

OSELAS.Support
OSELAS.Training
OSELAS.Development
OSELAS.Services

Application Note

Installing PTXdist-2012.12.0

PTXdist-2012.12.0 comes in an archive. This application note advises users what exactly needs to be done to get a working environment.



Pengutronix e. K.
Peiner Straße 6-8
31137 Hildesheim

+49 (0)51 21 / 20 69 17 - 0 (Fon)
+49 (0)51 21 / 20 69 17 - 55 55 (Fax)

info@pengutronix.de

PTXdist Installation

PTXdist is shipped in an archive. This application note provides information about which archive is required to be downloaded and installed to get a working environment to build filesystems for their target systems.

Main Parts of PTXdist

The most important software component which is necessary to build an OSELAS.BSP() board support package is the `ptxdist` tool. So before starting any work we'll have to install PTXdist on the development host.

PTXdist consists of the following parts:

The `ptxdist` Program: `ptxdist` is installed on the development host during the installation process. `ptxdist` is called to trigger any action, like building a software packet, cleaning up the tree etc. Usually the `ptxdist` program is used in a *workspace* directory, which contains all project relevant files.

A Configuration System: The config system is used to customize a *configuration*, which contains information about which packages have to be built and which options are selected.

Patches: Due to the fact that some upstream packages are not bug free – especially with regard to cross compilation – it is often necessary to patch the original software. PTXdist contains a mechanism to automatically apply patches to packages. The patches are bundled into a separate archive. Nevertheless, they are necessary to build a working system.

Package Descriptions: For each software component there is a “recipe” file, specifying which actions have to be done to prepare and compile the software. Additionally, packages contain their configuration snippet for the config system.

Toolchains: PTXdist does not come with a pre-built binary toolchain. Nevertheless, PTXdist itself is able to build toolchains, which are provided by the OSELAS.Toolchain() project. More in-deep information about the OSELAS.Toolchain() project can be found here: http://www.pengutronix.de/oselas/toolchain/index_en.html

Building a toolchain is not part of this application note, refer for application note “Building Toolchains” instead.

Board Support Package This is an optional component, mostly shipped aside with a piece of hardware. There are various BSP available, some are generic, some are intended for a specific hardware.

Download Required Archives

In order to follow this application note, some software archives are needed. There are several possibilities how to get these: either as part of an evaluation board package, or by download from a world wide web site.

The central place for OSELAS related documentation is <http://www.oselas.com>. This website provides all required packages and documentation (at least for software components which are available to the public).

To follow the steps below, the following archive has to be available on the development host:

- `ptxdist-2012.12.0.tar.bz2`

Extracting the Sources



Do the following steps at best in your own home directory (\$HOME). You need root permissions only in the `make install` step, and **nowhere** else.

To install PTXdist, the archive Pengutronix provides has to be extracted:

ptxdist-2012.12.0.tar.bz2 The PTXdist software itself

The PTXdist archive has to be extracted into some temporary directory in order to be built before the installation, for example the `local/` directory in the user's home. If this directory does not exist, we have to create it and change into it:

```
$ cd
$ mkdir local
$ cd local
```

Next step is to extract the archive:

```
$ tar -xjf ptxdist-2012.12.0.tar.bz2
```

If everything goes well, we now have a `PTXdist-2012.12.0` directory, so we can change into it:

```
$ cd ptxdist-2012.12.0
$ ls -lF
total 521
-rw-r--r--  1 jb user  18446 Dec 12 12:22 COPYING
-rw-r--r--  1 jb user   4048 Dec 12 12:22 CREDITS
-rw-r--r--  1 jb user 115540 Dec 12 12:22 ChangeLog
-rw-r--r--  1 jb user    57 Dec 12 12:22 INSTALL
-rw-r--r--  1 jb user   3868 Dec 12 12:22 Makefile.in
-rw-r--r--  1 jb user   4268 Dec 12 12:22 README
-rw-r--r--  1 jb user  63516 Dec 12 12:22 TODO
-rwxr-xr-x  1 jb user    28 Dec 12 12:22 autogen.sh*
drwxr-xr-x  2 jb user    72 Dec 12 12:22 bin/
drwxr-xr-x 11 jb user   320 Dec 12 12:22 config/
-rwxr-xr-x  1 jb user 224066 Dec 12 12:27 configure*
-rw-r--r--  1 jb user  12175 Dec 12 12:22 configure.ac
drwxr-xr-x 10 jb user   248 Dec 12 12:22 generic/
drwxr-xr-x 232 jb user  7832 Dec 12 12:22 patches/
drwxr-xr-x  2 jb user  1656 Dec 12 12:22 platforms/
drwxr-xr-x  4 jb user   112 Dec 12 12:22 plugins/
lrwxrwxrwx  1 jb user    7 Dec 16 11:31 projectroot -> generic/
drwxr-xr-x  6 jb user  56520 Dec 12 12:22 rules/
drwxr-xr-x  9 jb user   936 Dec 12 12:22 scripts/
drwxr-xr-x  2 jb user   512 Dec 12 12:22 tests/
```

Prerequisites

Before PTXdist can be installed it has to be checked if all necessary programs are installed on the development host. The configure script will stop if it discovers that something is missing.

