

Rapport Conception OS - Gentoo Installer

Introduction

Gentoo est une distribution Linux dont le gestionnaire de paquet (portage) installe les paquets à partir de leur code source. Gentoo est donc ce que l'on appelle une *source based distribution*, ce qui la rend plus configurable que n'importe quelle autre distribution basée sur des paquets binaires puisqu'il est possible de modifier les fonctionnalités des paquets avant leur compilation, par le biais d'un système de *USE-flags*

Grâce à cette flexibilité de configuration, Gentoo est un système de paquets très flexible et très puissant, mais aussi complexe à configurer. De ce fait, l'installation du système doit se faire manuellement en ligne de commande puisqu'aucun installateur n'est fourni. Il est possible d'installer ce qu'on appelle des "stage4" qui sont des archives contenant un système complètement fonctionnel, mais ceux-ci sont donc préconfigurés et non adaptés à une configuration plus spécifique.

C'est dans cette optique que nous avons décidé de créer un générateur de scripts d'installation, afin de pouvoir facilement décrire le système dans l'état final que l'on désire, et laisser le générateur créer le script avec les commandes dans le bon ordre afin d'installer le système à notre place. De cette façon, il est facile de réinstaller le système plusieurs fois et de façon répétable sans devoir être derrière son écran. Il suffit de télécharger les scripts d'installation, lancer le principal, et revenir dans une heure ou deux une fois que le système a été compilé et installé.

Structure du projet

Le projet est composé de deux parties, une partie qui se charge de générer un script d'installation utilisant des fichiers de configuration donnés ou une configuration par défaut, ainsi qu'une interface graphique permettant de générer ces différents fichiers de configuration pour les personnes n'ayant pas la volonté de les créer par eux-mêmes.

Configuration

Le générateur de script d'installation (qui sera décrit par la suite) a besoin de deux fichiers de configuration :

- `installer.conf` : Configuration globale du système à installer.
- `disks.conf` : Configuration des partitions et des disques (support pour un seul disque pour le moment).

Pour générer ces fichiers de configuration, un utilisateur possède deux possibilités :

- il est à l'aise avec la recherche et la création de ces deux fichiers, pour pouvoir configurer au mieux son OS. L'utilisateur peut alors créer et fournir lui-même ces fichiers de configuration grâce à la documentation Gentoo.
- il peut directement utiliser l'installateur que nous fournissons pour choisir la configuration qui lui convient le mieux, et ainsi générer automatiquement ces fichiers de configuration.

Configuration directement via les fichiers .conf

Configurer l'installateur directement avec les fichiers de configuration permet de ne spécifier que les options désirées, en effet en cas d'absence d'un paramètre l'installateur va automatiquement choisir des valeurs par défaut qui fonctionnent pour la plupart des systèmes. Bien que les fichiers de configuration doivent être présents, il est possible de fournir un fichier `installer.conf` vide qui est considéré comme valide, l'installateur prendra les valeurs par défaut (voir plus bas). Cela n'est pas possible avec les fichiers `disks.conf` car par défaut le système n'aura aucune partition. Voir plus bas comment configurer les partitions.

Valeurs configurables dans `installer.conf`

Pas encore toutes les valeurs présentes dans le fichier de configuration d'exemple ne sont utilisées. Les valeurs configurables, par section, sont:

[gentoo]

- `arch` : architecture du système (p.ex. `amd64` , `i386` , ...)
 - par défaut : `amd64`
- `initssystem` : (partiellement implémenté) système d'amorçage (`openrc` ou `systemd`)
 - par défaut : `openrc`

[mirror]

- `url` : URL du miroir de téléchargement du stage3
 - par défaut : `https://mirror.init7.net`

[portage]

- `profile` : profile Gentoo utilisé par le système
 - par défaut : `default/linux/amd64/17.1`

[timezone]

- `timezone` : fuseau horaire / zone géographique

- par défaut : Europe/Zurich

[kernel]

- config : (partiellement implémenté) comment configurer le kernel (implémenté et testé : distkernel / implémenté non testé : genkernel / non implémenté : configfile)
 - par défaut : distkernel
- distkernel : paquet distkernel à utiliser (paquet portage sys-kernel/*)
 - par défaut : kernel-gentoo-bin

[system]

- hostname : nom de la machine
 - par défaut : gentoo
- logger : outil de logging (paquet de app-admin/*)
 - par défaut : syslogd
- cron : outil cron (paquet de sys-process/*)
 - par défaut : cronic

[boot]

- mode : mode de boot (implémenté et testé, requiert une partition bios : bios , partiellement implémenté / non-testé : efi)
 - par défaut : bios
- bootloader : système de boot (implémenté et testé (en mode bios) : grub , non implémenté : lilo , efibootmgr)
 - par défaut : grub

Valeurs configurables dans disks.conf

Le fichier se compose de deux parties.

