OpenBMC Introduction and Porting Guide

Saravanan Palanisamy – Feb 6, 2021

saravanan.Palanisamy@linux.intel.com

Agenda

- OpenBMC Introduction
- Key features
- Make your first OpenBMC build and run it in QEMU
- Make your first customized build for your new platform (based on Intel WolfPass)
- Expose your new platform properties using EntityManager in systemd (D-Bus)
- References

OpenBMC Introduction

- Linux Foundation project.
- Goal: produce a customizable, open-source firmware stack for Baseboard Management Controllers (BMCs).
- BMC = an autonomous micro-controller operates on standby power, placed besides the main processor in the server board, that is always on and maintaining the system, independent of Host machine CPU & its software. It also has NV storage to store SDR, SEL, FRU.
- BMC Key Functionalities: Monitoring(Electrical, Thermal, etc), Recovery, Logging, Alerting, Inventory.
- FOSDEM 2020 Talk - OpenBMC and RedFish Introduction

OpenBMC Key Features

- Linux Distribution : Yocto
- Language Choice : Modern CPP (C++17 and beyond), Python, JSON
- IPC Mechanism : system (D-Bus)
- Managebility Protocols : IPMI, RedFish

Getting Started

- OpenBMC Developer Documentation

  https://github.com/openbmc/docs

- OpenBMC Presentations

  https://github.com/openbmc/openbmc/wiki/Presentations

- Cheatsheet

  https://github.com/openbmc/docs/blob/master/cheatsheet.md

Yocto

- ➢ Yocto – Open Source project that delivers set of tools to create Embedded Linux OS images.
- ➢ OS image – Analogy to Layered Cake ☺
- ➢ Layers – Groups related functionalities. Prefixed with 'meta-'. Ex.) meta-poky, meta-ast2500, meta-phosphor,….
- ➢ Poky – Reference Embedded Linux OS distribution given by Yocto.
- ➢ Bitbake – Python based task scheduler. Execution engine for baking the Yocto OS ☺.
- • Recipes - contains instructions the build system uses to create packages. Ex.) Repository URL

Photo courtesy : (Photo by Miti on Unsplash)

Your first OpenBMC WolfPass(Intel) build - QEMU

OpenBMC

- OpenBMC Development Environment - https://github.com/openbmc/docs/blob/master/development/dev-environment.md

- Toaster (WebUI for Yocto) – https://docs.yoctoproject.org/toaster-manual/setup-an

(pip3 install --user -r ./poky/bitbake/toaster-requirements.txt)

- QEMU

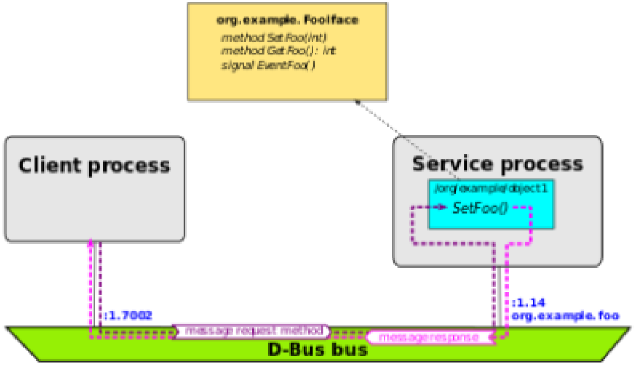
Intel® Server Board S2600WF Family

Build Commands

```
#Check my Ubuntu release version
cat /etc/os-release | grep PRETTY

#Clone OpenBMC Repo
git clone https://github.com/openbmc/openbmc.git
cd openbmc/
 git tag
git checkout 2.9.0 -b my-first-build

#Setup build environment for WolfPass
. setup
. setup  s2600wf build

#Start the toaster
source toaster start webport=10.190.201.114:9998

#Start the image compilation.
bitbake obmc-phosphor-image

#Download the QEMU
cd tmp/deploy/images/s2600wf/
wget https://jenkins.openbmc.org/job/latest-qemu-x86/lastSuccessfulBuild/artifact/qemu/build/qemu-system-arm
chmod u+x qemu-system-arm

#Start the QEMU on compiled output image
./qemu-system-arm -m 256 -M ast2500-evb -nographic -drive file=./obmc-phosphor-image-s2600wf-*.static.mtd,format=raw,if=mtd -net nic -net user,hostfwd=:127.0.0.1:2222-:22,hostfwd=:127.0.0.1:2443-:443,hostname=qemu
```

