameter thunker.

15.1.26 usr/src/uts/s390x/os/bitmap_arch.c

Find highest/lowest bit set.

15.1.27 usr/src/uts/s390x/os/cpc_subr.c

Platform specific CPU counter subroutines.

15.1.28 usr/src/uts/s390x/os/ddi_impl.c

Architecture specific DDI implementation support routines.

15.1.29 usr/src/uts/s390x/os/dtrace_subr.c

DTrace support functions.

15.1.30 usr/src/uts/s390x/os/exts390x.c

Handle external interrupts for system.

15.1.31 usr/src/uts/s390x/os/intr.c

Process interrupts and dispatch threads to handle the work request.

15.1.32 usr/src/uts/s390x/os/ioinit.c

Scan the system for I/O devices and create a double linked list of structures representing these devices. Get device characteristics of these devices.

15.1.33 usr/src/uts/s390x/os/lgrpplat.c

Platform-specific support for lgroups common to s390x based platforms.

15.1.34 usr/src/uts/s390x/os/mach_cpu_states.c

mdboot	
mdpreboot	
reboot_machine	
panic_idle	
panic_stopcpus	
panic_enter_hw	
panic_quiesce_hw	
power_down	
panic_dump_hw	
cpu_faulted_enter	
cpu_faulted_exit	
mach_dump_buffer_init	
clear_watchdog_on_exit	
kdi_watchdog_disable	
kdi_watchdog_restore	

15.1.35 usr/src/uts/s390x/os/mach_ddi_impl.c

Platform-specific DDI implementation subroutines.

15.1.36 usr/src/uts/s390x/os/mach_mp_startup.c

SMP startup support routines.

15.1.37 usr/src/uts/s390x/os/mach_mp_states.c

Various minor SMP related utility functions.

set_idle_cpu	
unset_idle_cpu	
mp_cpu_poweron	
mp_cpu_poweroff	

15.1.38 usr/src/uts/s390x/os/mach_startup.c

setup_trap_table	No-operation.
mach_cpu_halt_idle	Load a hardwait PSW.
phys_installed_has_changed	No-operation.

15.1.39 usr/src/uts/s390x/os/mach_sysconfig.c

Machine specific system configuration routines.

15.1.40 usr/src/uts/s390x/os/mach_trap.c

System trap support routines.

15.1.41 usr/src/uts/s390x/os/machdep.c

tick2ns	
thread_stk_init	
lwp_stk_init	
lwp_stk_fini	
lwp_forkregs	
lwp_freeregs	
lwp_attach_brand_hdlrs	
lwp_detach_brand_hdlrs	
blacklist	

kdi_pread	
kdi_pwrite	
kdi_kernpanic	
kdi_plat_call	
mach_kdi_init	
kdi_flush_caches	
get_cpu_mstate	
pg_plat_hw_shared	
pg_plat_hw_instance_id	
pg_plat_cmt_load_bal_hw	
pg_plat_cmt_affinity_hw	
pg_plat_get_core_id	
pg_plat_cpus_share	
pg_plat_hw_level	
cmp_set_nosteal_interval	
mach_cpu_pause	
plat_mem_valid_page	
dump_plat_addr	
dump_plat_pfn	
dump_plat_data	
plat_hold_page	
plat_release_page	
abort_sequence_enter	
console_enter	
console_exit	
halt	
debug_enter	
mach_cpu_start	
mach_cpu_stop	
add_cpu2devnodetree	
mach_set_softintr	

cpu_wakeup	
wake_others	
dumpThread	
dumpCPU	
dumpSys	
plat_mem_do_mmio	

15.1.42 usr/src/uts/s390x/os/mch_slih.c

Rudimentary machine check handling.

15.1.43 usr/src/uts/s390x/os/mem_config_arch.c

Architecture-specific physical memory configuration routines.

15.1.44 usr/src/uts/s390x/os/memlist.c

Memory list management routines.

15.1.45 usr/src/uts/s390x/os/memnode.c

mem_node_add_slice	
mem_node_pre_del_slice	
mem_node_post_del_slice	
startup_build_mem_nodes	
mem_node_alloc	
mem_node_memlist_pages	

15.1.46 usr/src/uts/s390x/os/memscrub.c

To detect ECC errors etc early a periodic accessing of all memory is undertaken by other architectures. For s390x we just disable this.

15.1.47 usr/src/uts/s390x/os/mlsetup.c

mlsetup	Setup routine called right before main(). Interposing this function before main() allows us to call it in a machine-indepedent fashion.
mach_modpath	Construct the directy path from the filename.
bop_compinfo	Fake information about a compressed image.

15.1.48 usr/src/uts/s390x/os/mp_startup.c

Start up other CPUs in this complex.

15.1.49 usr/src/uts/s390x/os/pgm_slih.c

Program interrupt second level handler.

15.1.50 usr/src/uts/s390x/os/platmod.c

Platform-specific module information (empty for System z).

15.1.51 usr/src/uts/s390x/os/ppage.c

rcopy	
rzero	
ppmapin	
ppmapout	
ppcopy	
pagezero	

15.1.52 usr/src/uts/s390x/os/s390xdep.c

setfpregs	
getfpregs	
setgregs	
setaregs	
getgregs	
getaregs	
setgregs32	
getgregs32	
getpsw	
setpc	
getuserpc	
setregs	
sendsig	
sendsig32	

lwp_load	
lwp_setrval	
lwp_setsp	
lwp_pcb_exit	
sync_icache	
panic_saveregs	
panic_savetrap	
panic_showtrap	

15.1.53 usr/src/uts/s390x/os/sclp.c

Handle communications with the service processor.

15.1.54 usr/src/uts/s390x/os/smp.c

Field interrupts pertaining to SMP facilities.

15.1.55 usr/src/uts/s390x/os/startup.c

Commence the startup of the Solaris OS on the System z hardware

15.1.56 usr/src/uts/s390x/os/sundep.c

Various startup support routines.

check_boot_version	
kern_setup1	
thread_load	
lwp_getdatamodel	

15.1.57 usr/src/uts/s390x/os/timer_s390x.c

Handle basic timer functions of initializing timers and setting interrupt vectors for them.

15.1.58 usr/src/uts/s390x/os/trap.c

Various trap and pre-emption related routines.

interrupts_enabled	
kpreempt	
trap_cleanup	

trap_rtt	

15.1.59 usr/src/uts/s390x/os/wdt.c

Watchdog timer support – disabled on System z.

15.1.60 usr/src/uts/s390x/os/x_call.c

Cross-system call functions.

xc_init	
xc_one	
xc_some	
xc_all	
xc_serv	
xc_trace	

15.1.61 usr/src/uts/s390x/sys/clock.h

Architecture specific cyclic timer definitions.

15.1.62 usr/src/uts/s390x/sys/ddi_subrdefs.h

DDI platform implementation subroutines definitions.

15.1.63 usr/src/uts/s390x/sys/intr.h

Architecture specific interrupt definitions.

15.1.64 usr/src/uts/s390x/sys/intreg.h

Soft-interrupt related definitions.

15.1.65 usr/src/uts/s390x/sys/mach_intr.h

Platform-dependent interrupt data structures.

15.1.66 usr/src/uts/s390x/sys/machbrand.h

Brand related definitions.

15.1.67 usr/src/uts/s390x/sys/machclock.h

TOD clock related definitions.

15.1.68 usr/src/uts/s390x/sys/machcpuvar.h

Machine specific fields of the cpu structure defined in the common code.

15.1.69 usr/src/uts/s390x/sys/machparam.h

Machine dependent parameters and limits.

15.1.70 usr/src/uts/s390x/sys/machpcb.h

Machine dependent per-thread data.

15.1.71 usr/src/uts/s390x/sys/machsystm.h

Numerous platform-dependent interfaces that don't seem to belongin any other header file.

15.1.72 usr/src/uts/s390x/sys/machthread.h

Platform dependent thread support definitions.

15.1.73 usr/src/uts/s390x/sys/nexusdebug.h

Various debug definitions.

15.1.74 usr/src/uts/s390x/sys/rootnex.h

System z root nexus implementation specific state.

15.1.75 usr/src/uts/s390x/sys/smp_impldefs.h

Software interrupt related definitions.

15.1.76 usr/src/uts/s390x/sys/x_call.h

Cross-system call definitions.

15.1.77 usr/src/uts/s390x/sys/traptrace.h

Mapping of output from TRAC and TRACG instructions.

15.1.78 usr/src/uts/s390x/vm/hat_kdi.c

kdi_vtop	Convert kdi virtual address to real.

15.1.79 usr/src/uts/s390x/vm/hat_pte.h

Defines for the bits in System z Region, Segment and Page Tables.

15.1.80 usr/src/uts/s390x/vm/hat_s390x.c

Virtual Memory - Hardware Address Translation management
Implementation of the interfaces described in <common/vm/hat.h>.
Nearly all the details of how the hardware is managed should not be visible outside this layer except for misc. machine specific functions that work in conjunction with this code.

15.1.81 usr/src/uts/s390x/vm/hat_s390x.h

This file describes the contents of the sun-reference-mmu(s390x)-specific hat data structures and the s390x-specific hat procedures. The machine-independent interface is described in <vm/hat.h>.

15.1.82 usr/src/uts/s390x/vm/hment.c

Manipulate and manage hment entries.

15.1.83 usr/src/uts/s390x/vm/hment.h

Mapping related definitions.

15.1.84 usr/src/uts/s390x/vm/htable.c

htable creation, allocation, stealing, and freeing. (Each hardware page table has an htable_t describing it.)

15.1.85 usr/src/uts/s390x/vm/htable.h

htable related definitions.

15.1.86 usr/src/uts/s390x/vm/mach_vm_dep.c

Machine specific VM support routines.

map_addr_proc	
pagescrub	
sync_data_memory	
contig_mem_init	
exec_get_spslew	
page_create_contig	
page_get_contigpage	
is_contigpage_free	

15.1.87 usr/src/uts/s390x/vm/mm_s390x.h

Architecture specific virtual memory related low-level hardware defintions.

15.1.88 usr/src/uts/s390x/vm/s390xmmu.c

va_to_pfn	
va_to_pa	
as_va_to_pa	
hat_kern_setup	
ndata_alloc_init	
ndata_maxsize	
ndata_spare	
ndata_extra_base	
ndata_select_chunk	
ndata_alloc	
dumpRSP	

15.1.89 usr/src/uts/s390x/vm/vm_dep.c

UNIX machine dependent virtual memory support.

impl_obmem_pfnum	
pf_is_memory	
pagefault	
map_addr	
valid_va_range	
valid_va_range	
valid_usr_range	
map_pgszheap	
map_pgszism	
map_szvec	
map_szvec	

map_pgszstk	
map_pgsz	
map_addr_vacalign_check	
alloc_page_freelists	
ndata_alloc_page_freelists	
pageout_init	
kdi_range_is_nontoxic	
page_coloring_init	
bp_color	

15.1.90 usr/src/uts/s390x/vm/vm_dep.h

UNIX machine dependent virtual memory support definitions.

15.1.91 usr/src/uts/zSeries/amsrc2/main.c

A pseudo-kernel to use when analyzing am_src2.c with warlock.

15.1.92 usr/src/uts/zSeries/asm/cpuvar.h

Architecture specific definitions relating to retrieving current process information.

15.1.93 usr/src/uts/zSeries/asm/flush.h

Architecture specific version of doflush()

15.1.94 usr/src/uts/zSeries/asm/sunddi.h

Instruction/memory synchronization definitions.

