



# Ebuild と オーバーレイ

このプレゼンテーションは私の意見(いけん)だけ.

## オーバーレイはなんですか

- overlayは追加のportage repositoryです。
- ここにあなたのebuildを追加ができます。

## ローカルオーバーレイの作り方

**make.conf**に設定します。

```
# PORTDIR_OVERLAY="/usr/local/portage/"
```

このディレクトリに**Portage**が追加の**package**を探しています。

ディレクトリをつくれます。

```
# mkdir -p /usr/local/portage/app-misc/hello-world
# cd $_
$_ recalls last argument
```

**ebuild**のテンプレを作ります。

```
# cp /usr/portage/header.txt ./hello-world-1.0.ebuild
```

これだけ動かない。

```
# echo 'SLOT="0"' >> ./hello-world-1.0.ebuild
```

## Ebuildの何ですか

Ebuildは何ですか

- Ebuildはテキストファイルです。
- Portageで使われるパッケージ管理用のBashスクリプトです。

## ebuildのインストール方

ebuildのインストール方。

```
# ebuild hello-world-1.0.ebuild manifest clean merge
```

ebuildを作った。

## Variableを追加

/usr/portage/skel.ebuild ドキュメント

これを読んで方がいいです。

## 大切なコマンド

man 5 ebuild

repoman manifest && repoman full

enalyze analyze -v USE

layman -S emerge --regen /etc/eixrc OVERLAY\_CACHE\_METHOD="assign"

<http://devmanual.gentoo.org/>

emerge --moo



## EAPI

PMS portage manager specificationはebuildの標準化（ひょうじゅんか）です。EAPIの番号はどんなPMSのバージョンを使います。

おすすめEAPIはEAPI5です

EAPIの情報はここ: <http://devmanual.gentoo.org/ebuild-writing/eapi/>

ebuildにEAPI="5"を追加します。このVariableは一?上です。

EAPI=5 も同じ

## DESCRIPTION

**DESCRIPTION**は**Package**の概要（がいよう）

```
DESCRIPTION="A simple ebuild learning example."
```

## homepageを追加

homepageは何のページにこのpackageを見つけた

```
HOMEPAGE="https://github.com/aliceinwire/Gentoo-landslide/tree/master/ebuild-overlay"
```

## SRC\_URIを追加

SRC\_URIはどこでこのpackageダウンロードをしていますか。

```
SRC_URI="https://github.com/aliceinwire/Gentoo-landslide/tree/master/ebuild-overlay/foo.tar.gz"
```

## LICENSEを追加

LICENSEはGPLやMIT等のソフトウェアライセンスです。

```
LICENSE="MIT"
```

## KEYWORDSを追加

KEYWORDSはどこでこのscriptを動きますか。

shell scriptだからどこでも動きます。

全部のarchを追加します。

```
KEYWORDS="~alpha ~amd64 ~arm ~hppa ~ia64 ~ppc ~ppc64 ~s390 ~sh ~sparc ~x86"
```

maskのPackageは -と追加しない。

untestedは ~

Stableは追加けど前の文字がない。

## EBUILDの最後

```
# Copyright 1999-2013 Gentoo Foundation
# Distributed under the terms of the GNU General Public License v2
# $Header: $

EAPI="5"

SLOT="0"

DESCRIPTION="A simple ebuild learning example."
HOMEPAGE="https://github.com/aliceinwire/Gentoo-landslide/tree/master/ebuild-overlay"
SRC_URI="https://github.com/aliceinwire/Gentoo-landslide/tree/master/ebuild-overlay/foo.tar.gz"

LICENSE="MIT"
KEYWORDS="~alpha ~amd64 ~arm ~hppa ~ia64 ~ppc ~ppc64 ~s390 ~sh ~sparc ~x86"
```

ビルドをします。

```
ebuild hello-world-1.0.ebuild manifest clean merge
```



でもなにもインストールをした。

インストールの**phase**

<http://devmanual.gentoo.org/ebuild-writing/functions/>

インストールの**function**

<http://devmanual.gentoo.org/function-reference/install-functions>

インストールはライブファイル**system**の中でなにもインストールをします。だから**ebuild**の中で **mv cp rm** のコマンドを普通使わない。

**Gentoo**インストール**function**のコマンドだけ使いますと **\${D}** (これは目的のディレクトリです。)

## src\_installとdobin

```
src_install() {  
    dobin hello-world  
}
```

dobinはhello-worldのscriptをビルドのディレクトリにコピーをしてexecutableのpermissionを設定して 後でPortageがこのファイルをチェックしてとライブのファイルsystemにコピーをします。

も一回びるどをします。

```
ebuild hello-world-1.0.ebuild manifest clean merge
```

```
>>> /usr/bin/hello-world
```

を見えます。

## オーバーレイの作り方

オーバーレイのソフトをインストールします。

```
# emerge layman
```

バージョン管理システムのためUSE FLAGを選びます。

**make.conf** ファイルの中で**layman**ディレクトリを追加します。

```
# echo "source /var/lib/layman/make.conf" >> /etc/portage/make.conf
```

**Layman**の設定をします。

```
# vim /etc/layman/layman.cfg  
overlays : file:///var/lib/layman/my-list.xml
```

追加をします。

## my-list.xml

```
<?xml version="1.0" ?>
<repositories version="1.0">
  <repo priority="50" quality="experimental" status="unofficial">
    <name>aliceinwire</name>
    <description>Custom stuff for Gentoo from aliceinwire.</description>
    <homepage>http://github.com/aliceinwire/</homepage>
    <owner>
      <email>alice.ferrazzi@gmail.com</email>
    </owner>
    <source type="git">git@github.com:aliceinwire/overlay.git</source>
  </repo>
</repositories>
```

## Proxy-maintainer

[https://wiki.gentoo.org/wiki/Project:Proxy\\_Maintainers](https://wiki.gentoo.org/wiki/Project:Proxy_Maintainers)

