Becoming an OpenSolaris Power User

Nicholas Solter and Dave Miner
Authors of *OpenSolaris Bible*

Agenda

- Introduction to OpenSolaris
- Installation and Basic Features
- Security Features
- Service Management
- ZFS
- Software Management
- Network Virtualization
- Zones
- DTrace
- Troubleshooting
- Resources

Community

- OpenSolaris launched June 2005
- Sun/community relationship defined via Charter
- Community structure governed by Constitution
  - Governing Board (OGB), communities, projects, user groups
- Primarily licensed via CDDL
- Sun owns OpenSolaris trademark, others have rights specified by trademark policy

Code

- Aggregated & built by consolidations, e.g.
  - OS/Net (kernel, core user land)
  - Desktop (GNOME)
  - X11
  - SFW (Supported open-source)
- Projects usually integrate code to consolidations
  - Some distros use project code pre-integration, e.g. IPS, Caiman installer

Code, 2

- Two source management systems
  - Subversion - Centralized
  - Mercurial - Distributed
- Architecture (ARC) review helps maintain compatibility and coherence
- Integration to consolidations via sponsor program
- OpenGrok source browser for all hosted code at src.opensolaris.org

Distributions

- Solaris Express Community Edition (SXCE) – The way we were...
- Schillix – First non-Sun distro
- BeleniX – KDE & XFCE Desktops
- Nexenta – OpenSolaris kernel meets Debian/GNU userland
- MilaX – Small footprint distro
- EON – NAS appliance
- Jaris – Japanese community distro

OpenSolaris Distribution

- Re-package Solaris technology into a modern distro
  - Easy to get and install
    - Fully redistributable binaries
  - Easy to update
    - Network package repositories
- Almost all open-source
- First release in May 2008, semi-annual release cadence
- Basis for next Solaris

Agenda

- Introduction to OpenSolaris
- Installation and Basic Features
- Security Features
- Service Management
- ZFS
- Software Management
- Network Virtualization
- Zones
- DTrace
- Troubleshooting
- Resources

Downloading OpenSolaris

- Supported releases available on opensolaris.com, latest 2009.06
- Development releases ~every 2 weeks, via genunix.org or Sun DLC's BitTorrent
- Currently four downloads
  - Live CD
  - Live USB
  - Automated Installation boot images
    - x86
    - SPARC

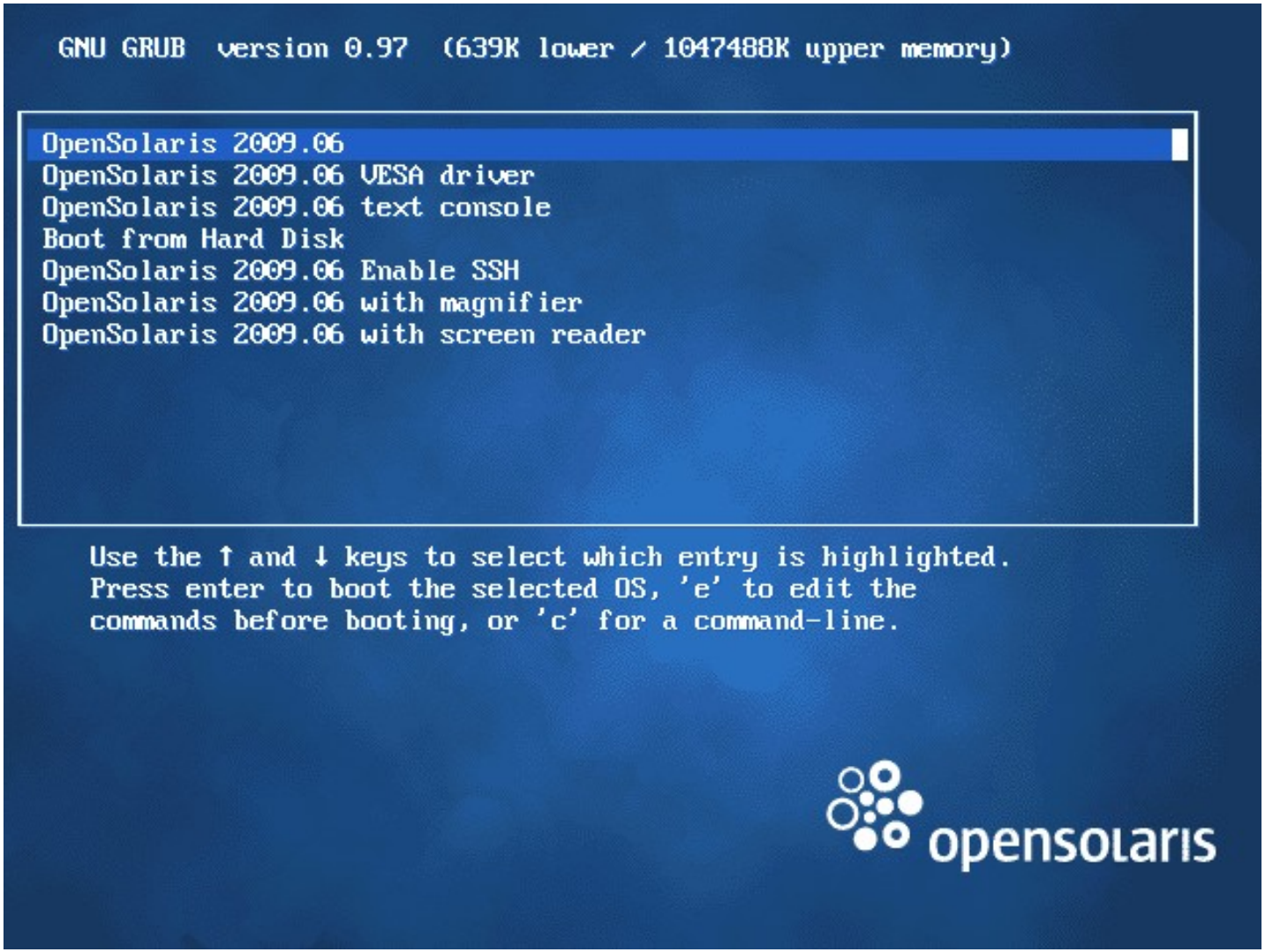
Install Requirements and Preparation

- Pentium III or later, UltraSPARC III or later
- One primary partition, recommended 10 GB
  - Free up space with Vista's Disk Manager or GParted from Linux
- 512 MB memory
- Burn CD or USB
  - USB “burn” using **usbcopy** on Solaris or Windows Live USB creator

Co-existing with Windows or Linux

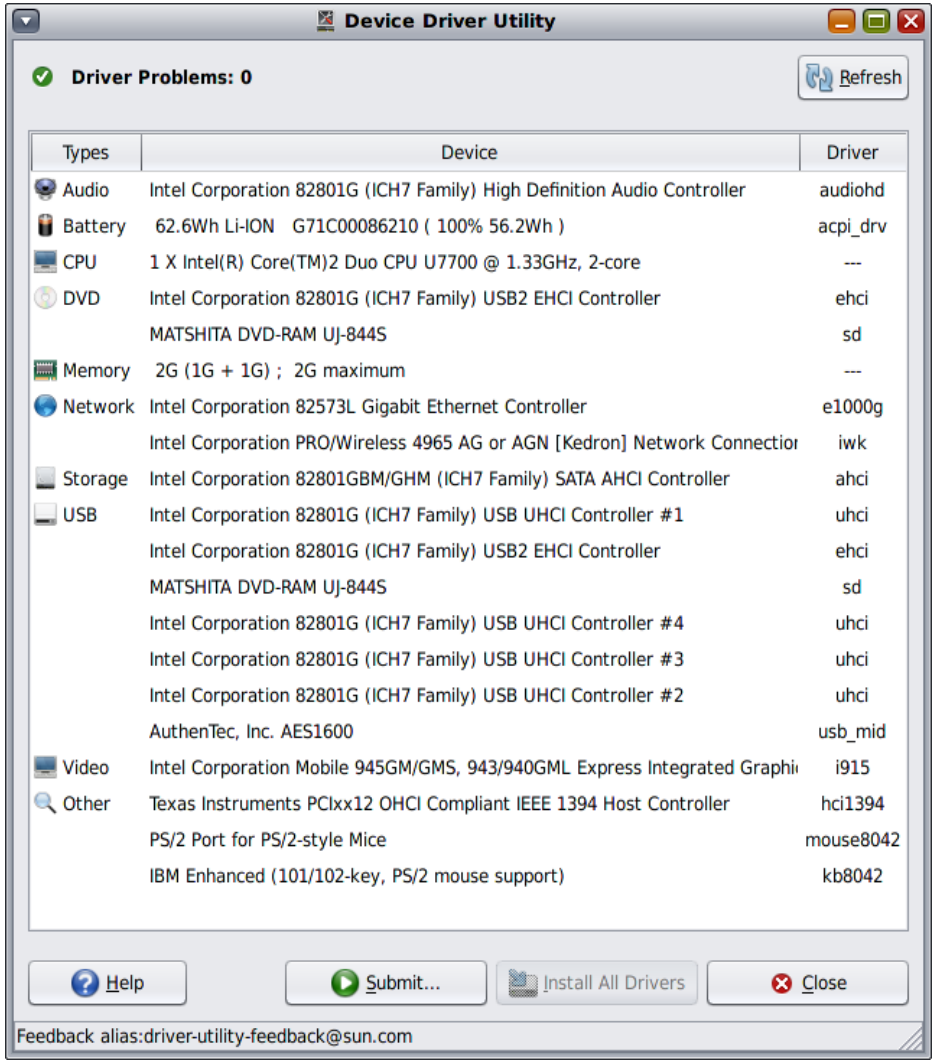
- Best option: virtualize with VirtualBox or VMware
- Otherwise, multi-boot
  - OpenSolaris installer finds Windows partitions, adds chainload to GRUB menu
  - Requires special GRUB version to boot ZFS
  - Two options with Linux
    - Add Linux menu details to OpenSolaris GRUB menu
    - Chainload from OpenSolaris to Linux GRUB

