

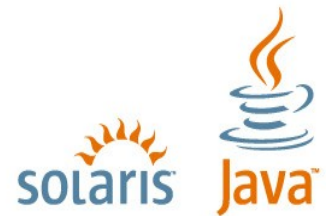


Building and Deploying **open**SOLARIS™

Wyllys Ingersoll

UNLOCK
OPPORTUNITY

What will you open?



SUN TECH DAYS 2006-2007
A Worldwide Developer Conference

What is OpenSolaris?

- Open development effort based on the source code for the Solaris Operating System
- Goals: innovation, collaboration and the extension of OpenSolaris technology.
- Collection of source bases (*consolidations*) and projects
- Developer communities, coordinated via the opensolaris.org infrastructure
- **Not an installable binary image or product**

What is available as open source ?

- OS/Net (“ON”)
 - > Operating system and Networking, the basic, fully-functional system support
- DevPro (partially)
 - > Development tools
- JDS
 - > GNOME-based desktop environment
- NWS
 - > Network Storage drivers, libraries, utilities
- G11N
 - > Message files for ON
- OpenGrok
 - > “wicked fast” source browser

What is available as open source (2)?

- SFW/CCD
 - > 3rd party open source software
- X11
 - > X Window system
- Packaging tools
 - > SVR4-style packaging tools
- Documentation
 - > Developer and administration technical documentation
 - > Source for several books available
- More coming (see the roadmap at <http://opensolaris.org/os/about/roadmap>)

OpenSolaris development structure

- `opensolaris.org`
 - Central site for members, projects, source
- Communities
 - Social groups
 - Mailing list discussions
 - Representation in OpenSolaris governance
- Projects
 - Collaborative efforts
 - Source code (repositories)
 - By-country localizations of `opensolaris.org`

OpenSolaris project examples

- DTrace
 - > Open sourced before OpenSolaris itself, even
 - > Has been ported to other systems
- ZFS
- IPSEC tunneling reform
- SCM Migration
- Xen virtualization support
- BrandZ Linux containers/zones
- ..and many more!

OpenSolaris Distributions

- Binary plus source distributions, based on the OpenSolaris source (compare: Red Hat, Ubuntu, etc. for Linux)
- Anyone can start their own if they want to
 - > With their own extensions and modifications
- Currently 4 non-Sun distributions, plus 3 from Sun

OpenSolaris Distribution: Solaris Express Developer Edition (SXDE)

- A development release of Solaris with limited support available
- Offers early view of new OpenSolaris and/or Solaris features
- Based on OpenSolaris plus the **closed parts of Solaris**
- Aimed at customers who want to try out the latest technology with some basic level of support
- Updated quarterly

OpenSolaris Distribution: Solaris Express Community Edition (SXCE)

- Unsupported release of Solaris Express
- Currently the only distribution that is guaranteed to build OpenSolaris
- Based on the current OpenSolaris sources, and the closed parts of Solaris
- Updated biweekly
- As bleeding edge as it comes

OpenSolaris Distribution: Nexenta

- Also known as GNU/OpenSolaris
- OpenSolaris kernel and runtime
- GNU commands, tools and utilities, and an Ubuntu Userland
- Debian package system

OpenSolaris Distribution: SchilliX

- First external OpenSolaris distribution
- Based on the ON consolidation
- Some more drivers + Schily tools
- Assembly kit to create bootable, “live” media

OpenSolaris Distribution: BeleniX

- Based on ON consolidation
- Adds more open source packages
- LiveCD-based distribution
 - > Updating OpenSolaris support for LiveCD-style systems