Une section [general] qui contient des informations sur le disque à proprement parler et a les options suivantes:

- size : taille du disque (en MiB)
- drive : chemin du disque physique dans linux, donc dans /dev

Il peut ensuite y avoir une quantité arbitraire de catégories représentant chacune une partition. Une partition peut avoir les propriétés suivantes:

- size : taille de la partition (en MiB)
 - Si valeur 0 ou moins : prend le reste de l'espace libre du disque

- `mountpoint` : chemin de montage de la partition
 - Peut-être vide (par exemple pour le swap ou pour la partition biosboot)
- `filesystem` : nom du système de fichier à utiliser
 - Peut-être vide (par exemple pour le swap ou pour la partition biosboot)
 - Si `swap` alors la partition sera utilisée comme swap et sera traitée différemment
- `bootable` : est-ce que la partition doit être marquée comme démarrable?
- `biosboot` : pour les systèmes utilisant un boot en mode BIOS mais avec une partition en mode GPT, indique la partition (de 2MiB) à utiliser pour mettre les informations relatives au boot. Utilisé par GRUB.

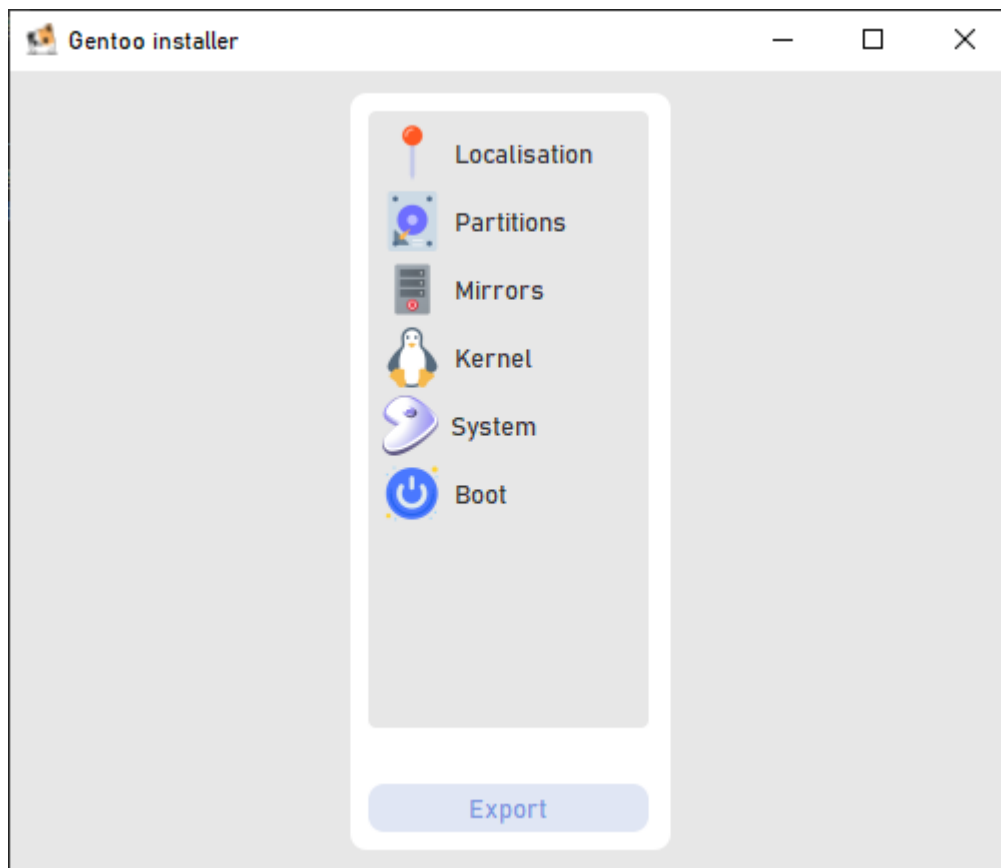
Générateur de configuration graphique

Dans le but de faciliter à des utilisateurs lambda la configuration de leur OS Gentoo, nous avons créé un installateur doté d'une interface graphique.