Booting the Live CD



Initial Desktop

Device Support Check with DDU

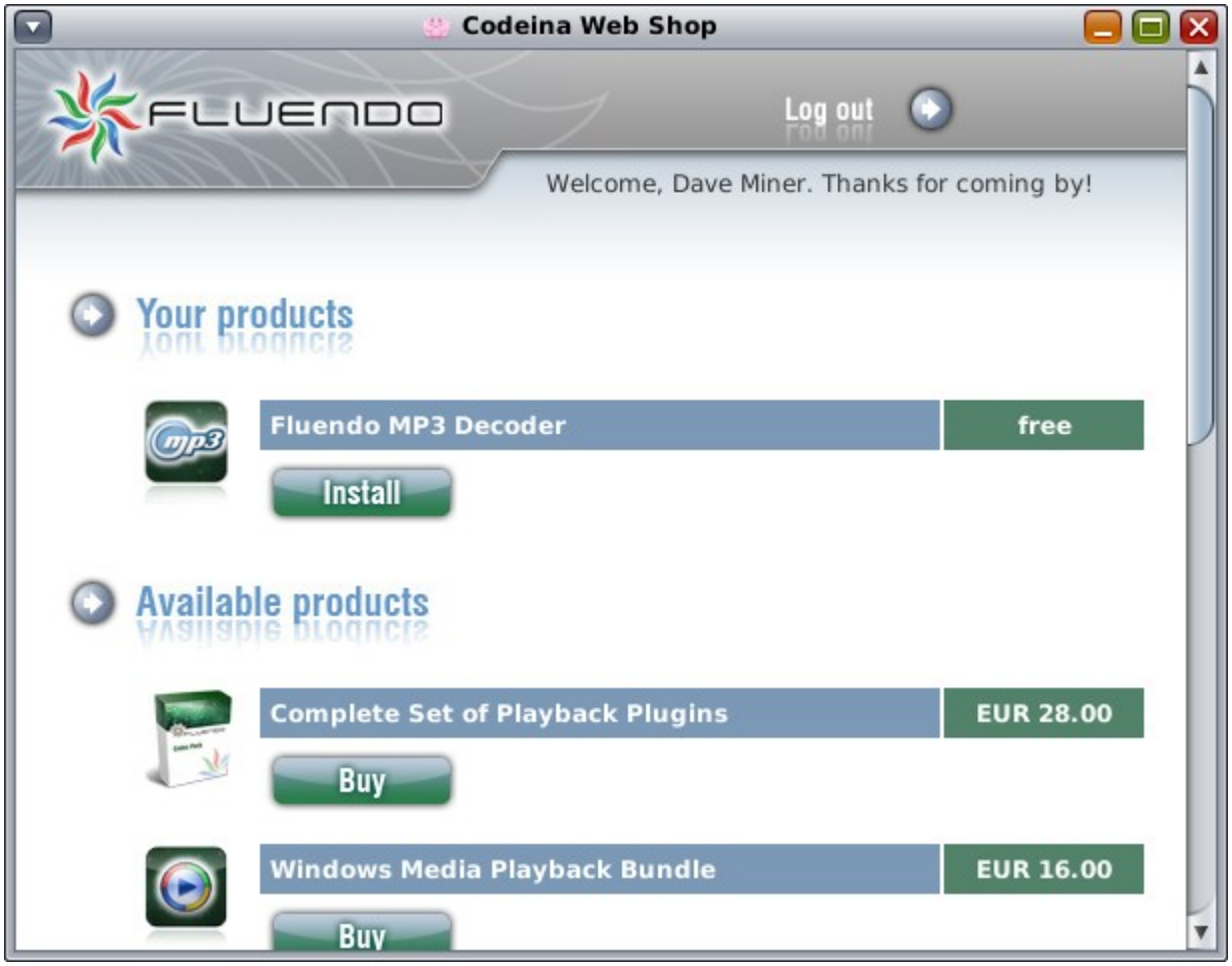


Interactive Installation Demonstration

Basic Features

- Choice of shells
  - bash, ksh93, csh, tcsh, zsh, sh
- Default path prefers GNU utilities via **`/usr/gnu/bin`**
- Replaced **`vi`** with **`vim`**
- Desktop
  - GNOME
  - X11
  - Metacity or Compiz window managers

Multimedia with Codeina

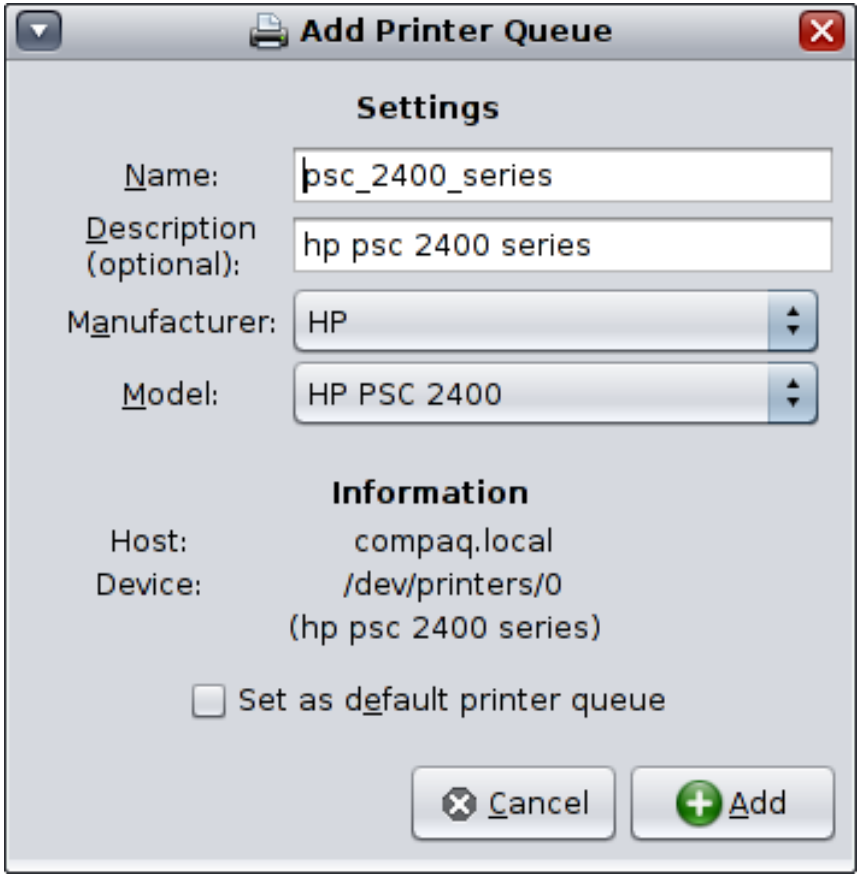


Network Access

- Default to automatic configuration via Network Auto-Magic (NWAM)
  - Just plug in to activate wired network, otherwise wireless is used
  - Select wireless network via system tray applet
  - DHCP automatically starts
- Convert to static IP using Network Manager
  - Manual process detailed in Chapter 9

Printing

- Two subsystems: LP & CUPS
- Choose with **`print-service`**
- “Presto” auto-discovery with LP
- CUPS familiarity to Linux users
- **`ppdmgr`** to install un-packaged PPD's



Power Management

- Suspend-to-ram & resume available
  - Especially pre-installed Toshiba laptops - R600 and M10 are current models
- Power management configured by **`/etc/power.conf`** and **`pmconfig`**
- Power monitoring with **`powertop`**
- Power-aware dispatcher reduces power consumption on Nehalem platforms

Agenda

- Introduction to OpenSolaris
- Installation and Basic Features
- Security Features
- Service Management
- ZFS
- Software Management
- Network Virtualization
- Zones
- DTrace
- Troubleshooting
- Resources

OpenSolaris Security Features

- Secure by Default
  - Most network services off
- **Role-based access control (RBAC)**
  - Similar to sudo (also available)
- IPsec
- Logs and auditing
- Kerberos
- Trusted Extensions

Role-Based Access Control

...Or, what happened to root?