Systemd

- https://dbus.freedesktop.org/doc/dbus-spec
- D-Bus : IPC mechanism for applications resic
- 2-types of buses : system bus, session bus
- 4-types of messages:
  METHOD_CALL, METHOD_RETURN, ERROR, and SIGNAL
- Service : ~ daemon, ~application.
- Interface : Namespace for target object.
- Object path : Target object's path.
- busctl utility



Example of one-to-one request-response message exchange to invoke a method over D-Bus. Here the client process invokes the `SetFoo()` method of the `/org/example/object1` object from the service process named `org.example.foo` (or `:1.14`) in the bus.

EntityManager

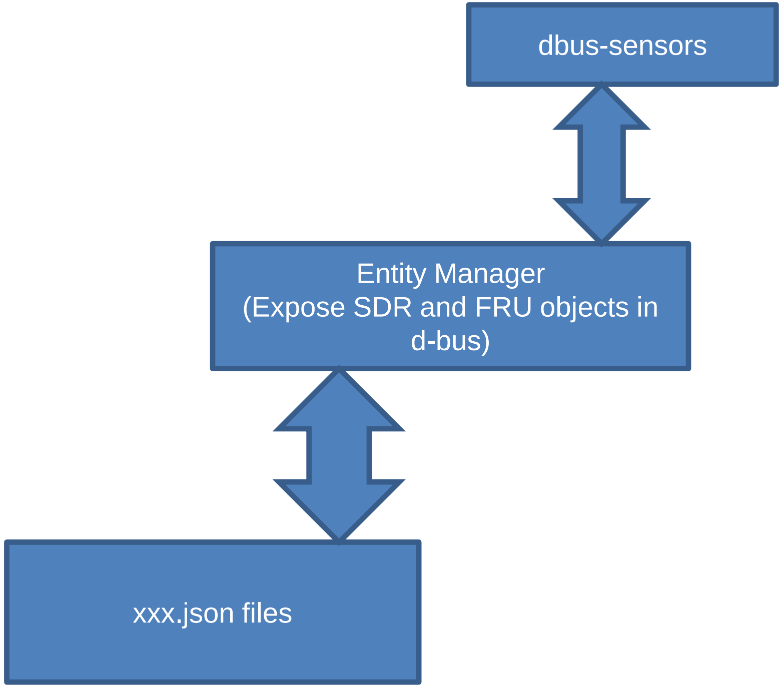
https://github.com/openbmc/entity-manager

Entity manager is a runtime configuration application which parses configuration files (in JSON format) and produces a best representation of the files on dbus using the xyz.openbmc_project.Configuration namespace. It also produces a system.json file for persistance.

https://github.com/openbmc/dbus-sensors

dbus-sensors is a collection of sensor applications that provide the xyz.openbmc_project.Sensor collection of interfaces. They read sensor values from hwmon, d-bus, or direct driver access to provide readings. Some advance non-sensor features such as fan presence, pwm control, and automatic cpu detection (x86) are also supported.

EntityManager Service



Actual WolfPass baseboard JSON file

- https://github.com/openbmc/entity-manager/blob/master/configurations/WFT%20Baseboard.json

```
….
 "Exposes": [
         {
       {
           "Index": 1,
           "Name": “P3V3",
           "ScaleFactor": 0.4107,
           "Thresholds": [
               {
                   "Direction": "greater than",
                   "Name": "upper critical",
                   "Severity": 1,
                   "Value": 3.647
               },
                           …..
           ],
           "Type": "ADC"
        },
………
                    ]
 }
 "Name": "WFP Baseboard",
 "Probe": "xyz.openbmc_project.FruDevice({'PRODUCT_PRODUCT_NAME': '.*WFT'})",
"Type": "Board",
    "
```