Pour lancer l'installateur, il suffit de lancer le script `main.py` se trouvant dans `/src/gui/` avec python.

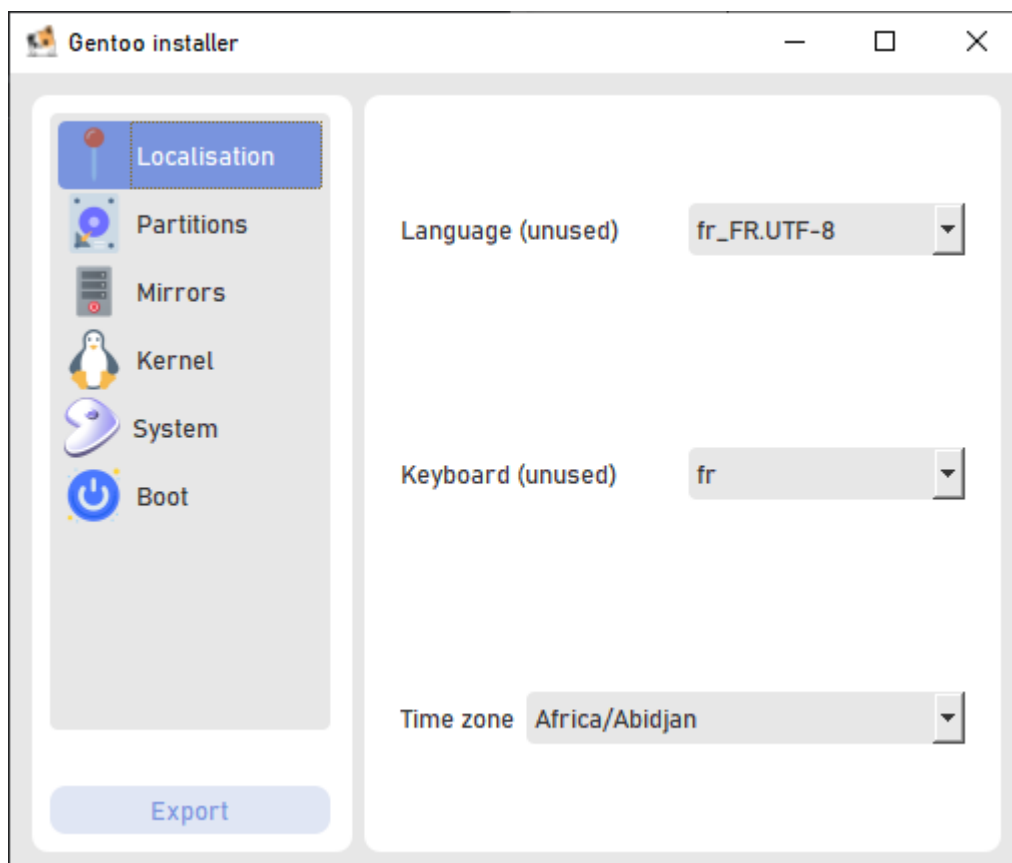
L'installateur a besoin d'une connexion internet pour être lancé, car celui-ci va récupérer certains choix de configuration, comme les miroirs, directement sur le site de Gentoo.

Cet installateur permet de générer des configurations dans plusieurs catégories, citées dans la partie précédente de ce document : localisation, partitions, miroirs, kernel, système, boot.



Localisation

La section localisation exporte les catégories `[locale]` et `[timezone]` dans le fichier `installer.conf`. Comme indiqué précédemment ainsi que sur l'interface, seule la `timezone` est pour le moment utilisée par l'installateur Gentoo.



Partitions

La section partition exporte les différentes partitions dans le fichier `disks.conf`. L'élément supérieur définit les paramètres du disque dans la section `[general]`. Chaque partition ajoutée en dessous ajoutera une section avec son attribut `name` comme label : `[name]`.

La taille de la dernière partition peut être laissée à 0 et sera calculée dynamiquement par l'installateur.