Traditional UNIX Model

- root or superuser
  - All-powerful
- Regular users
  - Everyone else
  - No administrative privileges

Problems with Traditional Model

- Root too powerful
  - Anyone with root access can do unlimited damage
  - Clueless admin can screw up entire system
- No way to delegate administration
  - Multiple people access root
  - Increased opportunities for attackers

Solution: Least Privilege

- Assign users exact authorizations needed
- Role-Based Access Control
  - Native Solaris/OpenSolaris approach
- Sudo
  - Popular on Linux and MacOS
  - Available on OpenSolaris

RBAC Components

- Authorization
  - Fine-grained capability for a specific task
- Rights profile
  - Grouping of authorizations
- Role
  - Like a user
  - Can't log into role directly
  - Assigned authorizations and rights profiles
  - Assigned to users

RBAC in OpenSolaris

- If you create a user in the installer
  - Root is a role
    - Cannot log in directly as root
    - Only users assigned root role can assume it
  - User has “Primary Administrator” and other Profiles and is assigned root role
- To execute privileged commands
  - Prefix with “pfexec”
  - Run from profile shell pfcsh or pfsh
  - su to root

RBAC Demonstration

sudo in OpenSolaris

- Installed by default
  - /usr/bin/sudo
  - /etc/sudoers
- Not configured by default

Sudo Demonstration

Agenda

- Introduction to OpenSolaris
- Installation and Basic Features
- Security Features
- Service Management
- ZFS
- Software Management
- Network Virtualization
- Zones
- DTrace
- Troubleshooting
- Resources

Service Management: Traditional Approach

- Processes started by init scripts
- No concept of services
- No monitoring or restarts of applications
- No grouping of related processes
- No explicit dependencies between services

Service Management Facility (SMF)

- Introduced in Solaris 10
- Concept of s*ervice* groups related processes
- Unified mechanism to start/stop services, Specify dependencies, Specify configuration properties
- Orderly startup/shutdown of system

What is a “Service”?

- Manifest (XML file describing service)
  - Defines dependencies
  - Specifies “method scripts”
  - Defines service properties
- Start method script
- Stop method script
- (Optional) Refresh method script

SMF Administrative Commands

- svcs
  - See state of services
  - “svcs -x” for information about faulted services
- svcadm
  - enable/disable services
- svccfg and svcprop
  - Configure and retrieve service properties

SMF Demonstration

Agenda

- Introduction to OpenSolaris
- Installation and Basic Features
- Security Features
- Service Management
- Storage Devices and ZFS
- Software Management
- Network Virtualization
- Zones
- DTrace
- Troubleshooting
- Resources

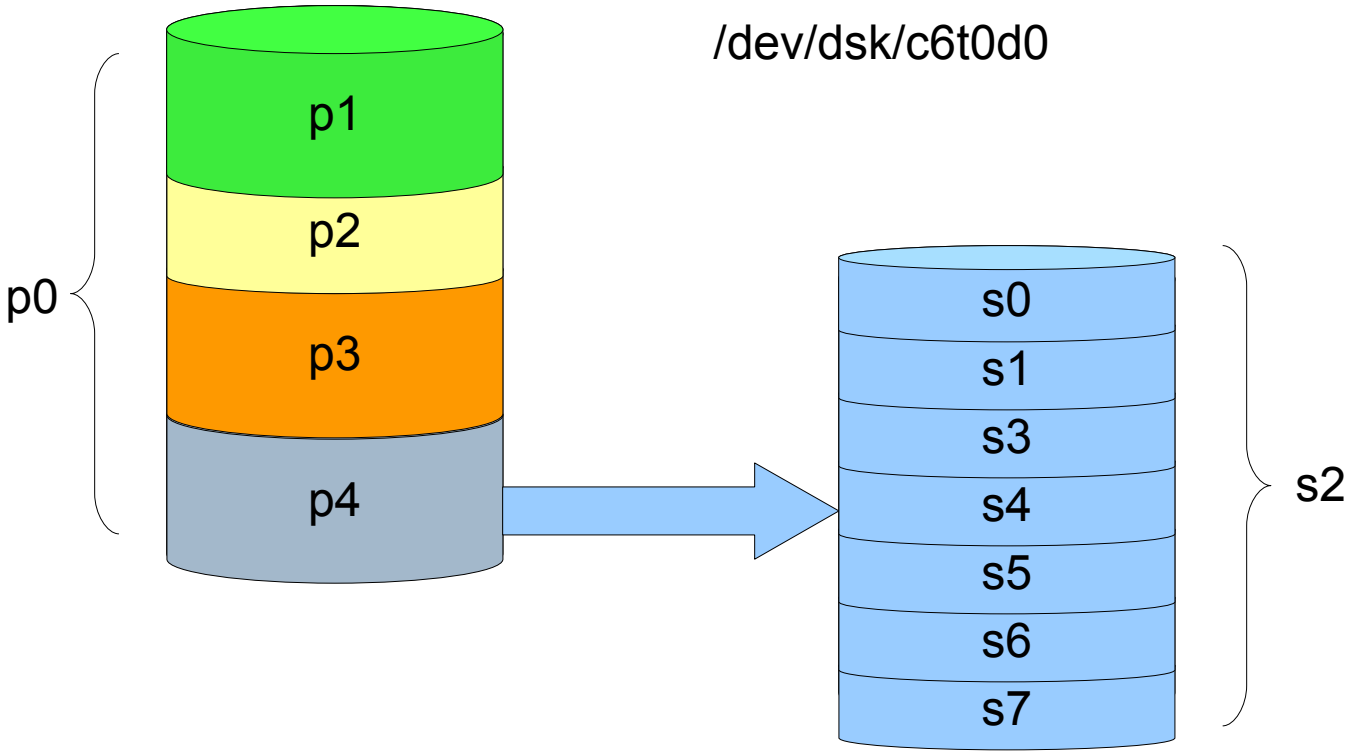
Device Naming, 1

- Controller, Target, Disk, Partition (c#t#d#p#)
  - Used on x86 only, refers to disks
  - Partitions 1-4, 0 = whole disk
  - Device nodes under **`/dev/dsk/`**
    - **`/dev/dsk/c6t0d0p0`** is whole disk

Device Naming, 2

- Controller, Target, Disk, Slice (c#t#d#s#)
  - SPARC disks , and x86 within Solaris partition
  - File systems use these devices
  - Slices numbered 0-7
  - Slice 2 is by convention the entire disk/partition
  - ZFS root pool usually on Slice 0, e.g. **`c6t0d0s0`**

Device Naming, 3



Partitioning Tools

- **`fdisk`** for x86 disk partitioning
  - GParted and **`parted`** in next release
- **`format`**, **`prtvtoc`** for slice management and viewing
- Additional tools for removable media
  - **`rmvolmgr`** – Automatically mounts media on insertion, unmounts on removal
  - **`eject`** – Unmount before removal
  - **`rmformat`** – Viewing and formatting removable media

ZFS

High-end storage for everyone

- Efficiency via pooled model
- High reliability
- Superb scalability – both up *and* down
- Superior ease-of-use
- Flexible architecture evolving with storage technology
  - “Hybrid storage pools” utilize SSD's for acceleration

ZFS Pools

- Managed using **`zpool`** command
- Pool Types
  - Simple – Single-device (disk or partition)
  - Concatenation – Multiple devices without redundancy
  - Mirror – Full duplication across devices aka RAID1
  - RAIDZ – ZFS version of RAID5/RAID6

ZFS Pools, 2

- Performance and Availability Capabilities
  - Spare Devices
  - Log Devices
  - Cache Devices
- Operational history - **`zpool history`**
- Simple performance monitoring – **`zpool iostat`**
- Tip: Use fewer pools for better utilization
- Tip: Be conservative upgrading pool version

ZFS Datasets

- Managed using **`zfs`** command
- Three types
  - File system
  - Volume
  - Snapshot – point-in-time copy of file system or volume
- Easy delegated management – **`zfs allow`**
- Tip: Use file systems liberally
  - They're the primary management boundary!

ZFS Dataset Properties

- Sharing – **`sharenfs, sharesmb, shareiscsi`**
- Reliability and space – **`copies, compression`**
- Security – **`aclinherit, aclmode, devices, exec, setuid`**
- And many more... plus user properties!
- Tip: Use compression, especially with slower disks
- Tip: Copies with mirrored/raidz pools can buy extra reliability for critical data

Demo - RAIDZ Pools

Time Slider

Integrating ZFS with the GNOME desktop

- Automatic snapshots scheduled via cron
- Configurable using Time Slider Manager
- Nautilus file browser extended to browse snapshot history and access snapshot contents

Demo – Time Slider

Agenda

- Introduction to OpenSolaris
- Installation and Basic Features
- Security Features
- Service Management
- ZFS
- Software Management
- Network Virtualization
- Zones
- DTrace
- Troubleshooting
- Resources

Software Management

- We're not in SVR4 anymore!
- Image Packaging System and Package Manager
- Boot Environments and Update Manager
- Running an IPS Repository
- Distribution Constructor
- Automated Installer

IPS Package Basics

- Package is modeled as a set of actions
  - Simple actions: file, directory, link
  - Complex actions: driver, user, group
  - Special actions: depend, set (attributes), license, legacy
- Manifest collects actions to define a package

IPS Images

- Packages install into images - 3 types
  - Full – OpenSolaris instance
  - Partial – OpenSolaris zone
  - User – User-managed software installation
    - Primarily application software
    - Use packages instead of tarballs

IPS Package Distribution

- Packages have a publisher (née authority)
  - Images install packages from one or more publishers
- Publishers provide repositories
  - Switch between repositories from same publisher for different rates of change
  - Tip: switch from /release to /dev for the bleeding edge
- Each repository has a browseable catalog

Software Repositories

pkg.sun.com

.../opensolaris/support/

Specific fixes for
2008.05, 2008.11, ...