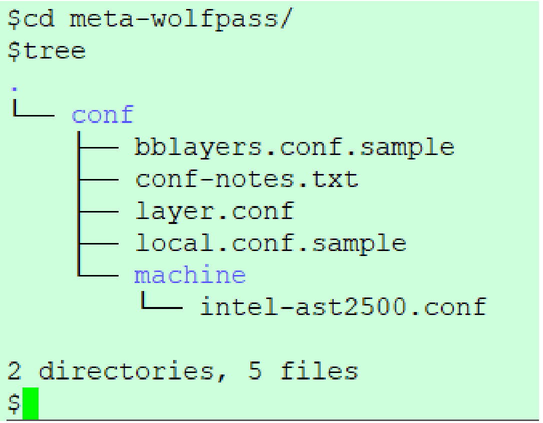
Our custom ‘FOSDEM_2021.json’ file

- **FOSDEM_2021.json file**

```
….
 "Exposes": [
        {
      {
         "Index": 1,
         "Name": "P3V3_FOSDEM_2021",
         "ScaleFactor": 0.6666,
         "Thresholds": [
            {
               "Direction": "greater than",
               "Name": "upper critical",
               "Severity": 1,
               "Value": 3.888
            },
                        …..
         ],
         "Type": "ADC"
      },
………
                ]
 }
 "Name": "WFP Baseboard",
 "Probe": "xyz.openbmc_project.FruDevice({'PRODUCT_PRODUCT_NAME': '.*FOSDEM'})",
"Type": "Board",
    "
```

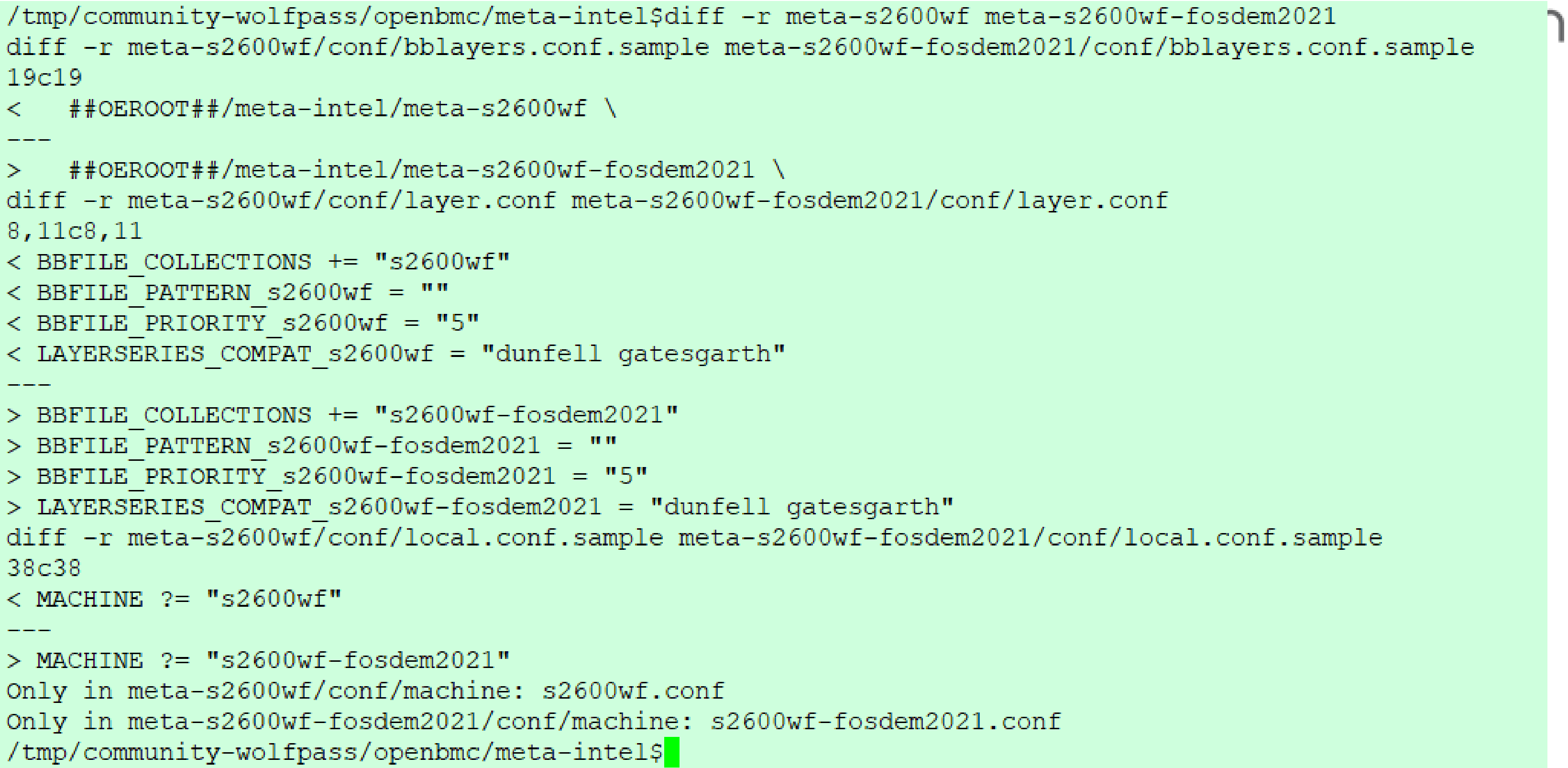
Intel-BMC/OpenBMC – Custom machine WolfPass - FOSDEM