Gentoo installer

Localisation

Partitions

Mirrors

Kernel

System

Boot

Export

Size: 0 MiB

Drive:

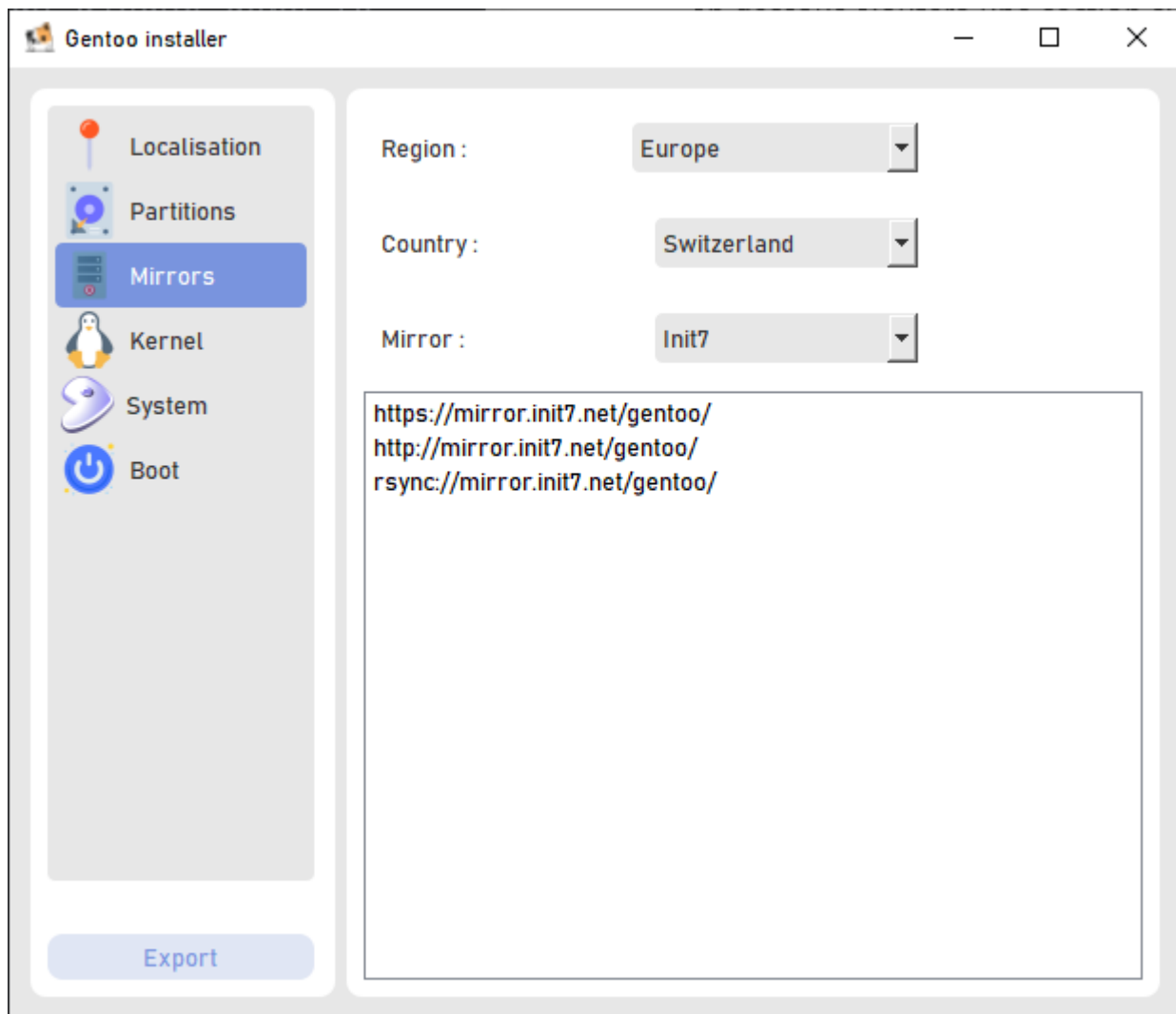
Label: gpt

Add

Partition	Partition
Name	Name
Size: 0 MiB	Size: 0 MiB
Mount point	Mount point
File system	File system
Bootable <input type="checkbox"/>	Bootable <input type="checkbox"/>
Bios boot <input type="checkbox"/>	Bios boot <input type="checkbox"/>
Remove	Remove