.../opensolaris/extra/

Encumbered software
requiring EULA

Registration and

SSL certificate required

pkg.opensolaris.org/

.../dev/
Development builds;
separate products:
redistributable only

.../release/
2008.05, 2008.11, ...

.../contrib/
Community packages

.../webstack/
Project repositories

IPS Package Naming

- General form is pkg://publisher/name@version
  - Uses Solaris FMRI scheme from FMA/SMF
- Names can be hierarchical using / as separator
- Version decomposed as *component_version*,*build_version*-*branch*:*timestamp*

IPS Package Naming, 2

- Usually use abbreviated names; these are all the same (today):
  - **`pkg install flash`**
  - **`pkg install flash@10.0.0.22.87`**
  - **`pkg install pkg://extra.opensolaris.org/web/firefox/plugin/flash@10.0.22.87,5.11-0.111:20090403T194538Z`**
- Tip: **`.p5i`** extension and application/package-manager mime type allow seamless installation links

Demo - Browsing, Searching and Installing Packages

Boot Environments

- Integrate ZFS with software management to take the fear out of installing or updating packages!
- Use ZFS clones to improve performance, reduce resource requirements
- Not the same as Solaris 10 Live Upgrade
- Three states
  - Active
  - Active on Reboot
  - Mounted

Boot Environment Tools

- CLI interface is **`beadm`**
- Delete, rename, activate in Package Manager
- Use **`pkg -R`** / **`packagemanager -R`** to act on mounted boot environments
- Update Manager provides notification of available updates and one-click updates
- No automatic BE pruning
  - Looking at integration with Time Slider

Boot Environment Tips

- BE's are free; use liberally to save you from software or configuration disasters
- Use **`pkg image-update -n`** to dry run an update
- Use **`pkg image-update –be-name <name>`** to name your BE's
- No automatic BE pruning yet
  - Anticipating integration with Time Slider

Demo – BE's & Update Manager

Running an IPS Repository

- Full repository - publish your own packages
  - Use **pkgsend** to populate
- Content mirror
  - Providing improved access to others' packages
  - Clients contact master repository for catalog, download files from your repository
  - See pkg.opensolaris.org mirroring instructions at http://www.opensolaris.org/os/project/pkg/Mirroring/

Running an IPS Repository, 2

- SMF service is **`svc:/application/pkg/server`**
- **`pkg.depotd(1M)`** has configuration details
  - All controllable as SMF service properties
- Repository provides BUI for browsing, searching

Distribution Constructor (DC)

- Build tools for OpenSolaris distribution available to all
  - **`pkg install SUNWdistro-const`**
- XML manifest defines full construction process
  - Sample manifests used for OpenSolaris in **`/usr/share/distro_const/slim_cd`** and .../**`auto_install`**
- Fully integrated with IPS

Distribution Constructor, 2

- Fully extensible – plug your own customization scripts into the build process
- ZFS snapshots for checkpoint/restart
- Pre-building of virtual machines (coming soon)
- **`distro-const(1M)`** for more information
- Use DC to build AI images customized with additional drivers/services

Demo – Distribution Constructor

Automated Installer (AI)

- Easy deployment for SMB environments
  - Get automation up and running in just minutes
- Scalability for enterprises
  - Standard protocols: DHCP, TFTP, PXE, HTTP
- Criteria to match clients with configurations, package sets
- Deep integration with IPS, SMF
  - Install from multiple publishers, use mirrors
  - Configure system using SMF profiles and custom services

Getting Started with AI

- One-stop shopping: **`installadm`**
- Three steps to automation:

```
# pkg install SUNWinstalladm-tools
DOWNLOAD                                PKGS       FILES     XFER (MB)
Completed                             10/10   1128/1128     6.30/6.30

PHASE                                                        ACTIONS
Install Phase                                              1509/1509
PHASE                                                        ITEMS
Reading Existing Index                                       8/8
Indexing Packages                                          10/10
# wget http://dlc.sun.com/osol/opensolaris/2009/06/\
osol-0906-ai-x86.iso
# installadm ...
```

Demo – Automated Installer

Agenda

- Introduction to OpenSolaris
- Installation and Basic Features
- Security Features
- Service Management
- ZFS
- Software Management
- Network Virtualization
- Zones
- DTrace
- Troubleshooting
- Resources

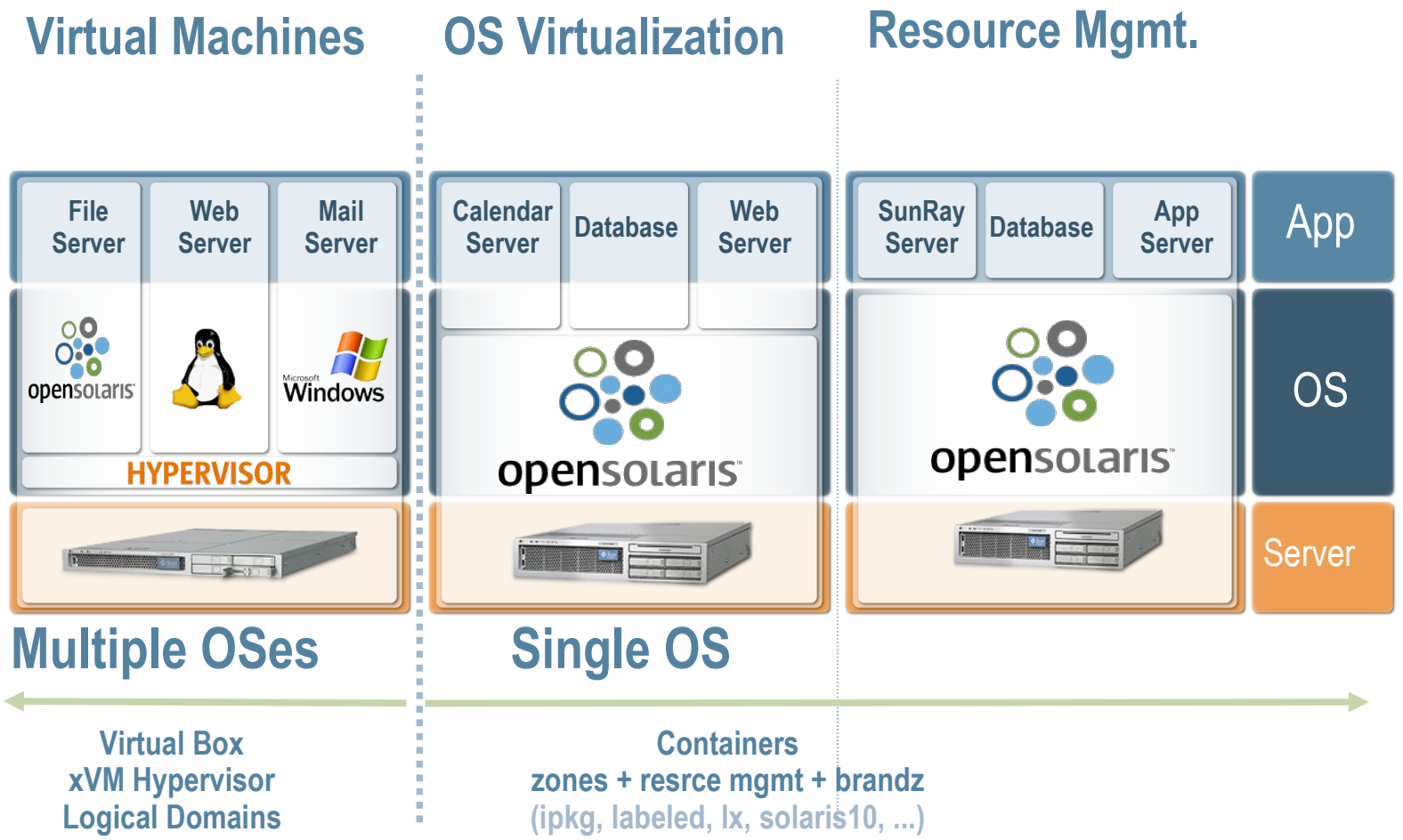
Virtualization

- Provides illusion of exclusive access to shared system resources
- Abstracts the physical resources
- Simplifies consolidation

Virtualization in OpenSolaris

- Resource management
- **Crossbow networking virtualization**
- **ZFS** and COMSTAR **storage virtualization**
- **Operating system virtualization (Zones)**
- Type 1 Hypervisors
  - XVM (Xen) on x86
  - Logical Domains (LDoms) on SPARC
- VirtualBox

