roadmap

Duzy Chan

March 4, 2016

Contents

1 Making a base root filesystem :CC:Base:

The bootstrap is creating a basic root filesystem (rootfs). Further efforts should be done in the base. The base is providing an identical environment to everyone (devs, geek, hacker, etc) involved.

1.1 Root filesystem :CC:Base:RootFS:

This is the base system. It should boot into a shell terminal with networks and Debian APT supported.

1.1.1 The package manager :CC:Base:APT:

With the networking enabled for the base, a fully funtional debian package manager should be configured. This is the basic facility to extend the system to different variants.

- Using the official debian package source **:CC:Base:APT:OriginSource:**
  While using the official debian package source, we derived the hardware compatibilities from Debian. Say, if Debian is able to run on Odroid board, our distro could too.
  - Compatibility Verification
    Some POC (proof-of-concept) efforts could be spent to examing the hardward compatibility. If we're going to provide very serious hardware supports, this might require lots of efforts.
- Build specific package source server for Community Cube **:CC:Base:APT:SpecificSource:**
  This is actually optional, without this, the official debian package should work. The specific source server obviously require extra resources, it could expend into lots of works to do.
  - Package error fixing or refining or customization.
    Per package (apt) refining is possible with the our own specific package source host.

1.2 Methods to work on the rootfs (development purpose) :CC:Base:Dev:

We're going to use QEMU quickly work in the base. This will give others (devs or someone from the community) a way to get involved. We also need a managed way to assembly a good result from the efforts of the community.

1.2.1 Very fundamental packages :CC:Base:Dev:Initialization:

Some packages might be widely used by other packages or in real life. These packages could be installed before futher installation.

1.2.2 A method to wrap external efforts :CC:Base:Dev:Modulization:

An external effort could be contributed by anyone from the community. We need to assembly these individual contribution.

- A module manager to assembly individual efforts **:CC:Base:Dev:Modulization:Manager:** A well designed manager to assembly and build external contribution into the system. A contribution should be wrapped into a **module**.
- A common way to contribute a module **:CC:Base:Dev:Modulization:Interfaces:** The devs, external individuals are trying to contribute a valuable efforts to the system. The job needs to be done in a common way. Our own customization and parameterization should also be done in such way. So that we will enable different forces to work on different kind of task, and tested in the development QEMU. If some task requires to be tested in a real environment, it could be done after the integration.

1.3 To run in a physical machine :CC:Base:Physical:

We will need the ISO image and burn it to a DVD or USB stick for installation on a Intel or AMD machine.

2 Customization and Parameterization over the base

This part will be ignored for now. This could be done by modules and the module manager.

3 Creating an installer for the prepared distribution

3.1 Installer same as https://wiki.debian.org/DebianInstaller

3.2 Making ISO image with the Installer