

Release Engineering

The Integration Dock

```
/net/infra01/export/integrate_dock/oi/oi_$(BUILDID)
```

Consolidations delivered to the integration dock should have the following directory structures:

- onnv_\$(BUILDID)/\$(ARCH)/nightly-nd/repo.redist
- slim_source_\$(BUILDID)/\$(ARCH)/nightly-nd/repo.redist
- xnv_\$(BUILDID)/pkg_\$(ARCH)
- pkg_\$(BUILDID)/\$(ARCH)/repo
- vpanels_\$(BUILDID)/\$(ARCH)/repo

Other consolidations provide SVR4 packages.

```
[trisk@fastdev01]% cd /net/infra01/export/integrate_dock/oi/oi_148
[trisk@fastdev01]% ls
admin_151a_import      jds-oi_148          sic_team_134b_import
admin+sic_team_148    l10n_134b_import   slim_source_148
cacao_134b_import     l10n_151a_import   smcc_134b_import
cde_134b_import       man_134b_import    SunVTS_134b_import
cns_134b_import       mozilla-l10n-oi_148 ub_javavm_151a_import
dbtg_151a_import      nspg_134b_import   vpanels_148
devpro_134b_import    nvidia_151a_import xnv_148
gfx_151a_import       onnv_148           xvm_151a_import
hcts_151a_import      pkg_148
jdmk_134b_import      sfw_148
```



*_import refers to SVR4 packages reproduced from IPS repositories.

/net/infra01/export/integrate_dock/svr4_import/ips2svr4.pl handles shell metacharacters in filenames now.

solaris_re Consolidation

The solaris_re consolidation is prepared last, and consists of /etc/release and some licence files for the release. These are SVR4 packages, so we have to add them to the dock.

```
[trisk@fastdev01]% cd /net/infra01/export/releng/oi/oi_148
[trisk@fastdev01]% hg clone http://hg.openindiana.org/hg/openindiana
[trisk@fastdev01]% cd openindiana/solaris_re
[trisk@fastdev01]% RELEASE=148 ./build-packages.sh
[trisk@fastdev01]% cd /net/infra01/export/integrate_dock/oi/oi_148
[trisk@fastdev01]% ln -s ../../../../releng/oi/oi_148/openindiana/solaris_re .
```

"Wad of Stuff"

The WOS is created using symlinks back to each SVR4 consolidation directory.

- ../oi\$(ARCH_INITIAL)_wos_148/\$(ARCH)/* -> {sfw_148,admin+sic_team,...}/\$(ARCH)/*
- ../oi\$(ARCH_INITIAL)_wos_148/all/* -> {sfw_148,admin+sic_team,...}/all/*

```

[trisk@fastdev01]% mkdir -p /net/infra01/export/integrate_dock/oi/oix_wos_148/i386
[trisk@fastdev01]% cd /net/infra01/export/integrate_dock/oi/oix_wos_148/i386
[trisk@fastdev01]% ln -s ../../oi_148/sfw_148/i386/nightly-nd/* \
  ../../oi_148/jds-oi_148/PKGS/* \
  ../../oi_148/mozilla-110n-oi_148/PKGS/* \
  ../../oi_148/admin+sic_team_148/i386/* \
  ../../oi_148/*_134b_import/i386/* \
  ../../oi_148/*_151a_import/i386/* \
  ../../oi_148/solaris_re/packages/i386/* \
  .
[trisk@fastdev01]% mkdir -p /net/infra01/export/integrate_dock/oi/oix_wos_148/all
[trisk@fastdev01]% cd /net/infra01/export/integrate_dock/oi/oix_wos_148/all
[trisk@fastdev01]% ln -s ../../oi_148/*_134b_import/all/* \
  ../../oi_148/*_151a_import/all/* \
  ../../oi_148/solaris_re/packages/all/* \
  .

```

pkg-gate for Release Engineering

We will need a partial build of the appropriate version of pkg(5) to create our release.

```

[trisk@fastdev01]% cd /net/infra01/export/releeng/oi/oi_148
[trisk@fastdev01]% hg clone -U http://hg.openindiana.org/pkg-gate
[trisk@fastdev01]% cd pkg-gate
[trisk@fastdev01]% hg clone http://hg.openindiana.org/mq_pkg-gate .hg/patches
[trisk@fastdev01]% hg up -r in148
[trisk@fastdev01]% hg qpush oi-releeng-distro-import
[trisk@fastdev01]% cd src
[trisk@fastdev01]% PATH=/opt/sunstudio12.1/bin:$PATH make install SUBDIRS= || \
  true # ignore CherryPy failure

```

Creating a Repository

We need to create a destination repository for our packages. Its contents will be merged into the /dev repository later to accommodate upgrades. To create a repository in the current format:

```

[trisk@fastdev01]% pkgrepo create file:///net/infra01/export/releeng/oi/oi_148/repo
[trisk@fastdev01]% pkgrepo set -s file:///net/infra01/export/releeng/oi/oi_148/repo \
  publisher/prefix=openindiana.org
[trisk@fastdev01]% pkgrepo refresh -s file:///net/infra01/export/releeng/oi/oi_148/repo

```

For the older repository format, do this instead:

```

[trisk@fastdev01]% pkgsend -s file:///net/infra01/export/releeng/oi/oi_148/repo \
  create-repository --set-property publisher.prefix=openindiana.org
[trisk@fastdev01]% pkgsend -s file:///net/infra01/export/releeng/oi/oi_148/repo refresh-index

```

distro-import

This step imports all of our SVR4 packages into IPS, creates metapackages (incorporation and cluster packages) and ensures package consistency. You'll want a copy of Rich Lowe's [Distro Importer Notes](#) for this process. However, we are not using his streamlined `ossnap_cluster`, but building a full `r edist_cluster` instead.

```
[trisk@fastdev01]% cd util/distro-import
[trisk@fastdev01]% hg qpush -a
[trisk@fastdev01]% hg qnew oi-releng-src-util-distro-import-148
[trisk@fastdev01]% /usr/bin/make 148/redis_import > log 2>&1 & tail -f log
```

The actual process will take forever, and probably won't work the first time.

