Gentoo: FeM-Overlay

Das FeM-Overlay ist von Bitbucket in das FeM-GitLab umgezogen. Die neue Upstream-URL lautet:

https://gitlab.fem-net.de/gentoo/fem-overlay.git

Bitte passt eure `/etc/portage/repos.conf` an und erstellt neue Commits ab sofort immer im GitLab. Der Bitbucket-Mirror wird ca. Dezember 2021 abgeschaltet.

Das FeM-Overlay ist ein Portage Overlay für die Gentoo Paketverwaltung. Hier werden Ebuilds und Patches für Software gepflegt, die auf FeM Servern Einsatz findet, so aber nicht im offiziellen Portage-Tree enthalten ist.

Hinweis: Manchmal sind Pakete nur temporär im FeM-Overlay bis die Erweiterungen / Patches in den offiziellen Portage-Tree eingepflegt wurden.

Kontaktadresse: fem-overlay@technik.fem-net.de

Nutzung

- URL des Overlays (git): https://gitlab.fem-net.de/gentoo/fem-overlay.git
  - Upstream-Repository. Alle Commits werden hier in den master gemerged.
- URL des Overlays (git, alte URL): https://bitbucket.fem.tu-ilmenau.de/scm/gentoo/fem-overlay.git
  - Hinweis: das Bitbucket wird per Push-Mirror aus dem GitLab gesynct. Bitte im GitLab committen.
- URL des Overlays (svn): http://subversion.fem.tu-ilmenau.de/repository/fem-overlay/trunk/
  - Hinweis: das Subversion wird minütlich aus dem Git-Master gesynct und ist nicht mehr schreibbar. Bitte im GitLab-Repo commiten.

Ohne Layman über repos.conf

Diese Variante bietet sich an, wenn man selber am Repository Änderungen vornehmen möchte.

```
[fem-overlay]
location = /var/db/repos/fem-overlay
sync-type = git
sync-uri = https://gitlab.fem-net.de/gentoo/fem-overlay.git
auto-sync = yes
```

Mit eselect-repository

Diese Variante verwendet in der Standardeinstellung ein Clone vom Repository mit zusätzlich generierten Metadaten, was emerge beschleunigt. Empfiehlt sich für Systeme, die das Overlay nur nutzen, aber lokal selbst keine Änderungen hinzufügen.

```
eselect repository enable fem-overlay
```

Mit Layman

Das Overlay ist in der offiziellen Liste enthalten.

Hinzufügen des Overlays

```
layman -a fem-overlay
```

layman wieder loswerden

```
# layman Referenzen entfernen
rm /etc/portage/repos.conf/layman.conf
sed -i -e '/source \/var\/lib\/layman\/make.conf/d' /etc/portage/make.conf
sed -i -e '/\*/d' /etc/eix-sync.conf

emerge -C layman

rm -rf /var/lib/layman

eix-sync # oder emerge --sync
```

Ebuilds einreichen

Für einen schreibenden Zugriff benötigt man einen Login im GitLab. Die ist üblicherweise der FeM LDAP-Zugang. Neue ebuilds können per Fork und anschließendem Merge Request eingericht werden. Für zusätzliche Berechtigunge (MRs selbst mergen, Issues bearbeiten, etc.) kann man sich an fem-overlay@technik.fem-net.de wenden.

Regeln für erfolgreiche Ebuilds im FeM-Overlay

- Verwende bei neuen Ebuilds die aktuellste EAPI. (EAPI Cheat-Sheet)
  - Einzelne eclasses unterstützen nicht die neueste EAPI. In diesem Fall kann die nächstältere benutzt werden. EAPIs, die im Overlay als veraltet markiert sind, *dürfen nicht* in neuen ebuilds benutzt werden.
- Im Files-Ordner sollen keine Archive, sondern nur Skripte und Patches liegen.

- Alle Ebuilds besitzen einen validen Header

Header

```
# Copyright 1999-2XXX Gentoo Authors
# Distributed under the terms of the GNU General Public License v2
```

- Variablen sollten in der richtigen Standard-Reihenfolge vorliegen

Standard-Reihenfolge

```
EAPI

inherit

MY_P
DESCRIPTION
HOMEPAGE
SRC_URI

LICENSE
SLOT
KEYWORDS
IUSE

DEPEND
RDEPEND
BDEPEND

S
RESTRICT

DOCS
```

- Variablen, die evtl. Leerzeichen enthalten könnten müssen bei Verwendung mit Kommandos gequotet werden
  - standardmäßig: `${S}, ${WORKDIR}, ${FILESDIR}, ${DISTDIR}, ${ROOT}, ${D}`

Beispiel

```
# Quoting nicht nötig, da Ergebnis eine Variable ist
S=${WORKDIR}/${PN}

# Quoting nötig, da der Pfad u.U. Leerzeichen enthält
```

```
cd "${S}" || die
```

- möglichst eine metadata.xml anlegen mit Informationen, wer das Paket im Overlay betreut und welche Useflags verwendet werden. Die E-Mail-Adressen der Betreuer werden in der CI-Konfiguration der fem-overlay-ci-tools hinterlegt, um eine automatische Zuweisung von Issues aufgrund der betroffenen Pakete zu ermöglichen.

Beispiel für metadata.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE pkgmetadata SYSTEM "http://www.gentoo.org/dtd/metadata.dtd">
<pkgmetadata>
        <maintainer type="person">
                <email>mamu@fem.tu-ilmenau.de</email>
                <name>Max Mustermann</name>
        </maintainer>
        <use>
                <flag name="ftps">Support for backing up on FTPS</flag>
        </use>
</pkgmetadata>
```

- Manifest erstellen mit pkgdev oder RepoMan:
  - `pkgdev manifest`
  - `repoman manifest`
- RepoMan zur Prüfung des Ebuilds verwenden
  - `repoman full -dx` (Prüft alle Ebuilds im aktuellen Verzeichnis)
- pkgdev oder RepoMan zum commiten benutzen
  - `repoman commit`
  - `pkgdev commit`
- pkgcheck zur Prüfung des Ebuilds verwenden
  - `pkgcheck scan` (prüft Ebuilds im aktuellen Verzeichnis auf gängige Konventionen und Fehler)

(Ideen aus den Coding-Standards des Sunrise-Overlays.)

Das Overlay wird bei Änderungen u.A. mittels Repoman und pkgcheck per CI-Pipeline geprüft. Zudem werden ebuilds per Tinderbox gebaut und Fehler gemeldet. **Sämtliche Pakete müssen erfolgreich die CI-Pipeline durchlaufen, um akzeptiert zu werden.**

Historie

- ca. 2008 - Das Overlay wurde angelegt
- 2017-11-23 - Migration auf Git
- 2018-03-23 - Aufnahme in overlays.gentoo.org
- 2021-07-29 - Erste Arbeiten an der Umstellung auf den neuen Upstream https://gitlab.fem-net.de/gentoo/fem-overlay
- 2021-08-02 - Änderung des Upstream-Repos für `api.gentoo.org` auf GitHub
- 2021-08-13 - Das Overlay benutzt nun das GitLab als Upstream-Repository

From:
https://wiki.fem.tu-ilmenau.de/ - **FeM-Wiki**

Permanent link:
https://wiki.fem.tu-ilmenau.de/public/technik/gentoo-overlay?rev=1670695030

Last update: **2022/12/10 18:57**