15.1.95 usr/src/uts/zSeries/asm/thread.h

Architecture specific kernel thread definitions (essentially empty for System z).

15.1.96 usr/src/uts/zSeries/fs/proc/prmachdep.c

/proc file system support routines.

15.1.97 usr/src/uts/zSeries/io/beeper.c

Beeper functions - no operations.

15.1.98 usr/src/uts/zSeries/io/ccw/ccw_autoconfig.c

Discover I/O configuration and create device tree.

15.1.99 usr/src/uts/zSeries/io/ccw/con3215.c

z/VM 3215 console driver.

15.1.100 usr/src/uts/zSeries/io/ccw/diag250_hl.c

Interface between diag250 DASD driver and the Solaris DDI stack.

15.1.101 usr/src/uts/zSeries/io/ccw/diag250_ll.c

DIAG 250 low-level disk driver.

15.1.102 usr/src/uts/zSeries/io/ccw/osa.c

Network driver for OSA card using DIAG interface.

15.1.103 usr/src/uts/zSeries/io/consplat.c

Console configuration routines.

15.1.104 usr/src/uts/zSeries/io/polled_io.c

Polled I/O support routines - all no-ops as Systemn z does not support this type of polled I/O.

15.1.105 usr/src/uts/zSeries/krtld/doreloc.c

Perform basic relocations for kernel modules.

15.1.106 usr/src/uts/zSeries/krtld/kobj_alloc.c

Miscellaneous storage allocation type routines.

15.1.107 usr/src/uts/zSeries/krtld/kobj_convrelstr.c

Convert a relocation type into a string id.

15.1.108 usr/src/uts/zSeries/krtld/kobj_crt.s

Exit routine from linker/loader to kernel.

15.1.109 usr/src/uts/zSeries/krtld/kobj_isa.c

Various kobj support routines for z/Architecture.

15.1.110 usr/src/uts/zSeries/krtld/kobj_reloc.c

Perform relocations for kernel modules.

15.1.111 usr/src/uts/zSeries/krtld/relmach.h

Architecture specific flags presented in a architecture neutral format.

15.1.112 usr/src/uts/zSeries/ml/copy.s

Various copying routines within kernel and between address spaces.

15.1.113 usr/src/uts/zSeries/ml/ip_ocsum.s

Do a 16 bit one's complement sum of a given number of (16-bit) halfwords.

15.1.114 usr/src/uts/zSeries/ml/modstubs.s

This file contains the stubs routines for modules which can be autoloaded.

15.1.115 usr/src/uts/zSeries/ml/xlate.s

Provide simple ASCII<->EBCDIC translation..

15.1.116 usr/src/uts/zSeries/os/archdep.c

Architecturally dependent routines.

getpcstack	
elfheadcheck	
bind_hwcap	
__ipltospl	
traceback	
traceregs	
exec_set_sp	
boot_virt_alloc	
bop_alloc	
bop_allreal	
xcopyin_nta	

xcopyout_nta	
kcopy_nta	
ucontext_32ton	
diag_24	
diag_a8	
diag_210	

15.1.117 usr/src/uts/zSeries/os/ddi_arch.c

Architectural specific DDI support routines.

15.1.118 usr/src/uts/zSeries/os/door_support.c

Platform specific doorfs support routines.

15.1.119 usr/src/uts/zSeries/os/syscall.c

System call support routines.

lwp_getsysent	
realsigprof	
get_syscall_args	
get_syscall32_args	
save_syscall_args	
reset_syscall_args	
nosys	
pre_syscall	
post_syscall	
syscall_ap	
lock_syscall	
loadable_syscall	
indir	
set_errno	
set_proc_pre_sys	
set_proc_post_sys	
set_proc_sys	

set_all_proc_sys	
set_proc_ast	

15.1.120 usr/src/uts/zSeries/os/cpcmd.c

Execute a CP command on behalf of an authorized user and return the results.

15.1.121 usr/src/uts/zSeries/promif/prom_emul.c

Boot PROM emulation routines.

15.1.122 usr/src/uts/zSeries/promif/prom_env.c

Various Boot PROM support routines.

15.1.123 usr/src/uts/zSeries/promif/prom_getchar.c

Return a byte read via the HMC.

15.1.124 usr/src/uts/zSeries/promif/prom_init.c

Initialize some stuff before using the prom_put/get/print routines.

15.1.125 usr/src/uts/zSeries/promif/prom_node.c

Routines for walking the PROMs devinfo tree. The prom tree is for /dev/openprom compatibility only, so we fail all calls except those needed by /dev/openprom

15.1.126 usr/src/uts/zSeries/promif/prom_printf.c

printf() support for HMC console output.

15.1.127 usr/src/uts/zSeries/promif/prom_prop.c

Emulation of prom-based proposition processing.

15.1.128 usr/src/uts/zSeries/promif/prom_putchar.c

Put a character into the buffer used by prom_printf() etc.

15.1.129 usr/src/uts/zSeries/promif/prom_reboot.c

Reboot the system.

15.1.130 usr/src/uts/zSeries/promif/prom_version.c

Emulate PROM version retrieval.

15.1.131 usr/src/uts/zSeries/sockfs/nl7ctokgen.h

sockfs related definitions.

15.1.132 usr/src/uts/zSeries/sys/archsystm.h

A selection of ISA-dependent interface definitions.

15.1.133 usr/src/uts/zSeries/sys/asm_linkage.h

Platform-specific definitions of entry/exit/linkage conventions and constants.

15.1.134 usr/src/uts/zSeries/sys/blockio.h

Defines the data structures and prototypes for doing disk I/O using the DIAG 0x250 interface.

15.1.135 usr/src/uts/zSeries/sys/bootconf.h

Boot time configuration information objects.

15.1.136 usr/src/uts/zSeries/sys/bootvfs.h

Boot time file system definitions.

15.1.137 usr/src/uts/zSeries/sys/ccw.h

CCW bus nexus driver definitions.

15.1.138 usr/src/uts/zSeries/sys/cpu.h

Platform-specific CPU management related definitions.

15.1.139 usr/src/uts/zSeries/sys/ddi_isa.h

Platform-specific DDI interface defintions.

15.1.140 usr/src/uts/zSeries/sys/devinit.h

Device initialization control block definitions.

15.1.141 usr/src/uts/zSeries/sys/diag250_ll.h

DIAG 250 disk driver structures.

15.1.142 usr/src/uts/zSeries/sys/exts390x.h

External interrupt handling related definitions.

15.1.143 usr/src/uts/zSeries/sys/fasttrap_isa.h

DTrace related definitions – place holder at the moment.

15.1.144 usr/src/uts/zSeries/sys/frame.h

Definition of the System z stack frame.

15.1.145 usr/src/uts/zSeries/sys/inline.h

Assembler inline routines – empty.

15.1.146 usr/src/uts/zSeries/sys/ios390x.h

System z I/O related structures (e.g. SCSW).

15.1.147 usr/src/uts/zSeries/sys/kdi_machimpl.h

Kernel CPU device interface definitions.

15.1.148 usr/src/uts/zSeries/sys/machelf.h

Make machine class dependent data types transparent to the common code.

15.1.149 usr/src/uts/zSeries/sys/machlock.h

Platform specific lock related definitions.

15.1.150 usr/src/uts/zSeries/sys/machsig.h

Machine dependent portion of siginfo.h.

15.1.151 /usr/src/uts/zSeries/sys/machs390x.h

System z Architecture loww-level definitions (e.g. control register contents and flags).

15.1.152 usr/src/uts/zSeries/sys/machtypes.h

Machine dependent type definitions.

15.1.153 usr/src/uts/zSeries/sys/osa.h

Declarations for the OSA driver.

15.1.154 usr/src/uts/zSeries/sys/memlist_plat.h

Boot time configuration information objects.

15.1.155 usr/src/uts/zSeries/sys/memnode.h

Defines the mappings between physical addresses and memory nodes.

15.1.156 usr/src/uts/zSeries/sys/mutex_impl.h

Platform-specific mutex implementation definitions.

15.1.157 usr/src/uts/zSeries/sys/obpdefs.h

Open Boot PROM standalone inclusion by non-prom library functions that need it.

15.1.158 usr/src/uts/zSeries/sys/old_procfs.h

Definitions for the old ioctl()-based version of the process file system.

15.1.159 usr/src/uts/zSeries/sys/pcb.h

Sun software process control block.

15.1.160 usr/src/uts/zSeries/sys/polled_io.h

Polled I/O related definitions.

15.1.161 usr/src/uts/zSeries/sys/privregs.h

Describes the cpu's privileged register set, and how the machine state is saved on the stack when a trap occurs.

15.1.162 usr/src/uts/zSeries/sys/procfs_isa.h

Instruction Set Architecture specific component of <sys/procfs.h>.

15.1.163 usr/src/uts/zSeries/sys/prom_debug.h

PROM debug routine definitions.

15.1.164 usr/src/uts/zSeries/sys/prom_emul.h

PROM emulations definitions.

15.1.165 usr/src/uts/zSeries/sys/prom_plat.h

Platform-specific promif interface definitions for s390x platforms.

15.1.166 usr/src/uts/zSeries/sys/promif.h

Boot PROM related definitions.

15.1.167 usr/src/uts/zSeries/sys/promimpl.h

Promif implementation functions and variables. These interfaces are not 'exported' in the same sense that those described in `promif.h`. Used so that the kernel and other stand-alones (e.g. boot) don't have to directly reference the prom (of which there are now several completely different variants).

15.1.168 usr/src/uts/zSeries/sys/psw.h

Macros to decode PSR.

15.1.169 usr/src/uts/zSeries/sys/reg.h

Solaris 7 compatability register definitions.

15.1.170 usr/src/uts/zSeries/sys/regset.h

Register set definitions and `mcontext_t` definitions.

15.1.171 usr/src/uts/zSeries/sys/sclp.h

Service processor Command Control Block definitions.

15.1.172 usr/src/uts/zSeries/sys/smp.h

Signal processor (sigp) related definitions.

15.1.173 usr/src/uts/zSeries/sys/spl.h

System priority level related definitions.

15.1.174 usr/src/uts/zSeries/sys/stack.h

64-bit and 32-bit ABI compatible stack definitions.

15.1.175 usr/src/uts/zSeries/sys/stat_impl.h

Implementation specific header for <sys/stat.h>.

15.1.176 usr/src/uts/zSeries/sys/synch32.h

32-bit definitions for synchronization primitives.

15.1.177 usr/src/uts/zSeries/sys/sysconfig_impl.h

Platform-specific variables for the SUN private sysconfig syscall.

15.1.178 usr/src/uts/zSeries/sys/trap.h

Trap related definitions (e.g. program exception codes).

15.1.179 usr/src/uts/zSeries/sys/ucontext.h

Platform implementation of user context definitions.

15.1.180 usr/src/uts/zSeries/sys/utrap.h

User trap definitions – currently unimplemented.

15.1.181 usr/src/uts/zSeries/sys/vm_machparam.h

Machine dependent constants for System z.

15.1.182 usr/src/uts/zSeries/sys/vmparam.h

Machine dependent virtual memory related parameter definitions.

15.1.183 usr/src/uts/zSeries/syscall/getcontext.c

Platform-specific implementation of get/save/restore context.