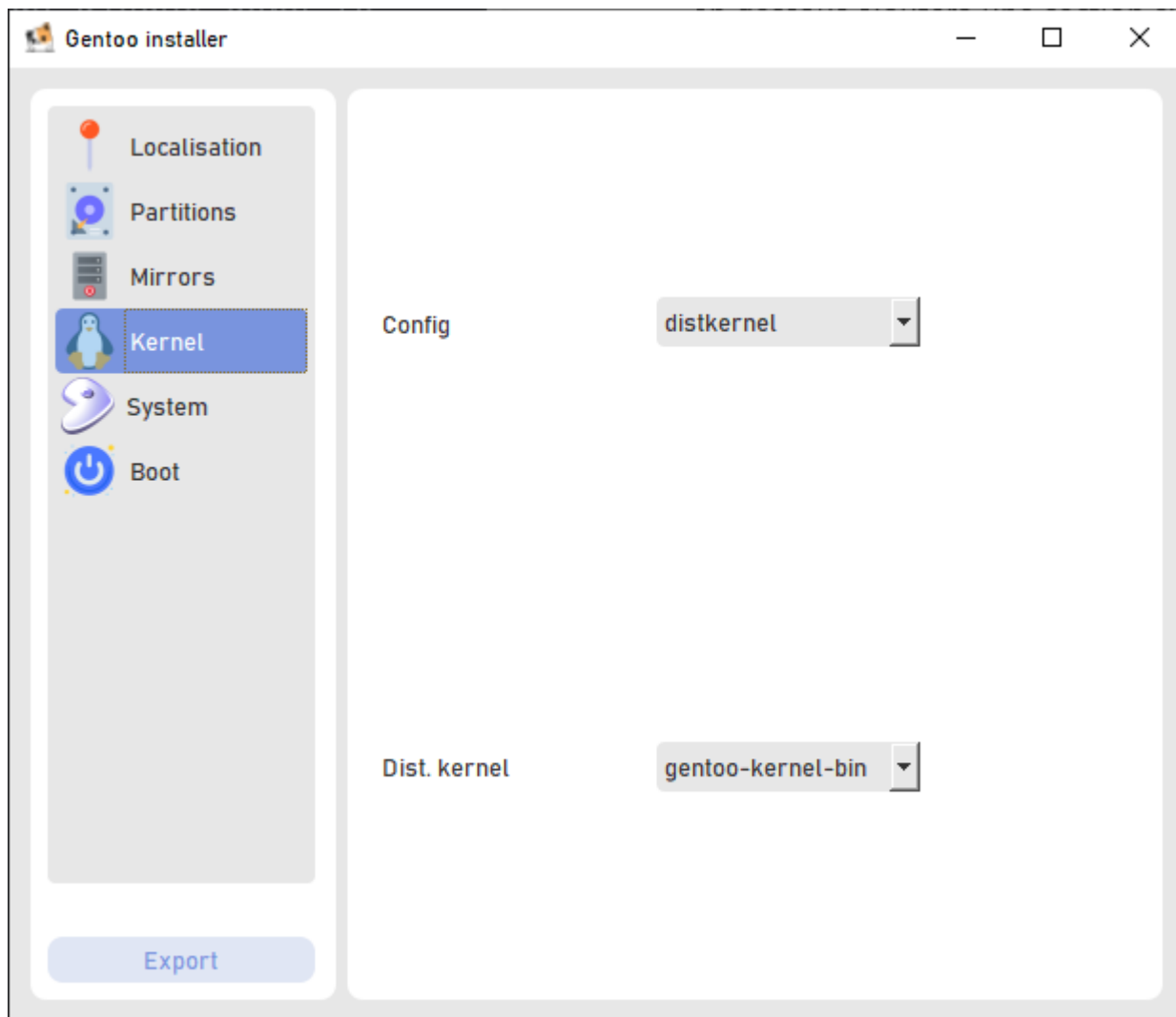
Mirrors

La section mirrors exporte la catégorie `[mirror]` dans le fichier `installer.conf`. Les miroirs sont récupérés à chaque lancement du programme sur l'api Gentoo afin d'être proposés à l'utilisateur.