OpenSolaris Virtualization Solutions



Crossbow Network Virtualization

- Scalable, virtualized network stack
- Virtual NICs, Virtual Switches, Virtual Wire
- Resource Controls
- Observability
  - real time usage
  - statistics
  - history
- Leveraged by Zones, xVM and LDOMs

Virtual NICs

- Layered on physical NIC or virtual switch
- Appear like physical NICs to higher-level software
  - IP addresses assigned normally
  - Snooped separately
  - Can be assigned to IPMP groups
- Can be assigned flow constraints like physical NICs
- Managed with dladm

Virtual Switches

- Useful for creating private networks between virtual machines or zones
- Called “etherstubs”
- Not layered on anything physical
- Managed with dladm

Flow Control

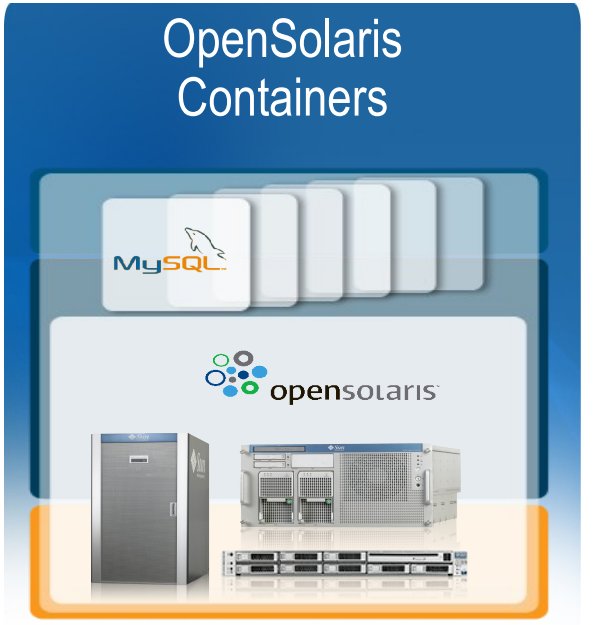
- Network “flows”
  - Can apply to both physical and virtual NICs
  - Set bandwidth limits and priorities
  - Filter based on services, transport, IP addresses and subnets
- Managed with flowadm
- Statistics Captured
  - Requires extended accounting to be activated

Agenda

- Introduction to OpenSolaris
- Installation and Basic Features
- Security Features
- Service Management
- ZFS
- Software Management
- Network Virtualization
- Zones
- DTrace
- Troubleshooting
- Resources

OpenSolaris Zones

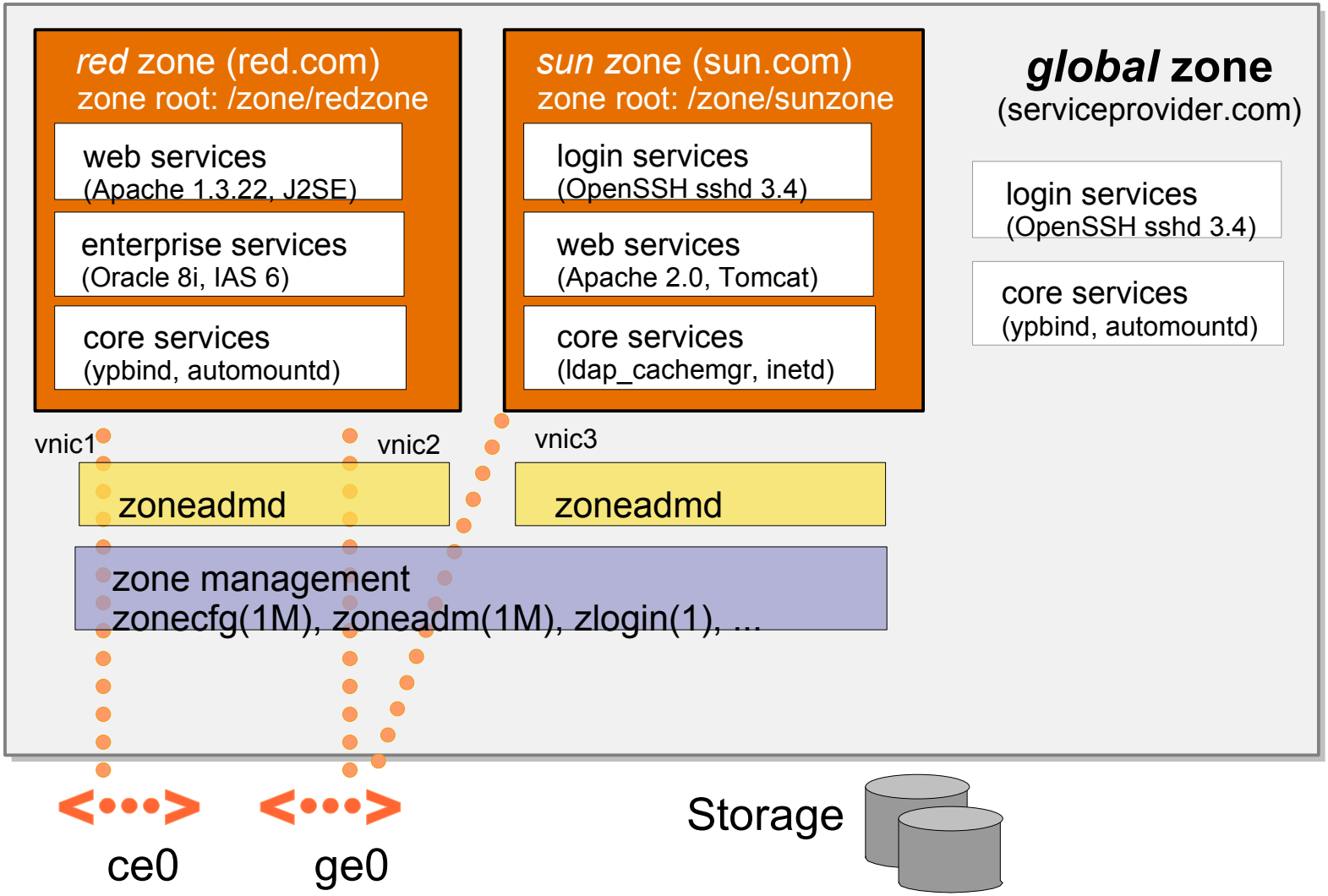
- Operating System level virtualization
  - Secure; Provide isolated environment on top of shared kernel
  - High performance; No virtual machine overhead
  - High scalability



Uses of Zones

- Server consolidation
- Application encapsulation
- Delegated administration

Zones


red zone (red.com)
zone root: /zone/redzone
web services
(Apache 1.3.22, J2SE)
enterprise services
(Oracle 8i, IAS 6)
core services
(ypbind, automountd)
sun zone (sun.com)
zone root: /zone/sunzone
login services
(OpenSSH sshd 3.4)
web services
(Apache 2.0, Tomcat)
core services
(ldap_cachemgr, inetd)
global zone
(serviceprovider.com)
login services
(OpenSSH sshd 3.4)
core services
(ypbind, automountd)
vnic1
vnic2
vnic3
zoneadmd
zoneadmd
zone management
zonecfg(1M), zoneadm(1M), zlogin(1), ...
ce0
ge0
Storage

Zones Configuration and Administration

- zonecfg
  - Zone configuration
- zoneadm
  - Ongoing zones administration
- zlogin
  - Login to a zone

Zone Properties

- Set with zonecfg
- Don't take effect until next boot
- Examples
  - zonepath
  - autoboot
  - bootargs
  - limitpriv
  - ...

Resource Management in Zones

- Resource Control Properties
  - cpu-shares, max-lwps, …
  - Use fair share scheduler
- pools
- dedicated-cpu
- capped-cpu
- capped-memory

Zones Device Management

- Delegate ZFS dataset to zone
  - Use dataset property
- Add additional file systems to zone
  - Use fs property
- Add device access to zone
  - Use device property

Zones Networking

- Shared IP stack
  - NIC shared by global zone and non-global zones
  - Global zone manages networking configuration.
  - IP address can't be changed in zone
- Exclusive IP stack
  - Complete IP software stack dedicated to each zone
  - Zone assigned dedicated NIC
  - Zone manages own IP address and routing

Zones Migration

- detach and attach
- Update on attach

Branded Zones

- Extension of the Zones infrastructure
- Allows creation of non-native zones
  - Makes a zone look like some other system than the global zone
  - Only supports user-land environments
  - Still bound by constraints for running in a zone
- Each distinct zone type is called a Brand
  - lx – Solaris Containers for Linux Applications
  - solaris8, solaris9 (on Solaris 10 only)
  - cluster (for zone clusters across multiple physical machines)