If for some reason you need to rebuild specific packages, use `JUST_THESE_PKGS`:

```
[trisk@fastdev01]% /usr/bin/make BUILDDID=148 JUST_THESE_PKGS="release/copyright
release/name slim_install babel_install server_install" \
redis_import > log2 2>&1 & tail -f log2
```



Using `EXTRA_OPTIONS=-d` helps in debugging `redis_import` errors such as:

```
Traceback (most recent call last):
  File "./importer.py", line 1895, in <module>
    main_func()
  File "./importer.py", line 1864, in main_func
    publish_pkg(_p, g_proto_area)
  File "./importer.py", line 575, in publish_pkg
    assert len(svr4_traversal_dict) == len(svr4_traversal_list)
AssertionError
*** Error code 1
make: Fatal error: Command failed for target `redis_import'
Current working directory /net/infra01/export/releng/oi/oi_148/pkg-gate/src/util/distro-import
*** Error code 1
make: Fatal error: Command failed for target `148/redis_import'
```

Caused by:

```
[trisk@fastdev01]% cat devpro_134b_import/i386/SUNWlibmsr/pkgmap | head -2
1 i copyright 9569 0 0
1 i copyright 11910 0 0
```

All of our SVR4 packages for CDE consolidation were regenerated from IPS packages. The IPS manifests assemble the IPS packages piecemeal from the contents of multiple SVR4 packages, a process which can't be inverted to reproduce the original SVR4 packages. To satisfy this process, we had to create a single SVR4 "mega-package" which has contents (`reloc` and `pkgmap`) of all of the CDE SVR4 packages merged.

set-publisher

This step imports packages from consolidations that produce their own IPS repositories into our new repository. The packages imported from IPS repositories may need their publisher names changed, and we may wish to perform other metadata transformations with `pkgmogrify`. To update the legacy package vendor strings ([issue 192](#)) we will use a modified `set-publisher.transforms`. This time we're only going forward one build so we can use the default `publish_setpublisher` target which doesn't add past metadata. `-b` may have to be removed for future builds.

```
[trisk@fastdev01]% cd ../publish
[trisk@fastdev01]% /usr/bin/make BUILDDID=148 publish_setpublisher
```



To manually merge the packages created by `set-publisher` into a repo, use a script like the following. Hopefully, you'll never need to do this.

```
#!/bin/ksh -p
PROTO=../../../../proto/root_i386
PYTHONPATH=$PROTO/usr/lib/python2.6/vendor-packages
PKGPATH=$PROTO/usr/bin
recv_dir=../../../../proto/set-publisher
publish_repo=file:///export/releng/oi/oi_148/pkg-gate/src/util/publish/repo
PYTHONPATH=$PYTHONPATH PATH=$PKGPATH:$PATH
export PYTHONPATH
export PATH
for pkg in $(echo $recv_dir/*/*); do
    pkgsend -s $publish_repo publish --fmri-in-manifest \
        --no-catalog --no-index \
        -d $pkg $pkg/manifest
done
$PROTO/usr/lib/pkg.depotd \
    -d /export/releng/oi/oi_148/pkg-gate/src/util/publish/repo \
    --add-content --exit-ready
```

Publishing to /dev

Finally, we will transfer contents of the new repository into a clone of our existing /dev repository, and rebuild the content index.

```
[trisk@fastdev01]% ssh infra01
[trisk@infra01]% pfexec zfs clone \
    data/zones/pkg01.uk.openindiana.org/ROOT/export/pkg/dev@2010-09-27_2239 \
    data/export/releng/pkg/dev
[trisk@infra01]% pfexec zfs snapshot data/export/releng/pkg/dev@oi_147
[trisk@infra01]% pfexec chown -R 510:510 /export/releng/pkg/dev
[trisk@infra01]% exit
```

If both repositories are the same version, simply `rsync` the files over.

```
[trisk@fastdev01]% rsync -avP /net/infra01/export/releng/oi/oi_148/repo/ \
    /net/infra01/export/releng/pkg/dev/
[trisk@fastdev01]% /usr/lib/pkg.depotd -d /net/infra01/export/releng/pkg/dev \
    --add-content --exit-ready
```

Alternative for the last two commands. May be slower, but necessary if the repository versions differ.

```
[trisk@fastdev01]% /usr/lib/pkg.depotd \
    -d /net/infra01/export/releng/oi/oi_148/repo --add-content --exit-ready
[trisk@fastdev01]% pkgrecv -s file:///net/infra01/export/releng/oi/oi_148/repo \
    -d file:///net/infra01/export/releng/pkg/dev '*'
```