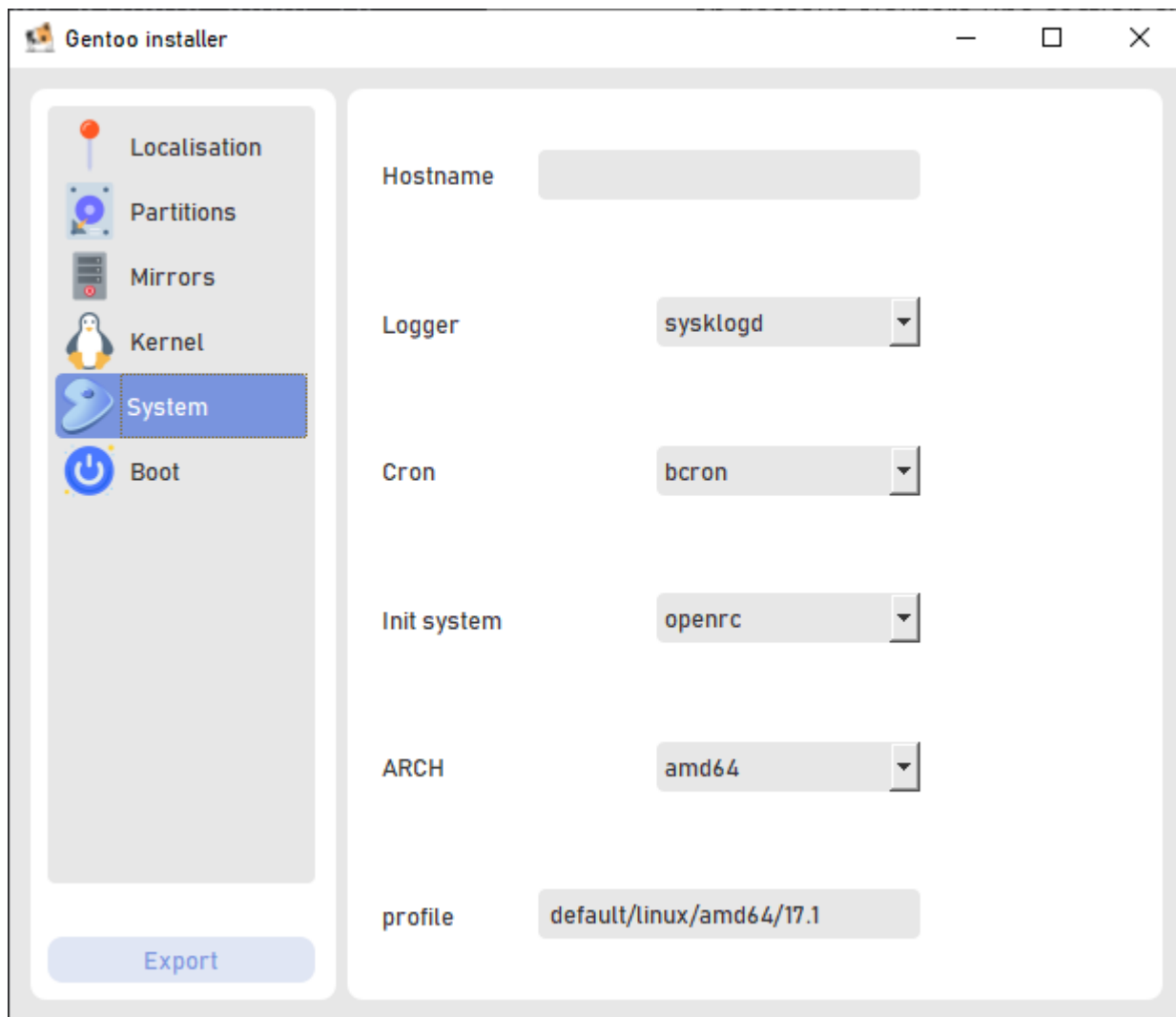
Kernel

La section kernel export la catégorie `[kernel1]` dans le fichier `installer.conf`.



