OpenEmbedded – A layered Approach

Khem Raj
Embedded Linux Conference
15-17 Feb 2012, Redwood Shores
Agenda

- OpenEmbedded – The classic way
- New Layered structure
- Existing layers
- Challenges with layers
- Q & A
OpenEmbedded Classic way

- All metadata together under one tree
  - Plethora of package recipes
  - Multiple versions of recipes
  - Numerous machines
  - Many distros
- push model for SCM
- Lot of pinning needed
- Release 2011.03 is still maintained
OpenEmbedded Classic Way

- Challenges
  - Lot of dead code.
  - Unmaintained recipes and machine
  - Push model broke build more often
Distribution – New way

Feature layers
Machine/BSP layer
Openembedded Core layer
Distribution policy layer
Tools

- Bitbake has .bbappend feature
  - Patch recipes
- Combo-layer
- bitbake-layers
Patching recipes

- `<recipe>.bbappend` can be used in layers to modify recipe data
- It can not patch `.inc` or class files
- Typically used for tweaking few settings
- Changing configure options, CFLAGS etc.
usage: bitbake-layers <command> [arguments]

Available commands:

flatten
  flattens layer configuration into a separate output directory.

help
  display general help or help on a specified command

show-appends
  list bbappend files and recipe files they apply to

show-layers
  show current configured layers

show-overlayed
  list overlayed recipes (where the same recipe exists in another layer that has a higher layer priority)

show-recipes
  list available recipes, showing the layer they are provided by
Usage: combo-layer [options] action

Create and update a combination layer repository from multiple component repositories.

Action:

init  initialise the combo layer repo
update get patches from component repos and apply them to the combo repo
splitpatch [commit] generate commit patch and split per component, default commit is HEAD

Options:

--version  show program's version number and exit
-h, --help  show this help message and exit
-c CONFFILE, --conf=CONFFILE
    specify the config file (conf/combo-layer.conf is the default).
-i, --interactive  interactive mode, user can edit the patch list and patches
-D, --debug  output debug information
OpenEmbedded – Layered

- Divide metadata into set of maintainable layers
  - Core layer
    - Provide common metadata
    - Recipes needed almost on all systems
    - Reference Emulation machines
    - Distroless
  - Meta-openembedded
    - Umbrella for OpenEmbedded hosted layers
    - meta-oe
OpenEmbedded – Core layer

- Provide common metadata
  - classes
  - conf
- Recipes needed almost on all systems
- Reference Emulation machines
  - qemuarm, qemumips, qemuppc, qemuux86,qemuux86-64
- Distroless
- Self contained
- Reference images and common tasks
Existing Layers

- Openembedded-core
  - Core meta data
- meta-oe
- meta-gnome
- meta-xfce
- meta-mozilla
- meta-java
- meta-micro
layers

- Distribution layers
  - Angström, SHR, micro, slugos, poky ...
- Board support layers
  - efikamx, NSLU2 ...
- Feature layers
  - Java, EFL, GNOME, GPE, XFCE ...
- Vendor layers
  - meta-texasinstruments, meta-intel, meta-xilinx ...
Further Information existing layers

http://www.openembedded.org/wiki/LayerIndex
Layer Ettiquettes

- Top level README
  - Describe inter layer dependencies
  - Setup instructions to include in an integration layer
  - Patch flow and process
  - Maintainers
  - Copyright Notice
Existing layers

- Meta-smartphone
- meta-handheld
More layers

- Plasma Active
- Linaro
- Oracle Java
- More ....
Challenges

- Layer interactions
- Recipe duplication
- Feature conflicts
- Diagnostics
  - Which recipe gets picked
- Overcrowded layers
Challenges

- Layering tools
- SCM
- Classes
- Layer Release
- Distribution release
- Metadata synchronization
Where are we

- Distributions like poky, Angström SHR already using the layered model.
- All recipes from Classic Openembedded are not available.
  - Ported as needed
Thank you