

Solarpack developer's guide

Author: Marc Herbert

Revision: 1.7

Contents

- 1 [Introduction : patches over patches](#)
- 2 [Understanding solarpack's organization](#)
- 3 [Solarpatches](#)
- 4 [The solarpack developer's point of view](#)

1 Introduction : patches over patches

The solarpack CVS repository is designed to port the NetBSD `pkgsrc` software distribution to Solaris. This document details the technical choices made to make solarpack development hopefully the easiest.

A particular and tricky problem it has to solve, is to manage sometimes patches "over patches", since the NetBSD distribution is itself composed of upstream sources and patches.

Ideally, most solarpack code should go up this chain and the solarpack layer become as thin as possible.

2 Understanding solarpack's organization

We document here the modifications made to the NetBSD distribution in order to automate the downloading and compiling of the solarpack source distribution. Solarpack contributors, who also want to *submit* code or patches to the solarpack tree, will find even more details in the next, "developer's" section.

Here is a summary of what contains the solarpack CVS repository:

- 1) a snapshot of the NetBSD `pkgsrc` source tree, hopefully "stable", and increased with...
- 2) ...numerous modifications of its build toolchain (Makefiles, scripts, etc.)
- 3) for some packages, modifications of the source code (`pkgsrc/foo/bar/solarpatches/solarpatch-*` files)

3 Solarpatches

Sometimes not only the build system (Makefiles, scripts,...) of NetBSD has to be modified here and there to make it work on Solaris, but we also need to change the original sources.

The trick is to insert our patches inside the patching step of the standard fetch-patch-compile-install NetBSD procedure (See `Packages.txt` [1] section 7.2, "Main targets")

This is achieved by a slight modification of the big, main `pkgsrc` Makefile (located at `pkgsrc/mk/bsd.pkg.mk`). The "patching" code of this Makefile (search "patchfile", lowercase, in `bsd.pkg.mk`) browses different locations searching for patches to apply. We inserted our own directory `foo/solarpatches/solarpatch-*` in this list, near the end of the list: after the patches from NetBSD located in `${PATCHDIR}`, but before the final `_${LOCALPATCHFILES}` (typically from `/usr/local`).

4 The solarpack developer's point of view

This section gives enough details to help you contribute to solarpack.

First read the previous section.

If you want to submit some modification to the NetBSD build toolchain (Makefiles, scripts, etc.), just proceed as for any other open-source project: send a patch and forget the rest. But if you want to modify original software sources, things are a little more complex, please read on.

The fact of maintaining patches over patches is both a drawback because of the increased complexity; but also an advantage since we can reuse some of the tools that NetBSD uses to maintain their patches. These `pkgtools` are:

`pkgvi`: is a simple script that launches your `$EDITOR` and makes a copy of the original `foo` file to `foo.orig` on exit (if modified). When `.orig` does already exist, it just edits the file.

`pkgdiff`: is an ergonomic diff with options tuned to produce a NetBSD patch. It takes `foo.orig` as the default reference.

`mkpatches`: recursively calls `pkgdiff`, and produces a set of `patches-XX` into a user-chosen directory (default: `.newpatches`)

`patchdiff`: compares two sets of `patches-XX`

Note: never submit differences between patches (i.e., patch on patches, like `patchdiff` produces) since most people find them unreadable. Always submit whole new versions of patches, and let maintainers diff patches whenever they want to.

Now please read the recommended way of contributing to NetBSD's `pkgsrc`, using the above tools: <<ftp://ftp.netbsd.org/pub/NetBSD/packages/pkgsrc/Packages.txt>> (section 9 : Debugging)

Congratulations for all this painful reading. Here is now the good news: since solarpack provides modified `pkgvi/pkgdiff/...` tools, the procedure to produce solarpatches is precisely the one you just read.

Below is a diagram of the successive patching steps invoked by the "bmake patch" build process. All steps are optional, and so all "orig" files may NOT be there:

[1] Available in your local `/usr/solarpkgsrc` checkout, or from <ftp://ftp.netbsd.org/pub/NetBSD/packages/pkgsrc/Packages.txt>

```

ftp://ftp.sourcesite.org/bar/
foo.c

      |
patch <      |
ftp.altsource.org |----> foo.c.orig_dist
      |
      v

foo.c

      |
patch <      |
ftp.netbsd.org  |----> foo.c.orig
      |
      v

foo.c

      |
patch <      |
solarpack.sf.net |----> foo.c.orig_solar
      |
      v

foo.c

```

The first two patching steps come from NetBSD, and the third one has been added for solarpack. NetBSD's versions of the pkgtools produce/take as reference "foo.c.orig". Solarpack's modified pkgtools use instead "foo.c.orig_solar". Corollary : you can NOT easily produce NetBSD patches when working with solarpack's CVS.