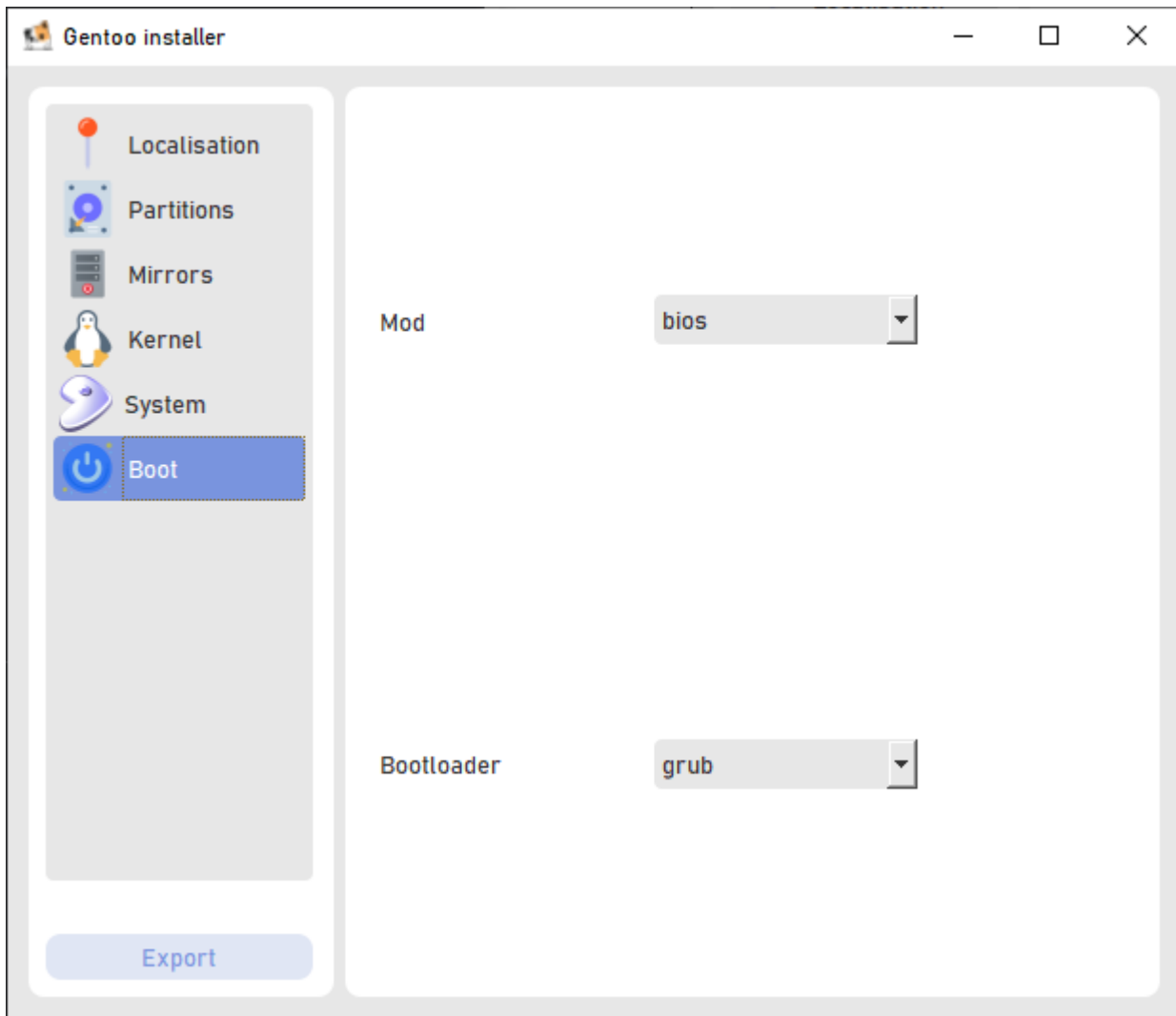
System

La section system export les catégories `[system]` , `[gentoo]` et `[portage]` dans le fichier `installer.conf` .



Boot

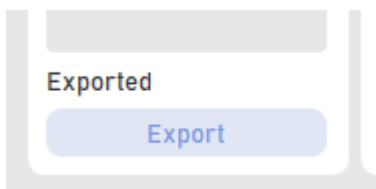
La section boot export la catégorie `[boot]` dans le fichier `installer.conf`.



Exportation

Une fois tous les champs à disposition remplis, et satisfait de la configuration créée, il vous faut l'exporter dans les deux fichiers requis par l'installateur. Pour cela, il vous suffit de cliquer sur le bouton "Export" en bas à gauche.