15.1.184 usr/src/uts/zSeries/syscall/install_utrap.c

install_utrap syscall hander (simply returns ENOSYS).

15.1.185 usr/src/uts/zSeries/syscall/cpcmd.c

Execute a CP command on behalf of an authorized user and return the results.

15.2 Commands

This section describes new files in the `cmd` directory.

15.2.1 Link/Loader Related

These files live in the `sgs` directory.

Component	Sub Component	Architecture	File
crle		s390	msg.h
			msg.c
		s390x	msg.h
			msg.c
elfdump		s390	msg.h
			msg.c
		s390x	msg.h
			msg.c
include		s390	machdep_s390.h
			machdep.h
ld		s390	msg.h
			msg.c
		s390x	msg.h
			msg.c
ldd		s390	msg.h
			msg.c
		s390x	msg.h
			msg.c
ldprof		s390	msg.h
			msg.c
		s390x	msg.h
			msg.c
libconv		s390	arch_msg.h
			cap_msg.h
			report_bufsize.h
			c_literal_msg.h
			config_msg.h
			corenote_msg.h
			dynamic_msg.h
			elf_msg.h
			globals_msg.h
			data_msg.h
			deftag_msg.h
			phdr_msg.h
			relocate_amd64_msg.h

			relocate_i386_msg.h
			relocate_sparc_msg.h
			sections_msg.h
			demangle_msg.h
			symbols_msg.h
			symbols_sparc_msg.h
			dl_msg.h
			dwarf_ehe_msg.h
			group_msg.h
			lddstub_msg.h
			segments_msg.h
			version_msg.h
			relocate_s390_msg.h
			syminfo_msg.h
		s390x	report_bufsize.h
			arch_msg.h
			c_literal_msg.h
			config_msg.h
			corenote_msg.h
			data_msg.h
			deftag_msg.h
			demangle_msg.h
			dl_msg.h
			dwarf_ehe_msg.h
			group_msg.h
			lddstub_msg.h
			segments_msg.h
			version_msg.h
			cap_msg.h
			dynamic_msg.h
			elf_msg.h
			globals_msg.h
			phdr_msg.h
			relocate_i386_msg.h
			relocate_amd64_msg.h
			relocate_s390_msg.h
			relocate_sparc_msg.h
			sections_msg.h
			symbols_msg.h
			symbols_sparc_msg.h
			syminfo_msg.h
libcrle		s390	msg.h
			msg.c
		s390x	msg.h

			msg.c
libelf		s390	msg.h
			msg.c
			xlate.c
			xlate64.c
		s390x	msg.h
			msg.c
			xlate.c
			xlate64.c
libld		s390	msg.h
			msg.c
		s390x	msg.h
			msg.c
liblddbg		s390	msg.h
			msg.c
		s390x	msg.h
			msg.c
libldstab		s390	msg.h
			msg.c
		s390x	msg.h
			msg.c
librtld		s390	_relocate.c
			msg.h
			msg.c
		s390x	_relocate.c
			msg.h
			msg.c
librtld_db		s390	plt32_resolution.c
			msg.h
			msg.c
		s390x	plt64_resolution.c
			msg.h
			msg.c
moe		s390	msg.h
			msg.c
		s390x	msg.h
			msg.c
pvs		s390	msg.h
			msg.c
		s390x	msg.h
			msg.c
Rtld	mdbmod	s390	msg.h
			msg.c
		s390x	msg.h

			msg.c
		s390	_setup.c
			gnu_pragma.h
			s390_elf.c
			msg.h
			msg.c
		s390x	_setup.c
			gnu_pragma.h
			s390x_elf.c
			msg.h
			msg.c

15.2.2 Other

This section contains the new files added to the rest of the `cmd` tree. Note the dtrace and mdb trees are not listed as they only contain partial or stub code.

Component	Sub Component	Architecture	File
agents	snmp agent	s390	personal.c
			personal.lex.c
		s390x	personal.c
			personal.lex.c
csh		s390	signal.c
			signal.h
			sh.tconst.h
devfsadm		s390	misc_link_s390.c
		s390x	misc_link_s390x.c
dfs.cmds	sharectl	s390	shareutil.c
eeprom		s390x	noprom.c
fm	eversholt eftinfo	s390	y.tab.h
			y.tab.c
	eversholt esc	s390	y.tab.h
			y.tab.c
	fmd	s390	fmd_svc_adm.c
			fmd_svc_api.c
			fmd_xdr_adm.c
			fmd_xdr_api.c
	fminject	s390	inj_grammar.h
			inj_grammar.c
			inj_lex.c
	schemes cpu	s390	cpu_mdesc.c
			cpu_mdesc.h
		s390x	cpu_mdesc.c

			cpu_mdesc.h
	schemes mem	s390	mem_disc.c
		s390x	mem_disc.c
geniconvtbl		s390	y.tab.h
			y.tab.c
			lex.yy.c
ipf	tools	s390	ipf.tab.h
			ipf.tab.c
			ipmon.tab.h
			ipmon.tab.c
			ipnat.tab.h
			ipnat.tab.c
			ippool.tab.h
			ippool.tab.c
		s390x	ipf.tab.h
			ipf.tab.c
			ipmon.tab.h
			ipmon.tab.c
			ipnat.tab.h
			ipnat.tab.c
			ippool.tab.h
			ippool.tab.c
lvm	rpc.mdcommd	s390	mdmn_commd_xdr.c
	rpc.metad	s390	metad_svc.c
	rpc.metamedd	s390	metamed_svc.c
			metamed_xdr.c
			meta_basic_xdr.c
	rpc.metamhd	s390	mhdx_xdr.c
			metamhd_svc.c
			metamhd_xdr.c
prtdiag		s390x	smbios.c
tnf	prex	s390	y.tab.h
			prexgram.c
			prexlex.c
		s390x	y.tab.h
			prexgram.c
			prexlex.c

15.3 Libraries

These new files live in the lib directory.

Component	Sub Component	Architecture	File

fm	libfmd_adm	s390	fmd_rpc_adm.h
			fmd_rpc.c
			fmd_xdr.c
		s390x	fmd_rpc_adm.h
			fmd_rpc.c
			fmd_xdr.c
gss_mechs	mech_krb5	s390	kwarnd.h
			kwarnd_clnt_stubs.c
			kwarnd_clnt.c
			kwarnd_handle.c
			kwarnd_xdr.c
		s390x	kwarnd.h
			kwarnd_handle.c
			kwarnd_clnt_stubs.c
			kwarnd_clnt.c
			kwarnd_xdr.c
lvm	libmeta	s390	mdiox_xdr.c
			meta_basic_xdr.c
			metad_clnt.c
			metamhd_xdr.c
			mhdx_xdr.c
			metad_xdr.c
			metamed_clnt.c
			metamed_xdr.c
			metamhd_clnt.c
			mdmn_commd_xdr.c
libc		s390	genassym.c
			assym.h
		s390x	genassym.h
			assym.h
	crt	s390	_ftou.c
		s390x	_ftou.c
	fp	s390	_D_cplx_div.c
			_D_cplx_div_ix.c
			_D_cplx_div_rx.c
			_D_cplx_lr_div.c
			_D_cplx_lr_div_ix.c
			_D_cplx_lr_div_rx.c
			_D_cplx_mul.c
			_F_cplx_div.c
			_F_cplx_div_ix.c
			_F_cplx_div_rx.c
			_F_cplx_lr_div.c
			_F_cplx_lr_div_ix.c

			_F_cplx_lr_div_rx.c
			_F_cplx_mul.c
			_Qp_qtoux.c
			_Qp_qtox.c
			_Qp_uxtoq.c
			_Qp_xtoq.c
			_X_cplx_div.c
			_X_cplx_div_ix.c
			_X_cplx_div_rx.c
			_X_cplx_lr_div.c
			_X_cplx_lr_div_ix.c
			_X_cplx_lr_div_rx.c
			_X_cplx_mul.c
			_base_il.c
			fp.h
			fpgetmask.c
			fpgetround.c
			fpgetsticky.c
			fpsetmask.c
			fpsetround.c
			fpsetsticky.c
			fpstart.c
		s390x	_D_cplx_div.c
			_D_cplx_div_ix.c
			_D_cplx_div_rx.c
			_D_cplx_lr_div.c
			_D_cplx_lr_div_ix.c
			_D_cplx_lr_div_rx.c
			_D_cplx_mul.c
			_F_cplx_div.c
			_F_cplx_div_ix.c
			_F_cplx_div_rx.c
			_F_cplx_lr_div.c
			_F_cplx_lr_div_ix.c
			_F_cplx_lr_div_rx.c
			_F_cplx_mul.c
			_Qp_qtoux.c
			_Qp_qtox.c
			_Qp_uxtoq.c
			_Qp_xtoq.c
			_X_cplx_div.c
			_X_cplx_div_ix.c
			_X_cplx_div_rx.c
			_X_cplx_lr_div.c

			_X_cplx_lr_div_ix.c
			_X_cplx_lr_div_rx.c
			_X_cplx_mul.c
			_base_il.c
			fp.h
			fpgetmask.c
			fpgetround.c
			fpgetsticky.c
			fpsetmask.c
			fpsetround.c
			fpsetsticky.c
			fpstart.c
	gen	s390	byteorder.c
			ecvt.c
			getctxt.c
			lexp10.c
			llog10.c
			makectxt.c
			memmove.c
			siginfolst.c
			siglongjmp.c
			strcasecmp.c
			swapctxt.c
			sync_instruction_memory.c
		s390x	byteorder.c
			ecvt.c
			getctxt.c
			lexp10.c
			llog10.c
			makectxt.c
			memmove.c
			siginfolst.c
			siglongjmp.c
			strcasecmp.c
			swapctxt.c
			sync_instruction_memory.c
	inc	s390	SYS.h
		s390x	SYS.h
	sys	s390	ptrace.c
		s390x	ptrace.c
	threads	s390	machdep.c
		s390x	machdep.c
libcpc		s390	conf_s390x.c
			event_s390x.c

		s390x	conf_s390x.c
			event_s390x.c
libdisasm		s390x	dis_s390x.c
libdll		s390	dlldefs.h
		s390x	dlldefs.h
libdtrace		s390	dt_isadep.c
			dt_lex.c
			dt_grammar.h
			dt_grammar.c
		s390x	dt_isadep.c
			dt_lex.c
			dt_grammar.h
			dt_grammar.c
libpp		s390	ppdebug.h
			ppdef.h
			pptab.h
libproc		s390	Pisadep.c
		s390x	Pisadep.c
libsec		s390	acl.tab.h
			acl_lex.c
			acl.tab.c
		s390x	acl.tab.h
			acl_lex.c
			acl.tab.c
libsqlite		s390	opcodes.h
			lempar.c
			parse_tmp.c
			parse_tmp.h
			parse.h
			opcodes.c
			parse.c
libsys		s390	libsys.c
smbsrv	libmlrpc	s390	rpcpdu_ndr.c
		s390x	rpcpdu_ndr.c
	libmlsvc	s390	dssetup_ndr.c
			eventlog_ndr.c
			lsarpc_ndr.c
			netdfs_ndr.c
			netlogon_ndr.c
			samrpc_ndr.c
			spoolss_ndr.c
			srvsvc_ndr.c
			svcctl_ndr.c
			winreg_ndr.c

		s390x	dssetup_ndr.c
			lsarpc_ndr.c
			eventlog_ndr.c
			netdfs_ndr.c
			samrpc_ndr.c
			netlogon_ndr.c
			spoolss_ndr.c
			srvsvc_ndr.c
			svcctl_ndr.c
			winreg_ndr.c

15.4 Tools

This section describes new files added to the `tools` subdirectory.