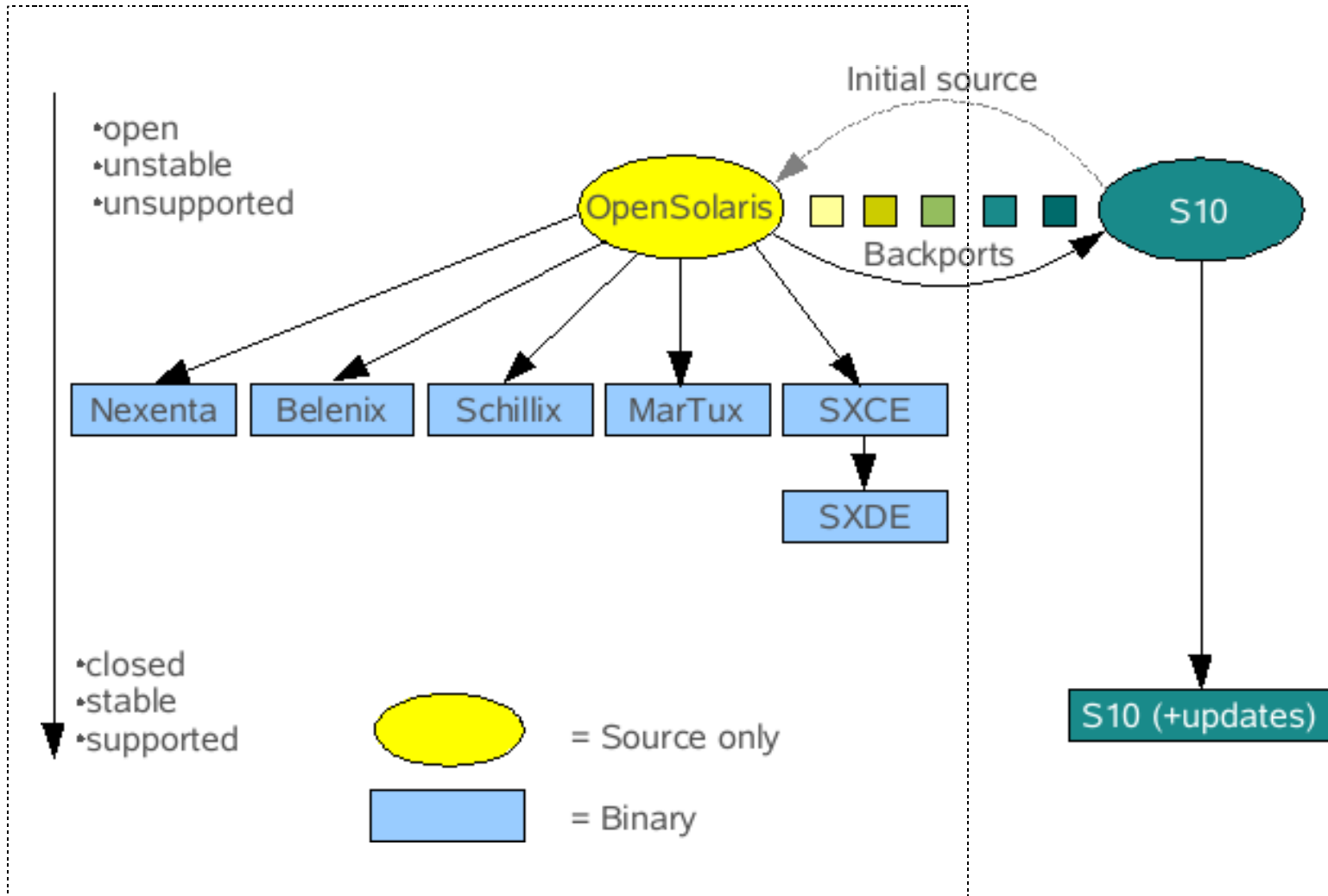
OpenSolaris Distribution: marTux

- First external SPARC distribution
- Key feature: first Xorg window server for OpenSolaris/SPARC

Distribution: Solaris

- Solaris is based on OpenSolaris, plus the other parts that have not been open sourced (yet).
- No full **major** Solaris release based **fully** on OpenSolaris available yet.
- The next Solaris release (currently codenamed “Nevada”) will be the first release based on OpenSolaris.

The OpenSolaris/Solaris Relationship



Building OpenSolaris

- ... or in this case: building ON (OS/Net)
- ON is what you need to get a basic, fully-functional system up and running
 - > Contains the kernel, system commands and libraries, and networking and driver support
- Other consolidations may have other build methods

Building ON: the basic steps

- Install Solaris Express Community Edition
- Get the source
- Get the tools
- Set up the build environment
- Start the build

Installing SXCE

- Currently, SXCE build 57 or later is needed
 - > Starter kit has Build 57
- To download SXCE, follow the download instructions (at the URL below)
- Then perform a normal Solaris install
- The rest of this presentation assumes you're up and running with SXCE

<http://opensolaris.org/os/downloads/on>

Source code management systems

- The opensolaris.org website supports two Source Code Management (SCM) solutions:
 - > Mercurial (hg) is the default. It was chosen for a distributed SCM (DSCM) solution and is recommended for projects/consolidations currently using a distributed development model and/or TeamWare.
 - > Subversion (SVN) is provided for exceptions. It was chosen for a centralized solution for individual projects and consolidations that have a requirement to use SVN or centralized source code management.

Getting the source

- Create a directory on a file system with at least 1G of disk space available
 - > For our example, use `/builds/onnv-gate`
- check out the ON source from the Mercurial repository at `hg.openolaris.org`

```
$cd /builds
$whence hg
/usr/bin/hg
$ hg clone -ssh "ssh -C" \
  ssh://anon@hg.openolaris.org/hg/onnv/onnv-gate
```

Get the tools

- Tools needed
 - > Sun Studio 11 (compiler)
 - > ON-specific build tools (SUNWonbld)
- Also needed: some binaries from parts of Solaris that have not (yet) been open-sourced (closed-bins tarball)
- Follow download instructions at
 - > <http://www.opensolaris.org/os/downloads/on/#tarfiles>
- Also available on the starter kit (all on the *<install>* disc)

