



opensolaris

Developer's Startup Guide

Aubrey Li

Intel OpenSolaris Team

Enter OpenSolaris 200805

- **OpenSolaris reference binary distribution**
- **Sun and community jointly built**
- **Single CD install**
- **Role-Based Access Control (RBAC)**
- **New network package repository**
- **100% re-distributable**
- **Live CD/DVD/USB functionality**
- **ZFS as default root file-system**
- **More intuitive update experience with ZFS rollback functionality**

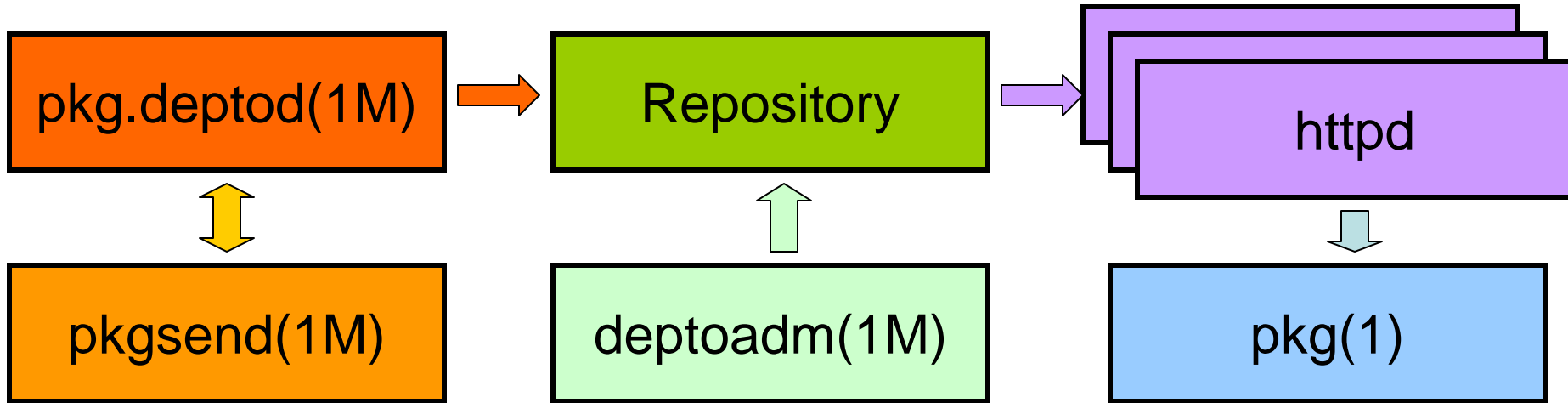
Indiana **VS** SXCE/SXDE

	Indiana	SXCE/SXDE
ON bits	the same source tree	the same source tree
Distribution	Live CD	Installation DVD
Packaging	IPS & SVR4	SVR4
Upgrading	pkg image-update	BFU
File system	ZFS	UFS

Image Packaging System (IPS)

- Network-centric packaging system written in Python
- No scripting in packages - exhaustive metadata
- No build system in packaging - only lays out binary bits
- Dynamic package re-factoring
- Support concept of filesystem images
- Images can be linked and updated in parallel
- Per-user images
- ZFS aware - snapshots and rollback

IPS Components



Package developer

- create, update, remove package and group versions

Depot operator

- close versions, remove stale versions and files

Customer

- Install, update and delete local packages

IPS Repositories

- **<http://pkg.opensolaris.org>**
 - set as default IPS authority
 - 4460~ packages
- **<http://blastwave.network.org:10000>**
 - 1700~ packages
- **<http://pkg.sunfreeware.com:9000>**
 - 290~ packages
- **SVR4 packages are still supported**

System Installation and configuration

- **System requirements - minimum 512MB RAM, recommended 1GB disk**

- **Live CD account:**

- ✓ username – *jack*, password – *jack*
- ✓ su, password - *opensolaris*

- **Simple installation, no separate partition, no custom software components selection**

- **Check and change NWAM configuration**

```
$ svcs svc:/network/physical
STATE      STIME    FMRI
disabled   18:19:05 svc:/network/physical:default
online     19:42:22 svc:/network/physical:nwam
```

- **Check and change personal environment variables**

```
$ cat $HOME/.bashrc
```

- **Need root login?**

```
# rolemod -K type=normal root
```

- **Need console? for mdb -K maybe ?**

```
# svcadm disable gdm (disable x-windows)
```

Install packages for development environment

- **Check authority repository**

\$ pkg authority

AUTHORITY

URL

opensolaris.org (preferred)