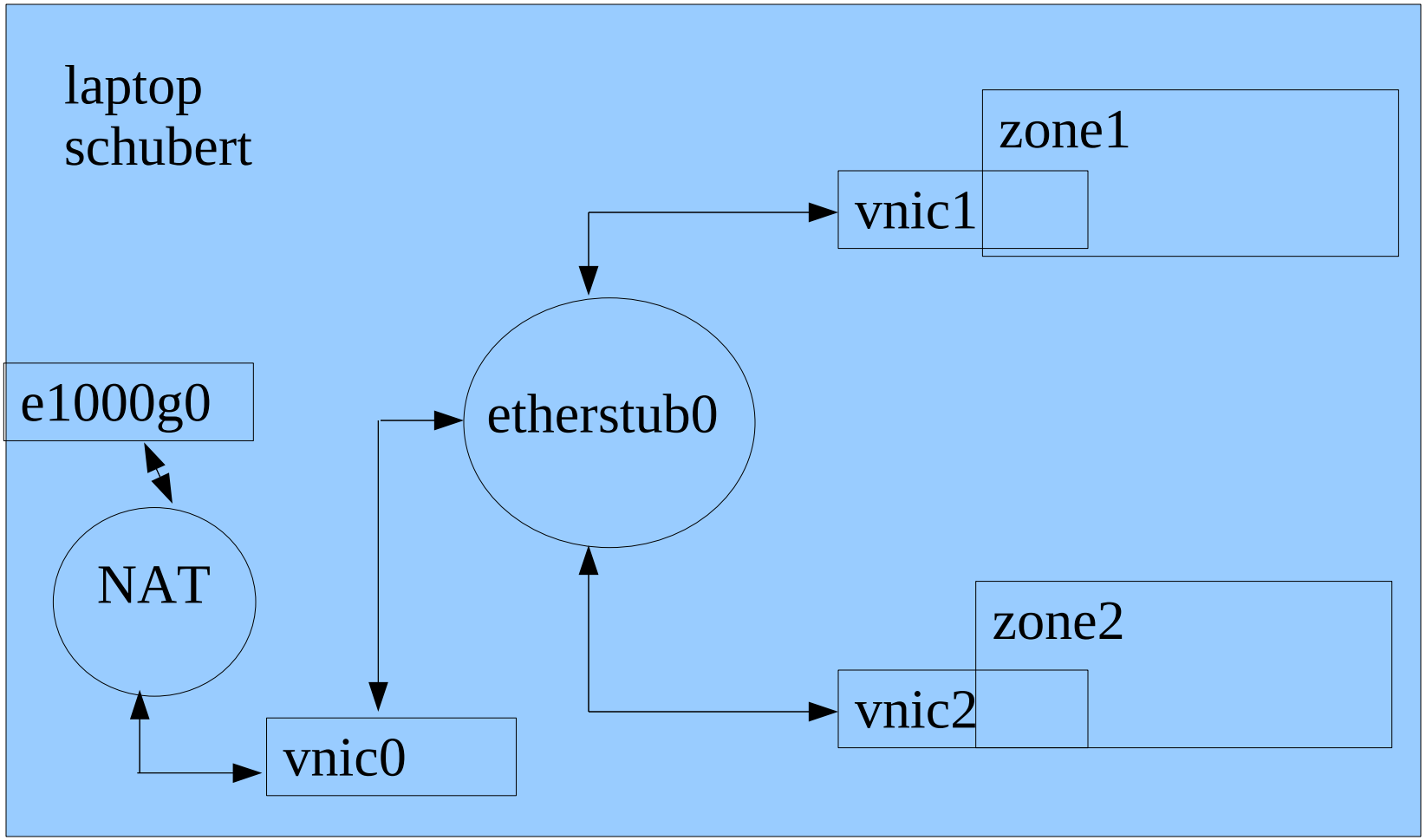
ipkg Brand

- “Native” brand on OpenSolaris
- Software management is different
  - Uses IPS instead of System V
- Whole-root only
- Installs from repository, not global zone
- Newly installed zones are small
  - ~55 packages
  - ~145MB
  - Must add additional packages manually

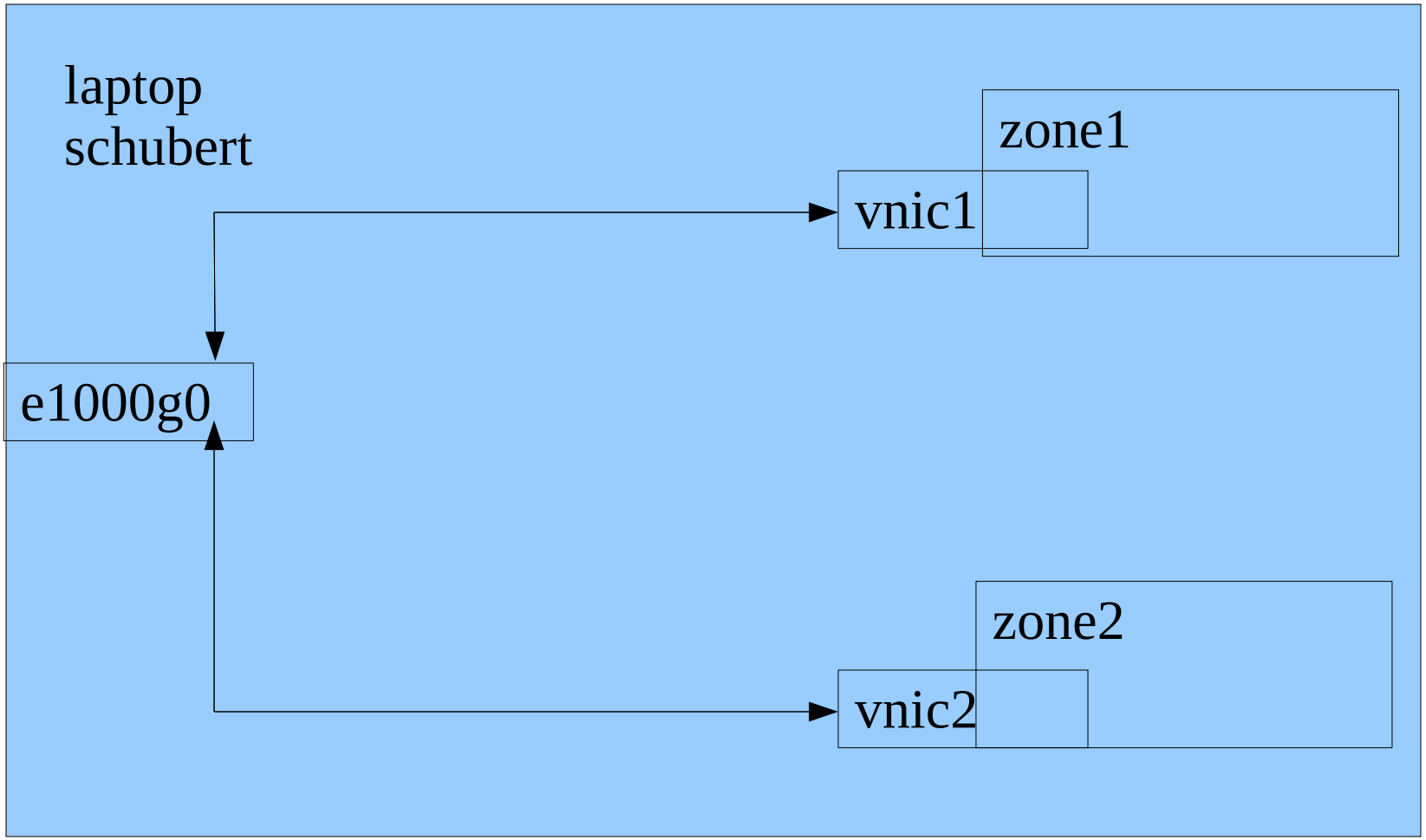
ipkg Brand (continued)

- Zones are integrated with `beadm`
- Zone roots have their own zfs datasets
  - Goal: `beadm` and `image-update` to work inside a zone
  - Active zone dataset mounted when zone boots
- Existing zones from 2008.05 can't be used
- Zones decoupled from the global zone
  - `pkg image-update` not fully integrated yet
  - Use `detach` and `attach -u` as workaround

Diagram of Demo Configuration



Alternative Configuration



Zones and Crossbow Demonstration

Solaris 10 Brand

- Adoption and compatibility aid for OpenSolaris
  - Protect investment in S10 (infrastructure, training, support)
  - Leverage new technology while limiting risk to production environment
  - Avoid required application recertification
- Physical to Virtual (“p2v”) process
  - Also v2v for Solaris 10 zones
- Project at `http://opensolaris.org/os/project/s10brand/`
- Shared stack only at this time

Containers Tradeoffs

- Pros
  - The preferred method for consolidating Solaris-only workloads
  - Runs on any platform supported by OpenSolaris
  - Widely adopted and mature
  - Lowest overhead, best scalability, best observability
  - Solaris 8 and 9 on Solaris 10; Solaris 10 on OpenSolaris
- Cons
  - Support for different operating systems (Linux (dated), Windows (none),...)
  - When different Solaris kernel levels are required
  - When specific restrictions on zones are involved (e.g. NFS server)

Agenda

- Introduction to OpenSolaris
- Installation and Basic Features
- Security Features
- Service Management
- ZFS
- Software Management
- Network Virtualization
- Zones
- DTrace
- Troubleshooting
- Resources

Monitoring and Observability Tools

- System Information
  - uname, prtconf, prtdiag, isainfo,
- Statistics
  - prstat/top, vmstat, mpstat, iostat, etc.
- Proc tools
  - pgrep, pstack, pkill, pfiles, etc.