- Let us create an Yocto Distribution for our custom machine ‘wolfpass-fosdem’.
- Let us assume ‘wolfpass-fosdem’ is also based ASPEED AST2500 and no changes in Device Tree. (*.dts, *.dtb) files.
- Create ‘FOSDEM_2021.json’ file in EntityManager which exposes board properties present in this file in D-Bus.



```
$cd meta-wolfpass/
$tree
.
└── conf
    ├── bblayers.conf.sample
    ├── conf-notes.txt
    ├── layer.conf
    ├── local.conf.sample
    └── machine
        └── intel-ast2500.conf

2 directories, 5 files
$
```

Intel-BMC/OpenBMC – Custom machine WolfPass - FOSDEM



```
/tmp/community-wolfpass/openbmc/meta-intel$diff -r meta-s2600wf meta-s2600wf-fosdem2021
diff -r meta-s2600wf/conf/bblayers.conf.sample meta-s2600wf-fosdem2021/conf/bblayers.conf.sample
19c19
<    ##OEROOT##/meta-intel/meta-s2600wf \
---
>    ##OEROOT##/meta-intel/meta-s2600wf-fosdem2021 \
diff -r meta-s2600wf/conf/layer.conf meta-s2600wf-fosdem2021/conf/layer.conf
8,11c8,11
< BBFILE_COLLECTIONS += "s2600wf"
< BBFILE_PATTERN_s2600wf = ""
< BBFILE_PRIORITY_s2600wf = "5"
< LAYERSERIES_COMPAT_s2600wf = "dunfell gatesgarth"
---
> BBFILE_COLLECTIONS += "s2600wf-fosdem2021"
> BBFILE_PATTERN_s2600wf-fosdem2021 = ""
> BBFILE_PRIORITY_s2600wf-fosdem2021 = "5"
> LAYERSERIES_COMPAT_s2600wf-fosdem2021 = "dunfell gatesgarth"
diff -r meta-s2600wf/conf/local.conf.sample meta-s2600wf-fosdem2021/conf/local.conf.sample
38c38
< MACHINE ?= "s2600wf"
---
> MACHINE ?= "s2600wf-fosdem2021"
Only in meta-s2600wf/conf/machine: s2600wf.conf
Only in meta-s2600wf-fosdem2021/conf/machine: s2600wf-fosdem2021.conf
/tmp/community-wolfpass/openbmc/meta-intel$
```

References

- FOSDEM 2020 Talk - OpenBMC and RedFish Introduction
- OpenBMC Developer Documentation - https://github.com/openbmc/docs
- OpenBMC Presentations - https://github.com/openbmc/openbmc/wiki/Presentations
- OpenBMC Cheatsheet - https://github.com/openbmc/docs/blob/master/cheatsheet.md
- Yocto Toaster - https://docs.yoctoproject.org/toaster-manual/setup-and-use.html
- D-Bus - https://dbus.freedesktop.org/doc/dbus-specification.html#introduction
- Systemd - https://www.wikiwand.com/en/D-Bus
- WolfPass JSON file - https://github.com/openbmc/entity-manager/blob/master/configurations/WFT%20Baseboard.json
- OpenBMC Community repo - https://github.com/openbmc/openbmc
- Intel OpenBMC public repo - https://github.com/Intel-BMC/openbmc
- Device Tree - https://www.kernel.org/doc/Documentation/devicetree/usage-model.txt
- Adding a new machine - https://www.youtube.com/watch?v=Cjj71ESJMoI , https://github.com/openbmc/docs/blob/master/development/add-new-system.md
- EntityManager - https://www.youtube.com/watch?v=avDr3FQiIMw
- Dbus-sensors - https://github.com/openbmc/dbus-sensors
- Intel WolfPass Platform

Thank You !!!

OpenBMC

Q & A