Building the image

There are several ways to build the image but all do rely on the central git repository at

Hey Alter / heyalter-ubuntu-iso · GitLab

Build on git

Simply login and start a build. You have to have an account to accomplish this.

Build locally

There are two ways to build locally. Both require to download the build environment, though. The snippet below will download the "master" branch. If you want to access another branch please read the git documentation.

```
git clone https://gitli.stratum0.org/heyalter/heyalter-ubuntu-iso.git
cd heyalter-ubuntu-iso
```

You can run **./build-local.sh** directly or use Docker with **./dockerbuild.sh.**

build-local.sh

If you want to run **build-local.sh** be aware that this has to be run on a plain vanilla Ubuntu image and it has to be the same you want to use as base image.

The build process downloads the image, exports it and uses apt-get to download additional packages with the necessary prerequisites. As you know apt-get will only install/download dependencies that are not yet installed.

dockerbuild.sh

So it is much simpler to use **dockerbuild.sh** instead. Running it the first time docker will be installed and the group docker will be added to the current userid. You have to logout/login to make it work, though.

File and directory structure

To be able to tinker with the build process you have to know the whereabouts. First all files involved directly for the build itself.

Name	Content/Remarks
build-local.sh	Main build script, also invoked by dockerbuild.sh<br>Invoking it will remove the old build and artifacts<br>directory but keeps the downloaded iso. <br>You have to remove it manually.
build.sh	called from build-local.sh to call all build steps
scripts (dir)	contains the build steps - 010_create-releasefile.sh - 020_install-host-software.sh - 030_download-iso.sh - 040_extract-iso.sh

	- 050_copy-files.sh - 055_download-extra-content.sh - 060_download_packages.sh - 100_create-iso.sh

scripts/env.sh	defines the environment variables,
	any global changes should happen here,
	espacially the Ubuntu version is defined.
	#
	# Version variables to avoid changes in several locations
	MAJOR="24"
	MINOR="04"
	NAME="noble"
	VERSION="$MAJOR.$MINOR"
	#
	DOWNLOAD_URL="https://releases.ubuntu.com/$NAME/ubuntu-$VERSION-desktop-amd64.iso"

build (dir)	temporary directory for the build process
artefacts (dir)	resulting iso and release file are stored here
files (dir)	contains files to manage the installation process.
files/grub.cfg	grub configurration for the installation
files/*.yaml	yaml files are referenced in grub.cfg They are used to automate the installation. heyalter_install.yaml heyalter_manual.yaml heyalter_nolvm.yaml
install_heyalter.sh	referenced in the yaml files this script is invoked at the end of the installation process