Monitoring and Observability Demo

What is DTrace?

- Tool to observe and understand dynamic behavior of entire software system
- Fundamentally different from previous tools
  - Software does not have to be instrumented or explicitly built for monitoring
  - Safe for use on live production systems
  - No overhead until *probes* are enabled, and then only minimal overhead
- Includes programming language for expressing variety of tracing behavior

Who Can Benefit from DTrace?

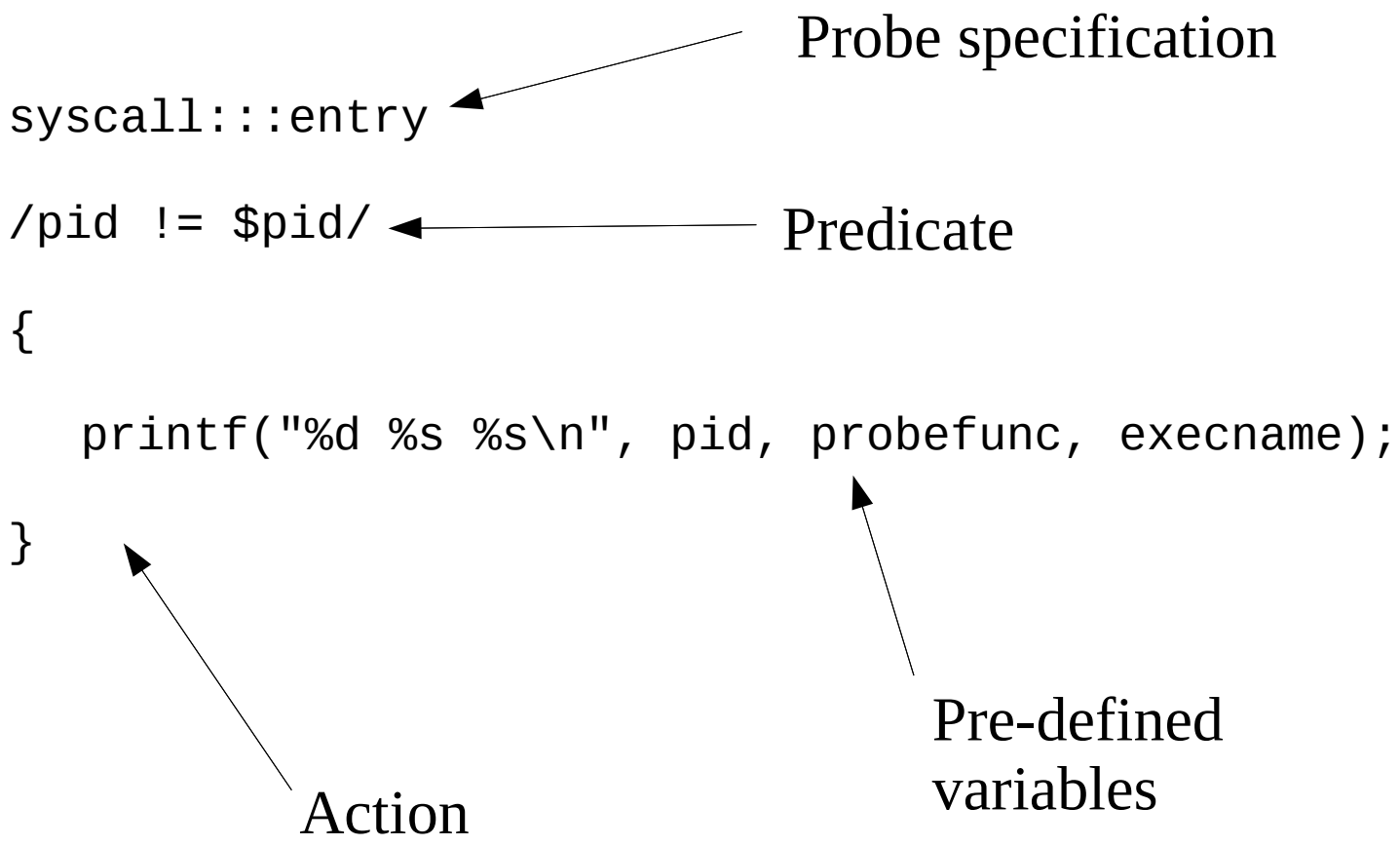
- Administrators
  - Understand the system in order to tune and set policies
  - Debug live problems
- Developers
  - Optimize applications
- Everyone else

What Can You Do with DTrace?

- Determine which process is modifying a file
- Measure the amount of time spent in each function by an application
- Determine which application is spending the most time on the cpu
- Keystroke logging (not necessarily recommended...)
- In summary, answer pretty much any question you have about anything on your system

Simple Example

Trace all system calls



Run syscall Example

```
# dtrace -q -s syscall.d
718 write firefox-bin
718 lwp_park firefox-bin
718 read firefox-bin
718 ioctl firefox-bin
718 pollsys firefox-bin
718 ioctl firefox-bin
718 pollsys firefox-bin
718 ioctl firefox-bin
718 pollsys firefox-bin
589 ioctl gnome-volume-man
589 pollsys gnome-volume-man
624 portfs clock-applet
624 ioctl clock-applet
624 pollsys clock-applet
585 portfs nautilus
585 ioctl nautilus
...
```

More Complex Example

Determine time each process is spending on CPU

```
sched:::on-cpu
/pid != $pid/
{
   self->t = timestamp;
}

sched:::off-cpu
/self->t/
{
   @cputime[execname] = sum(timestamp - self->t);
   self->t = 0;
}
```

Thread-specific variable

Associative array

Aggregation

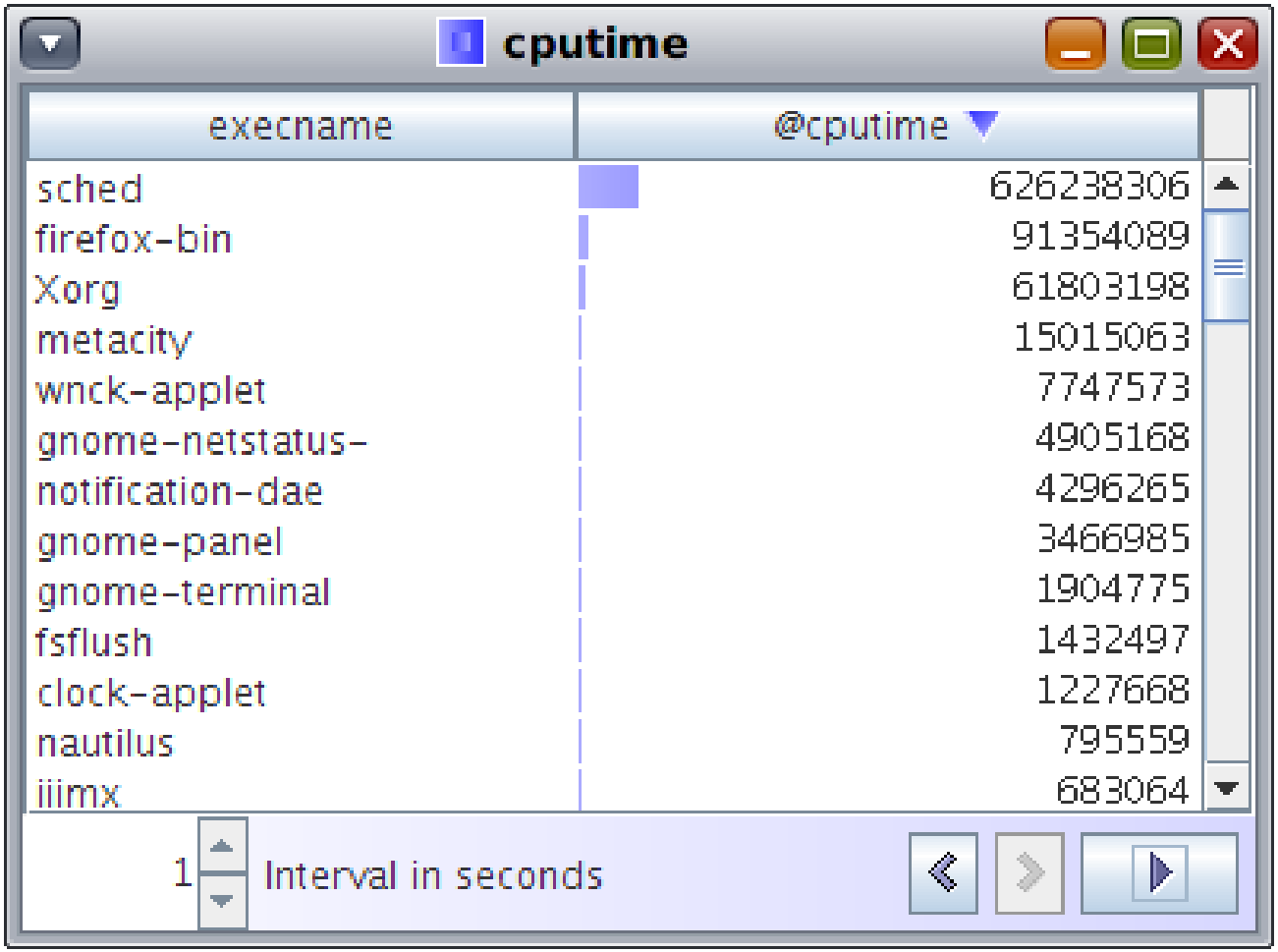
Run CPU-time Example

```
# dtrace -q -s cputime.d
^C
  metacity                                                 98
  devfsadm                                              168109
  mdnsd                                                 433063
  dhcpagent                                             550435
  sendmail                                              767811
  dlmgmtd                                              1455941
  gam_server                                           2398009
  xscreensaver                                         2578289
  gnome-settings-d                                     3728513
  updatemanagernot                                     4146873
  ...
  nautilus                                             7418007
  fsflush                                              7938174
  Xorg                                                 9267514
  gnome-terminal                                      12227762
  gnome-panel                                         13701543
  gnome-netstatus-                                    19195334
  firefox-bin                                         69696567
  sched                                             2720718306
```