15.4.1 usr/src/tools/kipl_cvt/kipl_cvt.c

This file contains the tool to convert the unix elf object into an IBM text deck that can then be written to the boot device.

15.5 ucblib

There are two new files in the `ucblib` directory:

1. libucb/s390/sys/signal.c
2. libucb/s390x/sys/signal.c

16. Scripts used to Build File System

The following scripts are used to build the ramdisk and to upload it and the kernel to a z/VM system where it can be booted.

16.1 bldram

This script creates a RAMDISK containing enough of an OpenSolaris system to boot.

```
#!/bin/bash
export PATH=/usr/sbin:/sbin:$PATH
ME=`whoami`
id_neale=1
id_mcohan=2
id_adam=3
id_llucius=4
id_tide=5
base[1]="ibm"
base[2]="sirius"
base[3]="sirius"
base[4]="sirius"
base[5]="sirius"
dir=${base[$((id_$ME))]}
umount $HOME/sirius/mnt
size=36m
USERDIR="ramdisk.d"

if [ ! -d $HOME/sirius/mnt ]
then
      mkdir -p $HOME/sirius/mnt
fi

while getopts s:r name
do
      case $name in
            r)
                  USERDIR="user.d"
            ;;
            s)
                  rm -f $HOME/sirius/sirius.ramdisk
                  size=$OPTARG
            ;;
      esac
done
shift $(($OPTIND - 1))

if [ ! -f $HOME/sirius/sirius.ramdisk ]
then
      lofiadm -d $HOME/sirius/sirius.ramdisk
      mkfile $size $HOME/sirius/sirius.ramdisk
      chown $ME $HOME/sirius/sirius.ramdisk
      lofiadm -a $HOME/sirius/sirius.ramdisk /dev/lofi/$((id_$ME))
      newfs -v /dev/rlofi/$((id_$ME)) <<___NEWFS
      y
___NEWFS
fi
mount /dev/lofi/$((id_$ME)) $HOME/sirius/mnt
```

```
SOURCE=$HOME/OpenSolaris/$dir/onnv-gate/usr/src/uts
SOURCEZ=$SOURCE/zSeries
SOURCES=$SOURCE/s390x
SOURCEC=$SOURCE/common

if [ -d $SOURCES/unix/debug64 ]
then
     OBJDIR=debug64
else
     OBJDIR=obj64
fi

SOURCEU=$SOURCES/unix/$OBJDIR
SOURCEG=$SOURCES/genunix/$OBJDIR

cd $HOME/sirius/mnt
rm -rf $HOME/sirius/mnt/*
mkdir -p kernel/misc/s390x/s390x
mkdir -p kernel/crypto/s390x
mkdir -p platform/s390x/exec
mkdir -p platform/s390x/kernel/z9
mkdir -p kernel/dacf/s390x
mkdir -p kernel/drv/s390x/s390x
mkdir -p kernel/fs/s390x/s390x
mkdir -p kernel/sched/s390x
mkdir -p kernel/strmod/s390x
mkdir -p kernel/sys/s390x
mkdir -p kernel/exec/s390x
mkdir -p etc
mkdir -p etc/svc/volatile
mkdir -p devices/pseudo
mkdir -p dev/pts
mkdir -p mnt
mkdir -p proc
mkdir -p sbin
mkdir -p system/object
mkdir -p system/contract
mkdir -p tmp
cp $SOURCEU/unix platform/s390x/unix
cp $SOURCEZ/consms/$OBJDIR/consms kernel/drv/s390x/kernel/z9
cp $SOURCES/rootnex/$OBJDIR/rootnex kernel/drv/s390x/
cp $SOURCES/ccwnex/$OBJDIR/ccwnex kernel/drv/s390x/
cp $SOURCES/cpunex/$OBJDIR/cpunex kernel/drv/s390x/
cp $SOURCEZ/consconfig_dacf/$OBJDIR/consconfig_dacf kernel/dacf/s390x/
cp $SOURCEZ/conskbd/$OBJDIR/conskbd  kernel/drv/s390x/
cp $SOURCEZ/consms/$OBJDIR/consms kernel/drv/s390x/
cp $SOURCEZ/diag250/$OBJDIR/diag250 kernel/drv/s390x/
cp $SOURCEZ/iwscn/$OBJDIR/iwscn kernel/drv/s390x/
cp $SOURCEZ/kstat/$OBJDIR/kstat kernel/drv/s390x/
cp $SOURCEZ/mm/$OBJDIR/mm kernel/drv/s390x/
cp $SOURCEZ/options/$OBJDIR/options kernel/drv/s390x/
cp $SOURCEZ/pseudo/$OBJDIR/pseudo kernel/drv/s390x/
cp $SOURCEZ/ramdisk/$OBJDIR/ramdisk kernel/drv/s390x/
cp $SOURCEZ/doorfs/$OBJDIR/doorfs kernel/drv/s390x/
cp $SOURCEZ/wc/$OBJDIR/wc kernel/drv/s390x/
cp $SOURCEZ/osa/$OBJDIR/osa kernel/drv/s390x/
cp $SOURCEZ/arp/$OBJDIR/arp kernel/drv/s390x/
cp $SOURCEZ/udp/$OBJDIR/udp kernel/drv/s390x/
cp $SOURCEZ/icmp/$OBJDIR/icmp kernel/drv/s390x/
cp $SOURCEZ/tcp/$OBJDIR/tcp kernel/drv/s390x/
cp $SOURCEZ/dld/$OBJDIR/dld kernel/drv/s390x/
cp $SOURCEZ/sad/$OBJDIR/sad kernel/drv/s390x/
cp $SOURCEZ/clone/$OBJDIR/clone kernel/drv/s390x/
```

```
cp $SOURCEZ/ptsl/$OBJDIR/ptsl kernel/drv/s390x/
cp $SOURCEZ/ptc/$OBJDIR/ptc kernel/drv/s390x/
cp $SOURCEZ/devinfo/$OBJDIR/devinfo kernel/drv/s390x/
cp $SOURCEZ/ip/$OBJDIR/ip kernel/drv/s390x/
cp $SOURCEZ/cn/$OBJDIR/cn kernel/drv/s390x/
cp $SOURCEG/genunix platform/s390x/kernel/z9
cp $SOURCEZ/specfs/$OBJDIR/specfs kernel/fs/s390x/
cp $SOURCEZ/devfs/$OBJDIR/devfs kernel/fs/s390x/
cp $SOURCEZ/dev/$OBJDIR/dev kernel/fs/s390x/
cp $SOURCEZ/ufs/$OBJDIR/ufs kernel/fs/s390x/
cp $SOURCEZ/mntfs/$OBJDIR/mntfs kernel/fs/s390x/
cp $SOURCEZ/ctfs/$OBJDIR/ctfs kernel/fs/s390x/
cp $SOURCEZ/tmpfs/$OBJDIR/tmpfs kernel/fs/s390x/
cp $SOURCEZ/objfs/$OBJDIR/objfs kernel/fs/s390x/
cp $SOURCEZ/procfs/$OBJDIR/procfs kernel/fs/s390x/
cp $SOURCEZ/fifofs/$OBJDIR/fifofs kernel/fs/s390x/
cp $SOURCEZ/namefs/$OBJDIR/namefs kernel/fs/s390x/
cp $SOURCEZ/fdfs/$OBJDIR/fdfs kernel/fs/s390x/
cp $SOURCEZ/lofs/$OBJDIR/lofs kernel/fs/s390x/
cp $SOURCEZ/sockfs/$OBJDIR/sockfs kernel/fs/s390x/
cp $SOURCEZ/md5/$OBJDIR/md5 kernel/misc/s390x/
cp $SOURCEZ/kcf/$OBJDIR/kcf kernel/misc/s390x/
cp $SOURCEZ/neti/$OBJDIR/neti kernel/misc/s390x/
cp $SOURCEZ/gld/$OBJDIR/gld kernel/misc/s390x/
cp $SOURCEZ/dls/$OBJDIR/dls kernel/misc/s390x/
cp $SOURCEZ/hook/$OBJDIR/hook kernel/misc/s390x/
cp $SOURCEZ/consconfig/$OBJDIR/consconfig kernel/misc/s390x/
cp $SOURCEZ/fssnap_if/$OBJDIR/fssnap_if kernel/misc/s390x/
cp $SOURCEZ/kbtrans/$OBJDIR/kbtrans kernel/misc/s390x/
cp $SOURCEZ/ipc/$OBJDIR/ipc kernel/misc/s390x/
cp $SOURCEZ/mac/$OBJDIR/mac kernel/misc/s390x/
cp $SOURCEZ/strplumb/$OBJDIR/strplumb kernel/misc/s390x/
cp $SOURCEZ/swapgeneric/$OBJDIR/swapgeneric kernel/misc/s390x/
cp $SOURCEZ/tem/$OBJDIR/tem kernel/misc/s390x/
cp $SOURCEZ/ccw_autoconfig/$OBJDIR/ccw_autoconfig kernel/misc/s390x/
cp $SOURCEZ/swrand/$OBJDIR/swrand kernel/crypto/s390x/
cp $SOURCEZ/sha1/$OBJDIR/sha1 kernel/crypto/s390x/
cp $SOURCEZ/timod/$OBJDIR/timod kernel/strmod/s390x/
cp $SOURCEZ/ldterm/$OBJDIR/ldterm kernel/strmod/s390x/
cp $SOURCEZ/ttcompat/$OBJDIR/ttcompat kernel/strmod/s390x/
cp $SOURCEZ/c2audit/$OBJDIR/c2audit kernel/sys/s390x/
cp $SOURCEZ/doorfs/$OBJDIR/doorfs kernel/sys/s390x/
cp $SOURCEZ/kaio/$OBJDIR/kaio kernel/sys/s390x/
cp $SOURCEZ/msgsys/$OBJDIR/msgsys kernel/sys/s390x/
cp $SOURCEZ/pipe/$OBJDIR/pipe kernel/sys/s390x/
cp $SOURCEZ/semsys/$OBJDIR/semsys kernel/sys/s390x/
cp $SOURCEZ/shmsys/$OBJDIR/shmsys kernel/sys/s390x/
cp $SOURCEZ/TS/$OBJDIR/TS kernel/sched/s390x/
cp $SOURCEZ/TS_DPTBL/$OBJDIR/TS_DPTBL kernel/sched/s390x/
cp $SOURCEZ/elfexec/$OBJDIR/elfexec platform/s390x/exec/
cp $SOURCE/common/io/ramdisk.conf etc/
cp $SOURCE/common/io/kstat.conf kernel/drv
cp $HOME/sirius/etc/mnttab etc/
cp $HOME/sirius/etc/system etc/
cp $SOURCEZ/os/dacf.conf etc/
cp $SOURCEZ/os/device_policy etc/
cp $SOURCEZ/os/driver_aliases etc/
cp $SOURCEZ/os/driver_classes etc/
cp $SOURCEZ/os/minor_perm etc/
cp $SOURCEZ/os/name_to_major etc/
cp $SOURCEZ/os/name_to_sysnum etc/
cp $SOURCEZ/os/path_to_inst etc/
cp $SOURCEZ/os/priv_names etc/
```

```
cp $SOURCEZ/io/ccw/*.conf kernel/drv/
cp /kernel/drv/*.conf kernel/drv/
mknod devices/pseudo/cn\@0:console c 0 0
cd dev
ln -s ../devices/pseudo/cn\@0:console console
mkdir dsk
cd dsk
ln -s ../../devices/ccw/dasd@0x0200:dasd c0d512s3
ln -s ../../devices/ccw/dasd@0x0201:dasd c0d513s3
ln -s ../../devices/ccw/dasd@0x0202:dasd c0d514s3
ln -s ../../devices/ccw/dasd@0x0300:dasd c0d768s3
cd ..
mkdir rdsk
cd rdsk
ln -s ../../devices/ccw/dasd@0x0200:dasd,raw c0d512s3
ln -s ../../devices/ccw/dasd@0x0201:dasd,raw c0d513s3
ln -s ../../devices/ccw/dasd@0x0202:dasd,raw c0d514s3
ln -s ../../devices/ccw/dasd@0x0300:dasd,raw c0d768s3
cd ../..
if [ -e "$SOURCEZ/con3215/$OBJDIR/con3215" ]
then
      cp $SOURCEZ/con3215/$OBJDIR/con3215 kernel/drv/s390x/
fi
if [ -d "$HOME/sirius/$USERDIR" ]
then
      cd $HOME/sirius/$USERDIR/
      find . | cpio -pvdu $HOME/sirius/mnt >/dev/null
fi
cd /

umount $HOME/sirius/mnt
mount /dev/lofi/$((id_$ME)) $HOME/sirius/mnt
```