The PTXdist installation is based on GNU autotools, so the first thing to be done now is to configure the packet:

```
$ ./configure
```

This will check your system for required components PTXdist relies on. If all required components are found the output ends with:

```
[...]
checking whether python development files are present... yes
checking for patch... /usr/bin/patch
checking whether /usr/bin/patch will work... yes

configure: creating ./config.status
config.status: creating Makefile

ptxdist version 2012.12.0 configured.
Using '/usr/local' for installation prefix.

Report bugs to ptxdist@pengutronix.de
```

Without further arguments PTXdist is configured to be installed into `/usr/local`, which is the standard location for user installed programs. To change the installation path to anything non-standard, we use the `--prefix` argument to the `configure` script. The `--help` option offers more information about what else can be changed for the installation process.

The installation paths are configured in a way that several PTXdist versions can be installed in parallel. So if an old version of PTXdist is already installed there is no need to remove it.

One of the most important tasks for the `configure` script is to find out if all the programs PTXdist depends on are already present on the development host. The script will stop with an error message in case something is missing. If this happens, the missing tools have to be installed from the distribution before re-running the `configure` script.

When the `configure` script is finished successfully, we can now run

```
$ make
```

All program parts are being compiled, and if there are no errors we can now install PTXdist into its final location. In order to write to `/usr/local`, this step has to be performed as user `root`:

```
$ sudo make install
[enter password]
[...]
```

If we don't have root access to the machine it is also possible to install PTXdist into some other directory with the `--prefix` option. We need to take care that the `bin/` directory below the new installation dir is added to our `$PATH` environment variable (for example by exporting it in `~/.bashrc`).

The installation is now done, so the temporary folder may now be removed:

```
$ cd ../../
$ rm -fr local
```

Configuring PTXdist

When using PTXdist for the first time, some setup properties have to be configured. Two settings are the most important ones: Where to store the source archives and if a proxy must be used to gain access to the world wide web.

Run PTXdist's setup:

```
$ ptxdist setup
```

Due to PTXdist is working with sources only, it needs various source archives from the world wide web. If these archives are not present on our host, PTXdist starts the `wget` command to download them on demand.

Proxy Setup

To do so, an internet access is required. If this access is managed by a proxy `wget` command must be advised to use it. PTXdist can be configured to advice the `wget` command automatically: Navigate to entry *Proxies* and enter the required addresses and ports to access the proxy in the form:

`<protocol>://<address>:<port>`

Source Archive Location

Whenever PTXdist downloads source archives it stores these archives in a project local manner. This is the default behaviour. If we are working with more than one PTXdist based project, every project would download its own required archives in this case. To share all source archives between all projects, PTXdist can be configured to share only one archive directory for all projects it handles: Navigate to menu entry *Source Directory* and enter the path to the directory where PTXdist should store archives to share between its projects.

Generic Project Location

If we already installed the generic projects we should also configure PTXdist to know this location. If we already did so, we can use the command `ptxdist projects` to get a list of available projects and `ptxdist clone` to get a local working copy of a shared generic project.

Navigate to menu entry *Project Searchpath* and enter the path to projects that can be used in such a way. Here we can configure more than one path, each part can be delimited by a colon. For example for PTXdist's generic projects and our own previous projects like this:

```
/usr/local/lib/ptxdist-2012.12.0/projects:/office/my_projects/ptxdist
```

Leave the menu and store the configuration. PTXdist is now ready for use.



If there is no toolchain available yet, the next step is to build one at least for the desired target architecture. Refer to the application note “Building Toolchains” for further details.

Additional Questions?

Below is a list of locations where you can get help in case of trouble. For questions how to do something special with PTXdist or general questions about Linux in the embedded world, try these.

Mailing Lists

About PTXdist in Particular

This is an English language public mailing list for questions about PTXdist. See

http://www.pengutronix.de/maillinglists/index_en.html

on how to subscribe to this list. If you want to search through the mailing list archive, visit

<http://www.mail-archive.com/>

and search for the list *ptxdist*. Please note again that this mailing list is just related to the PTXdist as a software. For questions regarding your specific BSP, see the following items.

About Embedded Linux in General

This is a German language public mailing list for general questions about Linux in embedded environments. See

http://www.pengutronix.de/maillinglists/index_de.html

on how to subscribe to this list. Note: You can also send mails in English.

About Working on the Linux Kernel

The book *Linux Kernel in a Nutshell* from Greg Kroah-Hartman. Its online version can be read here:

<http://www.kroah.com/lkn/>

Chat/IRC

About PTXdist in particular

irc.freenode.net:6667

Create a connection to the **irc.freenode.net:6667** server and enter the chatroom **#ptxdist**. This is an English room to answer questions about PTXdist. Best time to meet somebody there is at European daytime.

Commercial Support

You can order immediate support through customer specific mailing lists, by telephone or also on site. Ask our sales representative for a price quotation for your special requirements.

Contact us at:

**Pengutronix
Peiner Str. 6-8
31137 Hildesheim
Germany
Phone: +49 - 51 21 / 20 69 17 - 0
Fax: +49 - 51 21 / 20 69 17 - 55 55**

or by electronic mail:

sales@pengutronix.de

**If you want to contribute to this document
send your suggestions and texts under the
Creative Commons License Attribution 2.0
to jbe@pengutronix.de**

This is a Pengutronix Application Note

**Copyright Pengutronix e.K.
All rights reserved.**

**Pengutronix e.K.
Peiner Str. 6-8
31137 Hildesheim
Germany**

**Phone: +49 - 51 21 / 20 69 17 - 0
Fax: +49 - 51 21 / 20 69 17 - 55 55**