Une fois la configuration exportée, un message s'affiche pour vous prévenir (ce qui en réalité ne prendra pas plus d'une seconde).



Les fichiers exportés se trouvent alors dans `/src/gui/installer/` sous les noms `disks.conf` et `installer.conf`. Ces fichiers contiennent alors toute la configuration que vous avez pu spécifier grâce à l'interface graphique.

Les fichiers ainsi générés sont sous la forme :

```
[category1]
option1=something
option2=something
option3=something
```

```
[category2]
option1=something
option2=something
```

...

Il vous faudra par la suite fournir ces deux fichiers à l'installateur Gentoo pour qu'il puisse créer votre OS en utilisant votre configuration souhaitée. Si vous lancez le programme graphique via `python3 src/gui/main.py` les fichiers seront exportés directement dans `config/*` ce qui est le dossier lu par l'outil qui génère le script d'installation.

Générateur du script d'installation

Pour rappel, la création de l'installateur a plusieurs phases. Il faut premièrement créer les fichiers de configuration comme expliqué plus haut, soit à la main, soit à l'aide de l'interface graphique. Une fois cette configuration générée, elle est lue par un autre programme qui va se charger de créer un script shell contenant les commandes à effectuer sur la machine cible, ainsi que plusieurs autres fichiers. Le tout sera ainsi compressé dans une unique archive que l'on pourra copier sur le système à installer. Celui-ci devra au préalable être démarré avec le disque d'installation minimal fourni par Gentoo (<https://www.gentoo.org/downloads/>).

Nous traitons ici de la génération du script d'installation qui se découpe en plusieurs étapes que voici :

- Lecture des fichiers de configuration
 - Le script chargé de la génération du script d'installation va lire les fichiers de configuration `installer.conf` et `disks.conf` et charger les paramètres lus en mémoire
 - Si à la lecture, des paramètres manquent dans `installer.conf`, des valeurs par défaut vont être utilisées
 - Les partitions citées dans `disks.conf` sont stockées en mémoire.
 - La catégorie `[general]` du fichier `disks.conf` ne représente pas une partition, mais définit les valeurs de configuration du disque globale
- Génération du fichier de partition `sfdisk`
 - Une classe spéciale `SFDisk` génère ensuite un fichier au format lisible, par la commande `sfdisk`, contenant toutes les partitions
 - Par exemple:

```
label: gpt
device: /dev/sda

/dev/sda1 : start=2MiB,size=2MiB,bootable,type=21686148-6449-6E6F-744D
/dev/sda2 : start=4MiB,size=1024MiB,bootable
/dev/sda3 : start=1028MiB
```

- Génération des scripts d'installation
 - Deux scripts d'installation sont générés
 - `install.sh` qui se charge de partitionner le disque, télécharger le système de base (stage 3, une archive gentoo qui contient tout le système de base dans lequel il est possible de chroot), l'y décompresser et copier toute la configuration dans le nouveau système de fichiers
 - `chroot.sh` script qui se charge de l'installation même, et s'exécute dans le chroot directement
 - Le chroot se trouve dans `/mnt/gentoo` comme indiqué dans le Handbook Gentoo (<https://wiki.gentoo.org/wiki/Handbook:AMD64>)
- Copie des fichiers contenus dans le dossier `etc` du projet (pas le dossier `/etc` de l'installation, mais bien le dossier qui se trouve dans le projet avec le code) dans le dossier `autogen` à la racine du projet
- Copie des scripts d'installation dans le dossier `autogen`
- Compression du dossier `autogen` dans une archive `install.tar.gz` que l'on pourra mettre sur le système cible pour l'installation

How to use

Voir fichier externe fourni (situé dans `docs/userguide/` à la racine du projet, fichier `userguide.md`).

Conclusion

Actuellement, l'installateur permet de générer et tester quelques configurations que nous avons pu valider, ainsi que la création d'une configuration par défaut. Cependant, ces configurations restent basiques et ne peuvent être aussi précises/pointues que ce que Gentoo permet. En effet, à travers ce projet nous avons pu finaliser un *proof of work* qui, du fait de son architecture, pourrait être amélioré afin de compléter les différentes situations manquantes.