16.2 upload

This script invokes the `kipl_cvt` command to convert the unix elf object into a “text deck” understood by IBM operating systems such as CMS. It then uploads this to a z/VM system where it can be post-processed and written to the boot device.

```
#!/bin/bash
ME=`whoami`
PASS=xxxxxxx
TARGET=xx.xx.xx.xx
FILEPOOL=”vmsysu:.”
id_neale=1
id_mcohan=2
id_adam=3
id_llucius=4
id_tide=5
kipl_cvt -o $SRC/uts/s390x/unix/debug64/sirius.text -u \
$SRC/uts/s390x/unix/debug64/unix
sudo umount $HOME/sirius/mnt
ncftpput -E -B 8192 -t 10 -u tideusr1 -p $PASS $TARGET $FILEPOOL \
      $SRC/uts/s390x/unix/debug64/sirius.text
ncftpput -E -B 8192 -t 10 -u tideusr1 -p $PASS $TARGET $FILEPOOL \
      $HOME/sirius/sirius.ramdisk
sudo mount /dev/lofi/$((id_$ME)) $HOME/sirius/mnt
```

16.3 Creating IPL Volume

This script uses the "RAMDISK EXEC" to create a DCSS which will hold the initial ramdisk. It takes the file resulting from the execution of the `kipl_cvt` command on the Solaris build host and converts it into a module that is then placed on the boot disk along with the bootstrap code (from SALIPL).

```
/* */
parse upper arg UserID .
if (UserID <> '') then
do
   call PROLOG
   call BUILD_IPL
   'EXEC RAMDISK' UserID 'SIRIUS'
   call EPILOG
end
else
   say 'Usage: SOLARIS <userid>'
exit

PROLOG:

   'EXEC VMLINK' UserID '191 (WRITE QUIET .FM .CU STEM'
   if (Rc = 0) then
   do
      parse var VMLink.1 '.FM' Fm '.CU' Cuu .
      Fm  = STRIP(Fm)
      Cuu = STRIP(Cuu)

      Ctl.1 = '&CONTROL OFF'
      Ctl.2 = '&1 &2 &3 HCPLDR LOADER'
      Ctl.3 = '&1 &2 &3 SOLARIS TEXT'
      Ctl.0 = 3
   end
   else
   do
      say 'Error linking to' UserID '191'
      exit Rc
   end

return

BUILD_IPL:

   'PIPE (name SIRIUS end ?)',
      '| < SIRIUS TEXT',
      '| fblock 80',
      '| xlate 73-80 a2e',
      '| sort 73-80',
      '| a: take 1',
      '| xlate 6.8 a2e',
      '| b: faninany',
      '| > SOLARIS TEXT' Fm'3 F',
      '? a:',
      '| b:'
   'PIPE (name BLDCTL end ?)',
      '| stem Ctl.',
      '| > SIRLOAD EXEC A3'
   'HCPLDR SIRLOAD (NOCTL MAP RLDSAVE MODULE FMODE' Fm
   'SALIPL' Cuu '(MODULE SIRLOAD IPLPARMS -v'
```

```
return

EPILOG:

   CPRc = DIAGRC(8,'TERM MORE 0 0 HOLD OFF')
   CPRc = DIAGRC(8,'SP CONS STA TO NEALE CL A NAME SIRIUS LOG')
   CPRc = DIAGRC(8,'SP P NEALE')

return
```

16.4 RAMDISK EXEC

This EXEC loads the RAMDISK image from a CMS file and saves it in a DCSS with the name specified. The EXEC will also determine the size of the RAMDISK and define the appropriately sized DCSS. The user running this needs to have sufficient privileges to define and save DCSS.

```
/* */
parse upper arg UserID RamName .
if (UserID <> '') then
do
   call PROLOG
   call BUILD_RAMDISK
   call EPILOG
end
else
   say 'Usage: RAMDISK <userid>'
exit

PROLOG:

   Start   = 2000000
   SegName = UserID
   parse value RamName 'SIRIUS' with RamName .
   'PIPE (name BLDRAMDISK)',
      '| <' RamName 'RAMDISK',
      '| count bytes',
      '| var Size'
   NSS_E = (X2D(Start) + Size + 1048575) % 1048576
   NSS_E = D2X(NSS_E * 256 -1)
   CPRC  = DIAG(8,'PUR NSS' SegName)
   CPRC  = DIAG(8,'DEFSEG' SegName '2000-'NSS_E 'EW')
   'PIPE (name SEGLOAD)',
      '| command SEGMENT RESERVE' SegName,
      '| hole'
   'SETKEY 14' SegName

return

BUILD_RAMDISK:

   'PIPE (name BLDRAMDISK)',
      '| <' RamName 'RAMDISK',
      '| fblock 4096',
      '| stem Ramdisk.'
   Addr  = Start
   do I_Ramdisk = 1 to Ramdisk.0
      Rc = STORAGE(Addr,4096,Ramdisk.I_Ramdisk)
      Addr = D2X(X2D(Addr) + 4096)
   end
```

```
   say "Ramdisk - Start:" Start "End:" Addr
   CPRc = DIAG(8,'SAVESEG' SegName)

return

EPILOG:

   'SEGMENT RELEASE' UserID

return
```

17. Annotated Log of System Boot

```
00: Boot commenced for kernel built on Sep 24 2008 14:42:11
```

Prefix of 00: or 01: are from the virtual Hardware Maintenance Console and represent the (virtual) CPU upon which the write was performed.

```
00: initialize scratch memory
00: Installed physical memory @ 4400000:
00: (0x00, 0x010000000)
00: Booter occupied memory (including modules) @ 4400060:
00: (0x0100000, 0x01c7000)(0x04400000, 0x0800000)
00: Ramdisk memory @ 4400080:
00: (0x02000000, 0x02400000)
00: Available physical memory @ 4400100:
00: (0x02c7000, 0x04139000)(0x04c00000, 0x0b400000)
00: Free physical memory @ 44000e0:
00: (0x02c7000, 0x01d39000)(0x04c00000, 0x0b400000)
00: Available virtual memory @ 44000c0:
00: (0x00, 0x0100000)(0x02c7000, 0x01d39000)(0x04c00000, 0xfffffffffb3fffff)
00: DAT Enabled using RTO 4c00000
00: Creating mappings for KPM
00:         Mapping ffffffff80000000 to 0 for 256MB
00: Relocating the KRTLD/UNIX executable
00: Opening /boot/solaris/bootenv.rc
00:
00: Boot properties:
00:         0x29b410 bios-boot-device = len=4
00:         0x29b530 bootprog = len=4 hmc
00:         0x29b650 bootargs = len=2 -v
00:         0x29b770 impl-arch-name = len=6 s390x
00:         0x29b890 mfg-id = len=12 IBM 2094S54
00:         0x29b9b0 ramdisk_start = len=8
00:         0x29bad0 ramdisk_end = len=8
00:         0x29bbf0 whoami = len=34 /platform/s390x/kernel/s390x/unix
00:         0x29bd10 mfg-name = len=6 s390x
00: krtld: file=/platform/s390x/kernel/s390x/unix
00:         text: 0x100000 size: 0x13a226
00:         data: 0x23b228 dsize: 0x8d1ce
00: krtld: file=/kernel/s390x/genunix
00:         text:0x2c8a70 size: 0x3aa920
00:         data:0x6733c0 dsize: 0xaf150
00: module /platform/s390x/kernel/s390x/unix: text at
[0x100000, 0x23a225] data at 0x23b228
00: module /kernel/s390x/genunix: text at [0x2c8a70, 0x67338f] data at 0x6733c0
00: OpenSolaris on System z - Startup commenced
00: Memory size: 256MB Chunks: 1
00: 0. 0000000000000000 10000000 0
00: CPU 0 trace table starts at 266000
00: Boot CPU hardware address: 0
00: 2 CPUs detected
```

During first part of boot we determine storage size and create memory lists that describe available pages, memory used by boot, physically installed memory and virtual memory