Starting the build process

- Building is done using the `nightly(1)` and `bldev(1)` tools
- First step: extract the tools and set up their environment
- Then set up the basic variables for the build

Setting up the build environment

- `nightly(1)` and `bldenv(1)` use an environment file with settings
- Example `opensolaris` file in
`/builds/onnv-gate/usr/src/tools/env/opensolaris.sh`
- Copy this file to a convenient spot (in `/builds/onnv-gate`) and edit it to setup the environment

Build environment variables

- **\$GATE**: the name of the “gate”
 - > Just use the name of the top level directory for this one, `onnv-gate`
- **\$CODEMGR_WS**: the path to the workspace
 - > Use the full pathname to your build directory, `/builds/onnv-gate`
- **\$STAFFER**: the person doing the build, and who will get notification email
 - > Use your login name

Building the entire system

- In your work directory
(/builds/onnv-gate):
 - > \$ nohup nightly ./opensolaris.sh &
- This will compile all of the ON sources from scratch, cleaning out any old object files
- To follow the progress of the build:
 - > \$ tail -f /builds/onnv-gate/log/nightly.log
- To do an incremental build, use the `-i` option to the `nightly(1)` tool

Building part of the system

- First, do a complete build as described before
 - > Unless you're doing a kernel build only
- Use `bldeenv(1)` to set up the environment
- Go to the directory in the source tree where what you want to rebuild is located
- Use `dmake(1)`

Building part of the system, 2

- Rebuilding vi(1):

```
$ cd /builds/onnv-gate
$ bldenv ./opensolaris.sh
...
$ cd usr/src/cmd/vi
$ dmake all
```

- Rebuilding the kernel:

```
$ cd /builds/onnv-gate
$ bldenv ./opensolaris.sh
...
$ cd usr/src/uts
$ dmake all
```

Installing and testing

- Rebuilding ON as described does not create any bootable install media
 - > Change coming: LiveMedia project begun
 - > Packages are created and can be individually installed on a running system
- Upgrade through the usual method is not possible
- Various special upgrade forms allow installation and testing of the (possibly modified) system you just built

<http://opensolaris.org/os/project/livemedia/>

Installing and testing a whole build

- BFU (Blindingly Fast Upgrade, or Bonwick-Faulkner Upgrade)
 - > Upgrades ON components of a system—and only ON—from `cpio(1)` archives
 - > These archives are optionally created during a `nightly(1)` build
- **Caution!** BFU is aimed at developers only, and there are some pitfalls
 - > Conflicts, from modified configuration files, will be flagged and have to be automatically merged using a script called `acr`, or manually.
 - > Normal upgrades and BFU upgrades don't mix
 - > `acr` is **VERY VERY important!!!**

Installing and testing a kernel

- Kernels can be installed separately using Install (“Cap-Eye Install”)
- Install creates a tarfile with the needed kernel object files
- This tarfile can be extracted in the root directory, and will put the files in a separate directory; the original kernel is not overwritten
- The new kernel can be test booted using its separate pathname (passed in the multiboot line of GRUB)

Install pitfalls

- **Double caution!!**
- Must be careful to pass the right flags on the command line, it is easy to forget the right platform-dependent object files that the kernel needs
- Not all drivers are in ON, so some of them may have to be copied over by hand into the new kernel directory that Install creates.

Source access

- Download tar archives and Mercurial bundles at opensolaris.org
- Direct source code management access at opensolaris.org
 - > “Anonymous”, read-only access for everyone
 - > Commit access (after approval) for OpenSolaris contributors
- External source mirrors

<http://opensolaris.org/os/project/onnv>

<http://svn.genunix.org/>

Contributing

- Sign up at opensolaris.org!
- Join the discussions on one of the mailing lists
 - > General OpenSolaris lists, and community- or project-specific ones
 - > Check out [opensolaris-code](#) or [tools-discuss](#)
- Reporting/querying bugs
 - > Work in progress to improve access to the Sun internal database.

<http://opensolaris.org/os/discussions/>

<http://bugs.opensolaris.org/>

Contributing

- Code / bugfixes
 - > File a bug or “RFE” via `bugs.opensolaris.org`
 - > Start out with a “bite size” bug
 - > Request a sponsor (who will guide you through the integration processes)
 - > Develop and test your code
 - > Have it reviewed, and integrated
- Documentation, I18N/L10N, marketing, ...
- Join the community, share your ideas, help make OpenSolaris even better!

http://opensolaris.org/os/bug_reports/oss_bite_size/request-sponsor@opensolaris.org

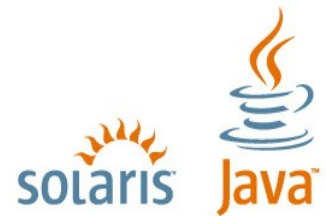


Building and Deploying **open**SOLARIS™

Wyllys Ingersoll

UNLOCK
OPPORTUNITY

What will you open?



SUN TECH DAYS 2006-2007
A Worldwide Developer Conference