<http://pkg.opensolaris.org:80>

- **Add a new repository**

pkg set-authority -O http://blastwave.network.com:10000 blastwave.org

- **Set pkg client timeout environment variable if necessary**

\$ export PKG_CLIENT_TIMEOUT=90

- **Update catalogs**

\$ pfexec pkg refresh

- **Install Mercurial (hg)**

\$ pfexec pkg install SUNWmercurial

- **Install SUN gcc, automatically including binutils, head files, etc**

\$ pfexec pkg install SUNWgcc

- **Install sunstudio express (sunstudio 12) if necessary**

\$ pfexec pkg install sunstudioexpress

Build ON

- **Download ON source**

\$ hg clone ssh://anon@hg.opensolaris.org/hg/onnv/onnv-gate

- **Download and install sunstudio**

- **Download and install ON specific build tools (onbld)**

- **speedup build process (optional)**

- `export CW_NO_SHADOW=1`

- In `$(SRC)/uts/i86pc/Makefile.i86pc.shared` and `$(SRC)/uts/intel/Makefile.intel.shared`, if you just need 64-bit target

- `DEF_BUILDS = $(DEF_BUILDS64) $(DEF_BUILDS32)`

- `ALL_BUILDS = $(ALL_BUILDS64) $(ALL_BUILDS32)`

- In `$(SRC)/uts/Makefile`, remove `i86xpv` token if you don't need it

- **change PATH to use UNIX utilities instead of GNU utilities in case of improper libdtracestubs.so created**

\$export PATH=/usr/ccs/bin:\$PATH (to use UNIX nm, not GNU nm)

- **Just build it**

Using Cap-eye-install to Install Kernels

- Check and change PATH to use UNIX dirname instead of GNU dirname

```
$ export PATH=/usr/bin:$PATH
```

- create kernel tar file

```
$ Install -G kernel.mine -k i86pc
```

- Extract the tar file to the root directory

- Add one entry to `"/rpool/boot/grub/menu.lst"` to point to your own kernel bits

- issue `"zpool status"` to get the root slice name, like `"c4d0s0"`

- Just the first time, re-install grub to make your kernel known ZFS is the root file-system, not UFS. (Don't need to do this second time)

```
#installgrub /boot/grub/stage1 /boot/grub/stage2 /dev/rdisk/c4d0s0
```

- Everything should be okay now, let's reboot

```
#reboot
```

BFUing to your own kernel bits

- **nightly build your ON bits**
- **Before use BFU to upgrade your system, create a new BE is recommended, it gives you a backup BE that you can fall back to in case the BFU fails**

\$ man beadm

- **Upgrade your system by BFU archives**
- **Re-install grub to make your kernel known ZFS is the root file-system, not UFS**
- **Everything should be okay now, just reboot**

#reboot

SNV_90 is available for Indiana

- **set PKG_CLIENT_TIMEOUT=200 first, since image is too big**

```
DOWNLOAD          PKGS    FILES  XFER (MB)
SUNWroute          401/533 8428/13882 825.02/1172.63
```

- **Upgrade to SNV_90**

```
$ pfexec pkg image-update
```

- **Due to changes in grub boot system change, MBR needs to be updated**

```
$ beadm list
```

<i>BE</i>	<i>Active</i>	<i>Active on</i>	<i>Mountpoint</i>	<i>Space</i>
<i>Name</i>		<i>reboot</i>		<i>Used</i>
----	-----	-----	-----	-----
<i>opensolaris-1</i>	<i>no</i>	<i>yes</i>	<i>-</i>	<i>17.06M</i>
<i>opensolaris</i>	<i>yes</i>	<i>no</i>	<i>-</i>	<i>33.92M</i>

```
$ pfexec mount -F zfs rpool/ROOT/opensolaris-1 /mnt
```

```
$ pfexec /mnt/boot/solaris/bin/update_grub -R /mnt
```

- **Reboot to enjoy SNV_90**

```
# reboot
```

Troubleshoot System Login

- **Option 1: Grub option “-s” still works**
- **Option 2:**
 - Boot up system by Live CD
 - *zpool import*, to get the existing zfs pool (pool name, device slice, etc)
 - *zpool import -f <zfs pool name>*
 - Now the existing zfs pool is mounted, try to recover everything needed
 - *reboot*

System backup and recovery (zfs send/receive)