Very early on we build page tables for the kernel and get into “DAT” mode

The genunix module is read from RAMDISK and relocated

Kernel supports memory configuration with “holes” – this one has none

We do CPU detection here but don’t bring them online until later

The TRACG instruction is used to maintain a trace table if required

```
00: ../../s390x/os/startup.c:747: 'core_base' is 0x60000000
00: ../../s390x/os/startup.c:748: 'core_end' is 0x70000000
00: ../../s390x/os/startup.c:763: 'sysSize' is 0x10000000
00: ../../s390x/os/startup.c:764: 'physmax' is 0xffff
00: ../../s390x/os/startup.c:765: 'physinstalled' is 0xffff
00: ../../s390x/os/startup.c:794: 'moddata' is 0x736958
00: ../../s390x/os/startup.c:795: 'RSPSize' is 0x971c0
00: ../../s390x/os/startup.c:796: 'nalloc_sz' is 0x1438000
00: ../../s390x/os/startup.c:797: 'nalloc_base' is 0x4eaa000
00: ../../s390x/os/startup.c:798: 'nalloc_end' is 0x62e2000
00: ../../s390x/os/startup.c:799: 'sdata' is 0x23b228
00: ../../s390x/os/startup.c:804: 'e_text' is 0x23a226
00: ../../s390x/os/startup.c:809: 'modtext' is 0x23b000
00: ../../s390x/os/startup.c:810: 'modtext_sz' is 0x0
00: ../../s390x/os/startup.c:829: 'extra_etpg' is 0x0
00: ../../s390x/os/startup.c:830: 'modtext_sz' is 0x0
00: ../../s390x/os/startup.c:831: 'extra_etva' is 0x23b000
00: ../../s390x/os/startup.c:837: 'nalloc_base' is 0x4eaa000
00: ../../s390x/os/startup.c:838: 'nalloc_end' is 0x62e2000
00: ../../s390x/os/startup.c:868: 'kmem64_base' is 0x70000000000
00: ../../s390x/os/startup.c:906: 'ndata_remain_sz' is 0x1436000
00: ../../s390x/os/startup.c:1075: 'page_hash' is 0x568c000
00: ../../s390x/os/startup.c:1076: 'memseg_base' is 0x56cd000
00: ../../s390x/os/startup.c:1077: 'kpm_pp_base' is 0x56ce000
00: ../../s390x/os/startup.c:1078: 'kpm_pp_sz' is 0x865d0
00: ../../s390x/os/startup.c:1079: 'pp_base' is 0x4eac000
00: ../../s390x/os/startup.c:1080: 'pp_sz' is 0x7df550
00: ../../s390x/os/startup.c:1081: 'alloc_base' is 0x70000000000
00: ../../s390x/os/startup.c:1089: 'kmem64_base' is 0x70000000000
00: ../../s390x/os/startup.c:1090: 'kmem64_end' is 0x70000000000
00: ../../s390x/os/startup.c:1096: 'memlist_sz' is 0x433000
00: ../../s390x/os/startup.c:1105: 'memlist' is 0x575d000
00: ../../s390x/os/startup.c:1106: 'memlist_end' is 0x5b90000
00: ../../s390x/os/startup.c:1107: 'sysbase' is 0x30000000000
00: ../../s390x/os/startup.c:1108: 'syslimit' is 0x70000000000
00: ../../s390x/vm/s390xmmu.c:416: 'wasteage' is 0x0
00: ../../s390x/os/startup.c:1150: 'memspace' is 0x575d000
00: ../../s390x/os/startup.c:1155: 'pp_base' is 0x4eac000
00: ../../s390x/os/startup.c:1156: 'memseg_base' is 0x56cd000
00: ../../s390x/os/startup.c:1157: 'npages' is 0x10cb6
00: ../../s390x/os/startup.c:1158: 'phys_avail' is 0x4400100
00: ../../s390x/os/startup.c:1166: 'availrmem' is 0xf539
00: SunOS Release 5.11 Version home/tide/OpenSolaris/sirius/onnv-gate 64-bit
00: Copyright 1983-2008 Sun Microsystems, Inc.  All rights reserved.
00: Use is subject to license terms.
00: DEBUG enabled
00: startup_modules() starting...
00: krtld: file=/kernel/fs/s390x/specfs
00:         text:0x68000000 size: 0xa6b8
00:         data:0x60000000 dsize: 0xe30
00: krtld: file=/kernel/fs/s390x/devfs
00:         text:0x6800b000 size: 0xa9b8
00:         data:0x60002000 dsize: 0xf20
00: krtld: file=/kernel/fs/s390x/dev
00:         text:0x68016000 size: 0x1d8c8
00:         data:0x60003000 dsize: 0x2e40
00: krtld: file=/kernel/misc/s390x/dls
00:         text:0x68034000 size: 0xdf10
00:         data:0x60006000 dsize: 0x21d8
00: krtld: file=/kernel/misc/s390x/mac
00:         text:0x68042000 size: 0x9398
00:         data:0x60009000 dsize: 0x1ae8
00: krtld: file=/kernel/sched/s390x/TS
```

A whole lot of debugging messages that will be disabled in the future

The module loading debug is enabled to show all kernel modules being loaded

```
00:         text:0x6804c000 size: 0x72d8
00:         data:0x6000b000 dsize: 0x1060
00: krtld: file=/kernel/sched/s390x/TS_DPTBL
00:         text:0x680159b8 size: 0x174
00:         data:0x600081d8 dsize: 0x918
00: Using default device instance data
00: startup_modules() done
00: mem = 262140K (0xffff000)
00: avail mem = 70356992
00: krtld: file=/kernel/drv/s390x/rootnex
00:         text:0x68054000 size: 0x1bb0
00:         data:0x600018fc dsize: 0x5d8
00: root nexus = s390x
00: krtld: file=/kernel/drv/s390x/options
00:         text:0x68055bb0 size: 0x28c
00:         data:0x60008af0 dsize: 0x1e0
00: krtld: file=/kernel/drv/s390x/pseudo
00:         text:0x68056000 size: 0x11c8
00:         data:0x6000aae8 dsize: 0x498
00: pseudo0 at root
00: pseudo0 is /pseudo
00: krtld: file=/kernel/drv/s390x/clone
00:         text:0x6804b398 size: 0xa44
00:         data:0x6000c060 dsize: 0x360
00: krtld: file=/kernel/drv/s390x/ccwnex
00:         text:0x68058000 size: 0x5580
00:         data:0x6000d000 dsize: 0xeb0
00: /ccw (ccwnex0) online
00: krtld: file=/kernel/drv/s390x/con3215
00:         text:0x6805e000 size: 0x2ec0
00:         data:0x6000c3c0 dsize: 0x830
00: /ccw/cns1@0x0009 (con32150) online
00: krtld: file=/kernel/drv/s390x/osa
00:         text:0x68061000 size: 0x2218
00:         data:0x6000f000 dsize: 0x660
00: krtld: file=/kernel/misc/s390x/gld
00:         text:0x68064000 size: 0x177f0
00:         data:0x60010000 dsize: 0x1dc0
00: Ethernet address = 2:0:0:0:0:6
00: CCW-device: osa@0x0bc0, osa0
00: osa0 is /ccw/osa@0x0bc0
00: /ccw/osa@0x0bc0 (osa0) online
00: CCW-device: osa@0x0bc1, osa1
00: osa1 is /ccw/osa@0x0bc1
00: /ccw/osa@0x0bc1 (osa1) online
00: CCW-device: osa@0x0bc2, osa2
00: osa2 is /ccw/osa@0x0bc2
00: /ccw/osa@0x0bc2 (osa2) online
00: krtld: file=/kernel/drv/s390x/diag250
00:         text:0x6807c000 size: 0x6068
00:         data:0x6000f660 dsize: 0x9a0
00: WARNING: Volume MNT190 has not been reserved
00:
00: WARNING: Disk 1 is not valid for I/O
00:
00: WARNING: Volume MNT19D has not been reserved
00:
00: WARNING: Disk 2 is not valid for I/O
00:
00: WARNING: Volume MNT19E has not been reserved
00:
00: WARNING: Disk 3 is not valid for I/O
00:
```

This is the System z root nexus driver being loaded and initializing

The CCW nexus driver is responsible for looking after the CCW “bus”

The OSA driver is the System z Ethernet driver. Note it gets up and detects the devices and MAC address