The DTrace Toolkit

- Many useful DTrace scripts provided for you
  - dtruss
  - xvmstat
  - opensnoop
  - And many, many more...
- In /opt/DTT/
  - /opt/DTT/Bin has symlinks to all of them

Chime for visualising DTrace output



Chime Installation

- Download package from
  http://opensolaris.org/os/project/dtrace-chime/install/
- Install Package
  ```
  # gunzip osol0chime-i386-1.4.pkg.gz
  # pkgadd -d osol0chime-i386-1.4.pkg
  ```
- Launch
  ```
  # /opt/OSOL0chime/bin/chime
  ```
- Use with DTrace Toolkit built-ins or add your own displays
  - Intuitive and easy to use

DTrace demo

Agenda

- Introduction to OpenSolaris
- Installation and Basic Features
- Security Features
- Service Management
- ZFS
- Software Management
- Network Virtualization
- Zones
- DTrace
- Troubleshooting
- Resources

If OpenSolaris Won't Boot...

- Symptom
  - Graphical boot screen just sits there
- Solution
  - Press any key to switch to text mode

If OpenSolaris Won't Boot...

- Symptoms
  - Reset back to GRUB menu
  - Hang after displaying initial banner
  - Hang at **`Configuring devices`**
- Solutions
  - Verbose boot: add **`-v`** to kernel options
  - Disable problematic driver: add **`-B disable-`*`driver`*`=true`**
  - Enable **`kmdb`**: add **`-k`** to kernel options

If OpenSolaris Won't Boot...

- Symptom
  - Never get to console login prompt
- Solution
  - Boot single-user (aka System Maintenance Mode): add **`-s`** to kernel options, login when prompted
  - Examine system log **`/var/adm/messages`**
  - Check service state with **`svcs -x`**

If The Desktop Won't Start...

- Symptom
  - No graphical login screen
- Solution
  - Ensure GNOME Display Manager (gdm) is enabled: **`svcadm enable gdm`**
  - Examine X log, **`/var/log/Xorg.0.log`**
  - Previous X log is **`/var/log/Xorg.0.log.old`**

If The Desktop Won't Start...

- Symptom
  - Keeps returning to login screen
- Solution
  - Select xterm failsafe session in login screen, then login
  - Examine X and gdm session logs
    - `~/.xsession-errors`
    - `/var/log/gdm/*`

System Crash Dumps

- Use **`dumpadm`** to manage dump configuration
- Force a crash dump with **`halt -d`**
- Use **`savecore`** to pull dump data from dump device

```
# zfs create -o mountpoint=/crash rpool/crash
# dumpadm -s /crash -y
      Dump content: kernel pages
       Dump device: /dev/zvol/dsk/rpool/dump (dedicated)
Savecore directory: /crash
  Savecore enabled: yes
```

Process Core Dumps

- Use **`coreadm`** to manage core dump configuration
- Generate process core dumps using **`gcore`**

```
# zfs create -o mountpoint=/cores rpool/cores
# coreadm -g /cores/%f.%p -e global -d process
```

Network Troubleshooting Tools

- Link status using **`dladm show-link`**
- WiFi status using **`dladm show-wifi`**
- Routing, interface, socket stats via **`netstat`**
- Basic connectivity checks with **`ping`** and **`traceroute`**
- Nameservice checks with **`dig`** and **`getent`**
- Packet sniffing with **`snoop`**
  - Development builds now have **`wireshark`**

Agenda

- Introduction to OpenSolaris
- Installation and Basic Features
- Security Features
- Service Management
- ZFS
- Software Management
- Network Virtualization
- Zones
- DTrace
- Troubleshooting
- Resources

Mailing lists

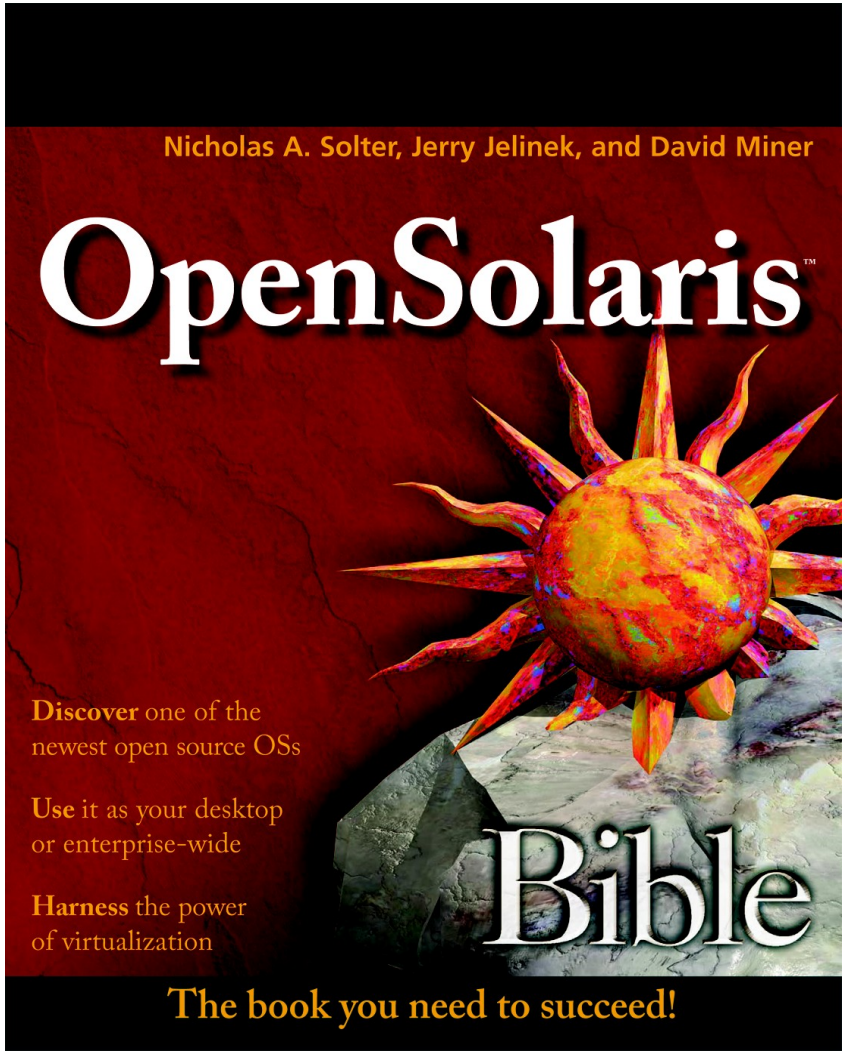
- For general help, opensolaris-help@opensolaris.org
- For announcments of releases, new projects, other important community events: opensolaris-announce@opensolaris.org
- Governance of the community: ogb-discuss@opensolaris.org
- Spreading the OpenSolaris word: advocacy-discuss@opensolaris.org

Web sites

- opensolaris.com – Release downloads, documentation, support
- opensolaris.org – Community portal
- genunix.org – Community wiki, other resources
- BigAdmin – Sun's portal for administrators

Books

- Solaris Internals (McDougall and Mauro; Prentice Hall, 2007)
- Solaris Performance and Tools (McDougall, Mauro, Gregg; Prentice Hall, 2007)
- Pro OpenSolaris (Foxwell and Tran; Apress, 2009)
- Solaris Application Programming (Gove; Prentice Hall, 2008)
- OpenSolaris Bible (Solter, Jelinek, Miner; Wiley, 2009)


Nicholas A. Solter, Jerry Jelinek, and David Miner
OpenSolaris™
Discover one of the newest open source OSs
Use it as your desktop or enterprise-wide
Harness the power of virtualization
Bible
The book you need to succeed!

Becoming an OpenSolaris Power User

Nicholas Solter
http://blogs.sun.com/nsolter
nicholas.solter@sun.com
Dave Miner
http://blogs.sun.com/dminer
dave.miner@sun.com