- **System Backup :**

- `zfs snapshot -r rpool@backup`

- hook USB driver, get its device name by *format*, say `/dev/dsk/c4t0s0`

- `zpool create bkpool /dev/dsk/c4t0s0`

- `zfs send -R rpool@backup > /bkpool/backup_file`

- **System Recovery :**

- startup system by Live CD

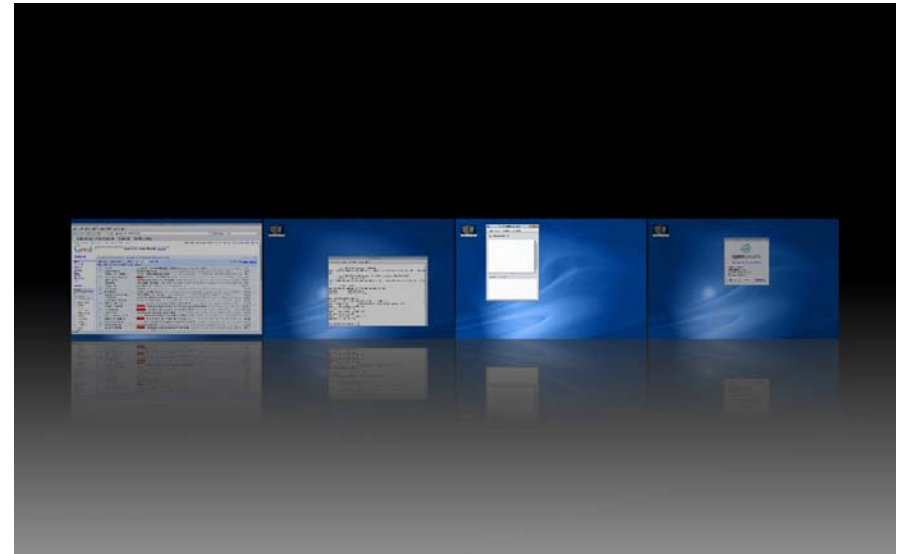
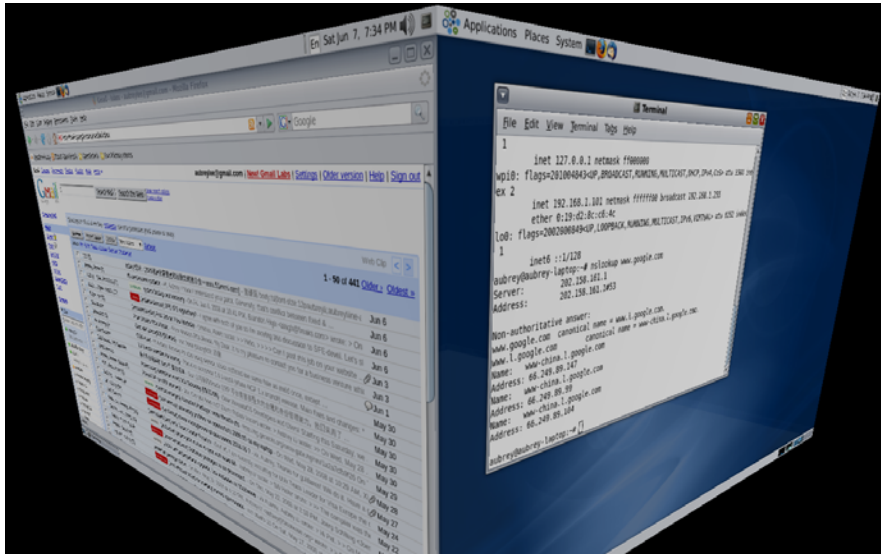
- Setup zfs pool and filesystem, say rpool on `/dev/dsk/c1t0s0`

- hook USB driver, zfs import it at `/bkpool`

- `cat /bkpool/backup_file | zfs receive -dF rpool`

3D desktop enabling

- Indiana makes compiz enabling too easy, if your video card is supported
 - [Desktop Background] -> [Visual Effects], change [None] to [Normal]



How Do I Get Involved?

● Indiana Core

- indiana-discuss@opensolaris.org
- <http://www.opensolaris.org/os/project/indiana/>

● Caiman Installer, Slim Install project

- caiman-discuss@opensolaris.org
- <http://www.opensolaris.org/os/project/caiman/>
- http://www.opensolaris.org/os/project/caiman/slim_install/

● New Packaging

- pkg-discuss@opensolaris.org
- <http://www.opensolaris.org/os/project/pkg/>

● Distro Construtor

- <http://www.opensolaris.org/os/project/caiman/Constructor/>



opensolaris

Thank you!

Aubrey Li

Intel OpenSolaris Team