The DIAG250 driver is the disk driver for System z. It too detects devices and determines if they are of the correct type for use by OpenSolaris. If not they are ignored. If they are then it will determine the characteristics of the device.

```
00: WARNING: Volume TCM592 has not been reserved
00:
00: WARNING: Disk 4 is not valid for I/O
00:
00: WARNING: Volume IPL191 has not been reserved
00:
00: WARNING: Disk 5 is not valid for I/O
00:
00: NOTICE: Volume TD1200 discovered at 0200 with blockize 4096 and offset 634
00:
00: CCW-device: dasd@0x0200, diag2506
00: diag2506 is /ccw/dasd@0x0200
00: /ccw/dasd@0x0200 (diag2506) online
00: NOTICE: Volume TD1201 discovered at 0201 with blockize 4096 and offset 634
00:
00: CCW-device: dasd@0x0201, diag2507
00: diag2507 is /ccw/dasd@0x0201
00: /ccw/dasd@0x0201 (diag2507) online
00: NOTICE: Volume TD1202 discovered at 0202 with blockize 4096 and offset 634
00:
00: CCW-device: dasd@0x0202, diag2508
00: diag2508 is /ccw/dasd@0x0202
00: /ccw/dasd@0x0202 (diag2508) online
00: NOTICE: Volume TD1300 discovered at 0300 with blockize 4096 and offset 89
00:
00: CCW-device: dasd@0x0300, diag2509
00: diag2509 is /ccw/dasd@0x0300
00: /ccw/dasd@0x0300 (diag2509) online
00: WARNING: Volume IPL191 has not been reserved
00:
00: WARNING: Disk 10 is not valid for I/O
00:
00: NULL device ptr in diag250 0 instance
00: System z specific initialization complete
00: krtld: file=/kernel/fs/s390x/ufs
00:         text:0x68083000 size: 0x73b70
00:         data:0x60012000 dsize: 0x7e48
00: krtld: file=/kernel/misc/s390x/fssnap_if
00:         text:0x680f6b70 size: 0x3e4
00:         data:0x60008cd0 dsize: 0x258
00: krtld: file=/kernel/fs/s390x/ctfs
00:         text:0x680f7000 size: 0x5718
00:         data:0x6001b000 dsize: 0x11d0
00: krtld: file=/kernel/fs/s390x/procfs
00:         text:0x680fd000 size: 0x35538
00:         data:0x6001d000 dsize: 0x3670
00: krtld: file=/kernel/fs/s390x/mntfs
00:         text:0x68133000 size: 0x44e0
00:         data:0x6001a824 dsize: 0x7a8
00: krtld: file=/kernel/fs/s390x/tmpfs
00:         text:0x68138000 size: 0xc8e8
00:         data:0x60021000 dsize: 0x111b0
00: krtld: file=/kernel/fs/s390x/objfs
00:         text:0x68145000 size: 0x3048
00:         data:0x6001c1d0 dsize: 0xb88
00: WARNING: Cannot mount /etc/dfs/sharetab
00: krtld: file=/kernel/drv/s390x/ramdisk
00:         text:0x68149000 size: 0x3a90
00:         data:0x60020670 dsize: 0x7c0
00: ramdisk0 at root
00: ramdisk0 is /ramdisk
00: Bringing CPU online
00: krtld: file=/platform/s390x/kernel/drv/s390x/cpunex
```

```
00:         text:0x68015b2c size: 0x424
00:         data:0x6000cbf0 dsize: 0x2d8
00: /cpus (cpunex0) online
00: krtld: file=/kernel/sys/s390x/c2audit
00:         text:0x6814d000 size: 0x208b0
00:         data:0x60033000 dsize: 0x5d10
00: krtld: file=/kernel/fs/s390x/sockfs
00:         text:0x6816e000 size: 0x53630
00:         data:0x60039000 dsize: 0x7298
00: krtld: file=/kernel/drv/s390x/ip
00:         text:0x681c2000 size: 0x264d28
00:         data:0x60041000 dsize: 0x2b1d8
00: krtld: file=/kernel/misc/s390x/md5
00:         text:0x68427000 size: 0x5148
00:         data:0x60040298 dsize: 0x7f8
00: krtld: file=/kernel/misc/s390x/kcf
00:         text:0x6842d000 size: 0x3b6a8
00:         data:0x6006d000 dsize: 0x4608
00: krtld: file=/kernel/crypto/s390x/swrand
00:         text:0x68469000 size: 0x1f40
00:         data:0x6006c1d8 dsize: 0xe00
00: krtld: file=/kernel/misc/s390x/hook
00:         text:0x6846b000 size: 0x25dc
00:         data:0x60040a90 dsize: 0x4f8
00: krtld: file=/kernel/misc/s390x/neti
00:         text:0x6846e000 size: 0x18cc
00:         data:0x60071608 dsize: 0x6b0
00: krtld: file=/kernel/crypto/s390x/sha1
00:         text:0x68470000 size: 0x7560
00:         data:0x600321b0 dsize: 0x7b8
00: krtld: file=/kernel/misc/s390x/strplumb
00:         text:0x6814d000 size: 0x2280
00:         data:0x60032968 dsize: 0x4b8
00: krtld: file=/kernel/drv/s390x/dld
00:         text:0x68150000 size: 0xd930
00:         data:0x60033000 dsize: 0x19a8
00: pseudo-device: dld0
00: dld0 is /pseudo/dld@0
00: krtld: file=/kernel/drv/s390x/ip6
00:         text:0x6814ca90 size: 0x47c
00:         data:0x600349a8 dsize: 0x3a0
00: krtld: file=/kernel/drv/s390x/tcp
00:         text:0x6815d930 size: 0x4b4
00:         data:0x60035000 dsize: 0x3d8
00: krtld: file=/kernel/drv/s390x/tcp6
00:         text:0x681448e8 size: 0x494
00:         data:0x600353d8 dsize: 0x3a0
00: krtld: file=/kernel/drv/s390x/udp
00:         text:0x6846f8cc size: 0x4a4
00:         data:0x60035778 dsize: 0x3d8
00: krtld: file=/kernel/drv/s390x/udp6
00:         text:0x680338c8 size: 0x494
00:         data:0x60035b50 dsize: 0x3a0
00: krtld: file=/kernel/drv/s390x/sctp
00:         text:0x6807b7f0 size: 0x47c
00:         data:0x60036000 dsize: 0x3a0
00: krtld: file=/kernel/drv/s390x/sctp6
00:         text:0x680fc718 size: 0x484
00:         data:0x600363a0 dsize: 0x3a0
00: krtld: file=/kernel/drv/s390x/icmp
00:         text:0x6800a6b8 size: 0x49c
00:         data:0x60036740 dsize: 0x3d8
00: krtld: file=/kernel/drv/s390x/icmp6
```

```
00:         text:0x6800ab54 size: 0x48c
00:         data:0x60036b18 dsize: 0x3a0
00: krtld: file=/kernel/drv/s390x/arp
00:         text:0x6815e000 size: 0xc518
00:         data:0x60037000 dsize: 0x19b0
00: krtld: file=/kernel/strmod/s390x/timod
00:         text:0x68478000 size: 0x8690
00:         data:0x6007b000 dsize: 0xfc0
00: krtld: file=/kernel/drv/s390x/sad
00:         text:0x6816b000 size: 0x1c30
00:         data:0x600389b0 dsize: 0x4c8
00: krtld: file=/kernel/misc/s390x/consconfig
00:         text:0x68144d7c size: 0x22c
00:         data:0x60019e48 dsize: 0x1a0
00: krtld: file=/kernel/dacf/s390x/consconfig_dacf
00:         text:0x68481000 size: 0x4e64
00:         data:0x6007c000 dsize: 0x1558
00: krtld: file=/kernel/drv/s390x/conskbd
00:         text:0x68486000 size: 0x4c08
00:         data:0x6007e000 dsize: 0xb78
00: krtld: file=/kernel/misc/s390x/kbtrans
00:         text:0x6848b000 size: 0x60f8
00:         data:0x6007f000 dsize: 0x2060
00: krtld: file=/kernel/drv/s390x/consms
00:         text:0x68492000 size: 0x2b18
00:         data:0x6007d558 dsize: 0x918
00: krtld: file=/kernel/drv/s390x/wc
00:         text:0x68495000 size: 0x2598
00:         data:0x60081060 dsize: 0x7d0
00: krtld: file=/kernel/misc/s390x/tem
00:         text:0x68498000 size: 0x7a88
00:         data:0x60082000 dsize: 0x5ef0
00: krtld: file=/kernel/drv/s390x/iwscn
00:         text:0x684a0000 size: 0x1524
00:         data:0x60081830 dsize: 0x4f0
00: boot scratch memory used: 0x471a080
00: Releasing ramdisk and associated pages
00: krtld: file=/kernel/drv/s390x/ib
00:         text:0x684a2000 size: 0x13770
00:         data:0x60088000 dsize: 0x17b8
00: krtld: file=/kernel/misc/s390x/ibdm
00:         text:0x684b6000 size: 0x15610
00:         data:0x6008a000 dsize: 0x1670
00: krtld: file=/kernel/misc/s390x/ibtl
00:         text:0x684cc000 size: 0x187a8
00:         data:0x6008c000 dsize: 0x3fb0
00: krtld: file=/kernel/misc/s390x/ibmf
00:         text:0x684e5000 size: 0x7e3e0
00:         data:0x60090000 dsize: 0x1d160
00: Waiting for CPU 1 to be readied
01: CPU 1 trace table starts at 1842000
01: krtld: file=/kernel/drv/s390x/hci1394
01:         text:0x68564000 size: 0x7f998
01:         data:0x600ae000 dsize: 0x282d8
00: krtld: file=/kernel/exec/s390x/elfexec
00:         text:0x684a2000 size: 0xe9b0
00:         data:0x600d62d8 dsize: 0xaa8
00: krtld: file=/kernel/misc/s390x/s1394
00:         text:0x685e4000 size: 0x78440
00:         data:0x600d7000 dsize: 0x23b90
00: krtld: file=/kernel/drv/s390x/kssl
00:         text:0x6865d000 size: 0xe3c8
00:         data:0x60088000 dsize: 0x13a8
```

When the DIAG250 driver is up that means we can switch from RAMDISK to hardware. When we've done what we've needed with it we can then reclaim the pages it had been using.

At this stage we can fire up the other CPUs we detected earlier. These also have their own trace table.

The kernel is now starting the init process – the elfexec module is used to load elf objects.

```
00: krtld: file=/kernel/strmod/s390x/ldterm
00:         text:0x6866c000 size: 0x132a0
00:         data:0x600fb000 dsize: 0x1338
00: krtld: file=/kernel/strmod/s390x/ttcompat
00:         text:0x684b1000 size: 0x3ab8
00:         data:0x600fab90 dsize: 0x460
00: krtld: file=/kernel/drv/s390x/cn
00:         text:0x6814f280 size: 0xd64
00:         data:0x6007eb78 dsize: 0x3f8
01: krtld: file=/kernel/drv/s390x/ptsl
01:         text:0x68680000 size: 0x215c
01:         data:0x6008b670 dsize: 0x4d8
01: krtld: file=/kernel/drv/s390x/ptc
01:         text:0x68683000 size: 0x2e0c
01:         data:0x6008bb48 dsize: 0x448
01: krtld: file=/kernel/strmod/s390x/ipsecesp
01:         text:0x68686000 size: 0xcc20
01:         data:0x600fd000 dsize: 0x1118
01: krtld: file=/kernel/drv/s390x/ipsecah
01:         text:0x68693000 size: 0x20778
01:         data:0x600ff000 dsize: 0x23e8
01: krtld: file=/kernel/drv/s390x/tl
01:         text:0x686b4000 size: 0xf7e0
01:         data:0x60102000 dsize: 0x1588
01: krtld: file=/kernel/drv/s390x/rts
01:         text:0x68494b18 size: 0x48c
01:         data:0x60103588 dsize: 0x3a0
01: krtld: file=/kernel/drv/s390x/sysmsg
01:         text:0x686c4000 size: 0x1e90
01:         data:0x601013e8 dsize: 0xa30
krtld: file=/kernel/drv/s390x/mm
        text:0x686c6000 size: 0x2758
        data:0x60103928 dsize: 0x648
krtld: file=/usr/kernel/drv/s390x/kstat
        text:0x686c9000 size: 0x1a2c
        data:0x60071cb8 dsize: 0x310
krtld: file=/kernel/sys/s390x/pipe
        text:0x68692c20 size: 0x3dc
        data:0x60038e78 dsize: 0x178
krtld: file=/kernel/fs/s390x/fifofs
        text:0x686cb000 size: 0x7ea8
        data:0x600893a8 dsize: 0xa98
krtld: file=/kernel/sys/s390x/doorfs
        text:0x686d3000 size: 0xa248
        data:0x60104000 dsize: 0xf20
krtld: file=/kernel/fs/s390x/namefs
        text:0x686de000 size: 0x3100
        data:0x600fc338 dsize: 0x7e0
krtld: file=/kernel/sys/s390x/portfs
        text:0x686e2000 size: 0xa630
        data:0x60105000 dsize: 0x1348
krtld: file=/kernel/exec/s390x/intpexec
        text:0x686c37e0 size: 0x7bc
        data:0x60089e40 dsize: 0x1a0
krtld: file=/kernel/drv/s390x/softmac
        text:0x686ed000 size: 0x8540
WARNING: mod_installdrv: no major number for softmac
krtld: file=/kernel/drv/s390x/softmac
        text:0x686ed000 size: 0x8540
        data:0x60109000 dsize: 0x1a70
WARNING: mod_installdrv: no major number for softmac
krtld: file=/kernel/drv/s390x/softmac
        text:0x686ed000 size: 0x8540
```

At this point we move from the “boot PROM” routines for writing to the console to the con3215 driver. Hence we lose the 00: prefix

```
        data:0x60109000 dsize: 0x1a70
WARNING: mod_installdrv: no major number for softmac
krtld: file=/kernel/drv/s390x/devinfo
        text:0x686ed000 size: 0xbce0
        data:0x600ad160 dsize: 0xe88
pseudo-device: devinfo0
devinfo0 is /pseudo/devinfo@0
krtld: file=/kernel/drv/s390x/softmac
        text:0x686f9000 size: 0x8540
krtld: file=/kernel/drv/s390x/sysevent
        data:0x60109000 text:0x68702000
        dsize: 0x1a70 size: 0x1c80
        data:0x60106348 dsize: 0x5b8
WARNING: mod_installdrv: no major number for softmac
krtld: file=/kernel/drv/s390x/softmac
        text:0x68704000 size: 0x8540
        data:0x60109000 dsize: 0x1a70
WARNING: mod_installdrv: no major number for softmac
Hostname: tideusr1
krtld: file=/kernel/drv/s390x/md
        text:0x68704000 size: 0x58220
        data:0x60109000 dsize: 0x79a0
krtld: file=/kernel/strmod/s390x/rpcmod
        text:0x6875d000 size: 0x2f500
        data:0x60111000 dsize: 0x6ea8
krtld: file=/kernel/misc/s390x/tlimod
        text:0x686f9000 size: 0x3778
        data:0x601109a0 dsize: 0x5b0
krtld: file=/kernel/drv/s390x/ib
        text:0x6878d000 size: 0x13770
        data:0x6011a000 dsize: 0x17b8
krtld: file=/kernel/drv/s390x/dca
        text:0x6878d000 size: 0x15710
        data:0x6011a000 dsize: 0x232
krtld: file=/usr/kernel/drv/s390x/dump
        text:0x684e47a8 size: 0x7f0
        data:0x60034d48 dsize: 0x2b8
krtld: file=/usr/kernel/fs/s390x/fdfs
        text:0x686fd000 size: 0x1ab0
        data:0x60106900 dsize: 0x480
krtld: file=/kernel/drv/s390x/zfs"
        text:0x687a0000 size: 0xf9ca0
        data:0x6011d000 dsize: 0x1dee8
krtld: file=/kernel/drv/s390x/random
        text:0x68480690 size: 0x95c
        data:0x60119858 dsize: 0x370
pseudo-device: zfs0
zfs0 is /pseudo/zfs@0
krtld: file=/kernel/drv/s390x/log
        text:0x680571c8 size: 0xe00
        data:0x60119bc8 dsize: 0x408
krtld: file=/kernel/drv/s390x/cryptoadm
        text:0x6889a000 size: 0x2080
        data:0x6013c8ec dsize: 0x520
Sep 24 16:20:15 svc.startd[100004]: svc:/system/resource-mgmt:default: Method
"/
lib/svc/method/svc-resource-mgmt start" failed with exit status 95.
Sep 24 16:20:15 svc.startd[100004]: system/resource-mgmt:default failed
fatally: transitioned to maintenance (see 'svcs -xv' for details)
Sep 24 16:20:15 svc.startd[100004]: svc:/network/ipsec/ipsecalgs:default:
Method "/usr/sbin/ipsecalgs -s" failed with exit status 1.
Sep 24 16:20:16 svc.startd[100004]: svc:/network/ipsec/ipsecalgs:default:
Method "/usr/sbin/ipsecalgs -s" failed with exit status 1.
```

A lot of services below fail because we have no kcfd – this is a closed source piece

```
Sep 24 16:20:16 svc.startd[100004]: svc:/network/ipsec/ipsecalgs:default:
Method "/usr/sbin/ipsecalgs -s" failed with exit status 1.
Sep 24 16:20:16 svc.startd[100004]: network/ipsec/ipsecalgs:default failed:
transitioned to maintenance (see 'svcs -xv' for details)
Reading ZFS config: *
done.
Sep 24 16:20:21 svc.startd[100004]: system/cryptosvc:default failed repeatedly:
transitioned to maintenance (see 'svcs -xv' for details)
Sep 24 16:20:21 svc.startd[100004]: failed to abandon contract 63: Permission
denied
Sep 24 16:20:23 rpcbind: failed to create "/var/run/daemon": Permission denied
Sep 24 16:20:23 svc.startd[100004]: svc:/network/nfs/nlockmgr:default: Method
"/lib/svc/method/nlockmgr" failed with exit status 1.
Sep 24 16:20:24 svc.startd[100004]: svc:/network/nfs/nlockmgr:default: Method
"/lib/svc/method/nlockmgr" failed with exit status 1.
Sep 24 16:20:24 svc.startd[100004]: svc:/network/nfs/nlockmgr:default: Method
"/lib/svc/method/nlockmgr" failed with exit status 1.
Sep 24 16:20:24 svc.startd[100004]: network/nfs/nlockmgr:default failed:
transitioned to maintenance (see 'svcs -xv' for details)
Sep 24 16:20:24 inetd[100246]: Property exec for method inetd_start of instance
svc:/network/rpc/gss:default is invalid
Sep 24 16:20:24 inetd[100246]: Invalid configuration for instance
svc:/network/rpc/gss:default, placing in maintenance
krtld: file=/kernel/sys/s390x/autofs
        text:0x6889d000 size: 0xfea0
        data:0x6013d000 dsize: 0x14c0
krtld: file=/kernel/misc/s390x/rpcsec
        text:0x688ad000 size: 0xa888
        data:0x6013f000 dsize: 0x1368
Sep 24 16:20:25 inetd[100246]: Property exec for method inetd_start of instance
svc:/network/rpc/meta:default is invalid
Sep 24 16:20:25 inetd[100246]: Invalid configuration for instance
svc:/network/rpc/meta:default, placing in maintenance
krtld: file=/kernel/drv/s390x/sy
        text:0x686ec630 size: 0x964
        data:0x6013e4c0 dsize: 0x390
tideusr1 console login:
Sep 24 16:20:26 tideusr1 inetd[100246]: Property
exec for method inetd_start of instance svc:/network/rpc/smserver:default is
invalid
Sep 24 16:20:26 tideusr1 inetd[100246]: Invalid configuration for instance
svc:/network/rpc/smserver:default, placing in maintenance
Sep 24 16:20:26 tideusr1 genunix: krtld: file=/kernel/sys/s390x/pset
Sep 24 16:20:26 tideusr1 genunix:       text:0x688b8000
Sep 24 16:20:26 tideusr1 genunix:  size: 0x25a8
Sep 24 16:20:26 tideusr1 genunix:       data:0x6013e850
Sep 24 16:20:26 tideusr1 genunix:  dsize: 0x480
Sep 24 16:20:26 tideusr1 inetd[100246]: Property exec for method inetd_start of
instance svc:/network/nfs/rquota:default is invalid
Sep 24 16:20:26 tideusr1 inetd[100246]: Invalid configuration for instance
svc:/network/nfs/rquota:default, placing in maintenance
Sep 24 16:20:26 tideusr1 sendmail[100303]: unable to write pid to
/var/run/sendmail.pid: Permission denied
Sep 24 16:20:26 tideusr1 inetd[100246]: Property exec for method inetd_start of
instance svc:/network/rpc/rstat:default is invalid
Sep 24 16:20:26 tideusr1 inetd[100246]: Invalid configuration for instance
svc:/network/rpc/rstat:default, placing in maintenance
Sep 24 16:20:27 tideusr1 inetd[100246]: Property exec for method inetd_start of
instance svc:/network/rpc/rusers:default is invalid
Sep 24 16:20:27 tideusr1 inetd[100246]: Invalid configuration for instance
svc:/network/rpc/rusers:default, placing in maintenance
Sep 24 16:20:27 tideusr1 genunix: krtld: file=/usr/kernel/drv/s390x/sppp
Sep 24 16:20:27 tideusr1 genunix:       text:0x688bb000
```

I can't work out why all the inetd services fail. The property looks fine when displayed using svccfg or looking in the manifest.

Here's our login prompt which means we're ready to work.

```
Sep 24 16:20:27 tideusr1 genunix:  size: 0x9750
Sep 24 16:20:27 tideusr1 genunix:       data:0x60141000
Sep 24 16:20:27 tideusr1 genunix:  dsize: 0x11e8
Sep 24 16:20:27 tideusr1 inetd[100246]: Property exec for method inetd_start of
instance svc:/network/rpc/mdcomm:default is invalid
Sep 24 16:20:27 tideusr1 inetd[100246]: Invalid configuration for instance
svc:/network/rpc/mdcomm:default, placing in maintenance
Sep 24 16:20:27 tideusr1 inetd[100246]: Property exec for method inetd_start of
instance svc:/network/rpc/metamed:default is invalid
Sep 24 16:20:27 tideusr1 inetd[100246]: Invalid configuration for instance
svc:/network/rpc/metamed:default, placing in maintenance
Sep 24 16:20:28 tideusr1 inetd[100246]: Property exec for method inetd_start of
instance svc:/network/rpc/metamh:default is invalid
Sep 24 16:20:28 tideusr1 inetd[100246]: Invalid configuration for instance
svc:/network/rpc/metamh:default, placing in maintenance
Sep 24 16:20:28 tideusr1 inetd[100246]: Property exec for method inetd_start of
instance svc:/network/security/ktkt_warn:default is invalid
Sep 24 16:20:28 tideusr1 inetd[100246]: Invalid configuration for instance
svc:network/security/ktkt_warn:default, placing in maintenance
Sep 24 16:20:28 tideusr1 inetd[100246]: Property exec for method inetd_start of
instance svc:/network/ftp:default is invalid
Sep 24 16:20:28 tideusr1 inetd[100246]: Invalid configuration for instance
svc:/network/ftp:default, placing in maintenance
Sep 24 16:20:29 tideusr1 inetd[100246]: Property exec for method inetd_start of
instance svc:/network/finger:default is invalid
Sep 24 16:20:29 tideusr1 inetd[100246]: Invalid configuration for instance
svc:/network/finger:default, placing in maintenance
Sep 24 16:20:29 tideusr1 inetd[100246]: Property exec for method inetd_start of
instance svc:/network/login:rlogin is invalid
Sep 24 16:20:29 tideusr1 inetd[100246]: Invalid configuration for instance
svc:/network/login:rlogin, placing in maintenance
Sep 24 16:20:30 tideusr1 inetd[100246]: Property exec for method inetd_start of
instance svc:/network/shell:default is invalid
Sep 24 16:20:30 tideusr1 inetd[100246]: Invalid configuration for instance
svc:/network/shell:default, placing in maintenance
Sep 24 16:20:30 tideusr1 inetd[100246]: Property exec for method inetd_start of
instance svc:/network/telnet:default is invalid
Sep 24 16:20:30 tideusr1 inetd[100246]: Invalid configuration for instance
svc:/network/telnet:default, placing in maintenance
Sep 24 16:20:31 tideusr1 inetd[100246]: Property exec for method inetd_start of
instance svc:/network/ftp/tcp:default is invalid
Sep 24 16:20:31 tideusr1 inetd[100246]: Invalid configuration for instance
svc:/
network/ftp/tcp:default, placing in maintenance
Sep 24 16:20:31 tideusr1 inetd[100246]: Property exec for method inetd_start of
instance svc:/network/telnet/tcp:default is invalid
Sep 24 16:20:31 tideusr1 inetd[100246]: Invalid configuration for instance
svc:/network/telnet/tcp:default, placing in maintenance
Sep 24 16:20:31 tideusr1 inetd[100246]: Property exec for method inetd_start of
instance svc:/network/finger/tcp:default is invalid
Sep 24 16:20:31 tideusr1 inetd[100246]: Invalid configuration for instance
svc:/network/finger/tcp:default, placing in maintenance
Sep 24 16:20:33 tideusr1 svc.startd[100004]: system/intrd:default failed
repeatedly: transitioned to maintenance (see 'svcs -xv' for details)
root
Sep 24 16:22:15 tideusr1 genunix: krtld: file=/kernel/exec/s390x/elfexec
Sep 24 16:22:15 tideusr1 genunix:       text:0x68788000
Sep 24 16:22:15 tideusr1 genunix:  size: 0xe9b0
Sep 24 16:22:15 tideusr1 genunix:       data:0x60140368
Sep 24 16:22:15 tideusr1 genunix:  dsize: 0xaa8
Password:
xxxxxxx
```

```
Sep 24 16:22:17 tideusr1 genunix: krtld: file=/kernel/drv/s390x/devinfo
Sep 24 16:22:17 tideusr1 genunix:       text:0x68686000
Sep 24 16:22:17 tideusr1 genunix:  size: 0xbce0
Sep 24 16:22:17 tideusr1 genunix:       data:0x601080f8
Sep 24 16:22:17 tideusr1 genunix:  dsize: 0xe88
Sep 24 16:22:17 tideusr1 genunix: pseudo-device: devinfo0
Sep 24 16:22:17 tideusr1 genunix: devinfo0 is /pseudo/devinfo@0
Sep 24 16:22:17 tideusr1 genunix: krtld: file=/usr/kernel/drv/s390x/pool
Sep 24 16:22:17 tideusr1 genunix:       text:0x6814d000
Sep 24 16:22:17 tideusr1 genunix:  size: 0x1a30
Sep 24 16:22:17 tideusr1 genunix:       data:0x6013ecd0
Sep 24 16:22:17 tideusr1 genunix:  dsize: 0x320
Sep 24 16:22:17 tideusr1 genunix: pseudo-device: pool0
Sep 24 16:22:17 tideusr1 genunix: pool0 is /pseudo/pool@0
Sep 24 16:22:17 tideusr1 login: ROOT LOGIN /dev/console
Last login: Wed Sep 24 14:45:13 from devlab.sinenomi
Sun Microsystems Inc.   SunOS 5.11      home/tide/OpenSolaris/sirius/onnv-gate
Sep. 04, 2008
SunOS Internal Development:  tide 2008-09-04 [onnv-gate]
#
```