



**OPEN**  
Compute Project



**OCP U.S. SUMMIT 2017**

Santa Clara, CA



# The OpenBMC Project

Chris Austen / Senior Software Manager / IBM

**OPEN HARDWARE.**

**OPEN SOFTWARE.**

**OPEN FUTURE.**



# Devils in the Details



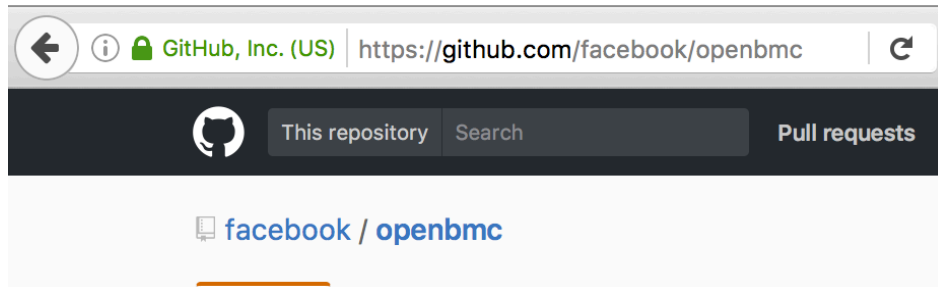
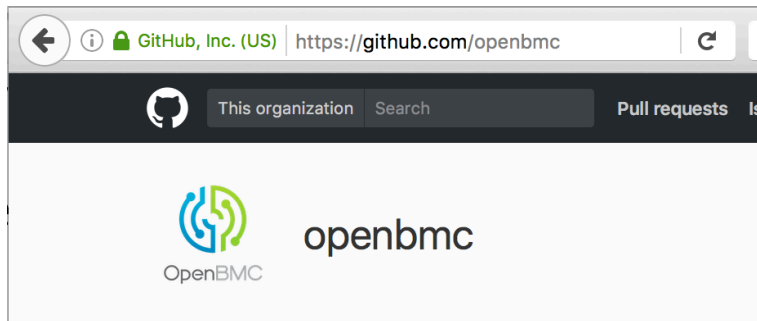




# BMC Development

	Outsourced	Home Grown	Open
Control of features	X	✓	✓
Fast bug fixes	X	✓	✓
Control of security updates	X	✓	✓
Cheaper to develop	✓	X	✓
Simple infrastructure support	✓	X	✓
Open Source Community Code Reviews	X	X	✓
Unique look	X	✓	✓

# Haven't I heard of this before?



# What is OpenBMC



Control the Functions in your OpenBMC...

- Built on Yocto-Linux

Simple infrastructure for you

- IPC via D-Bus
- Automatic REST API from D-Bus introspection
- Users via REST and IPMI





# What is OpenBMC

Contributions back to Open Source

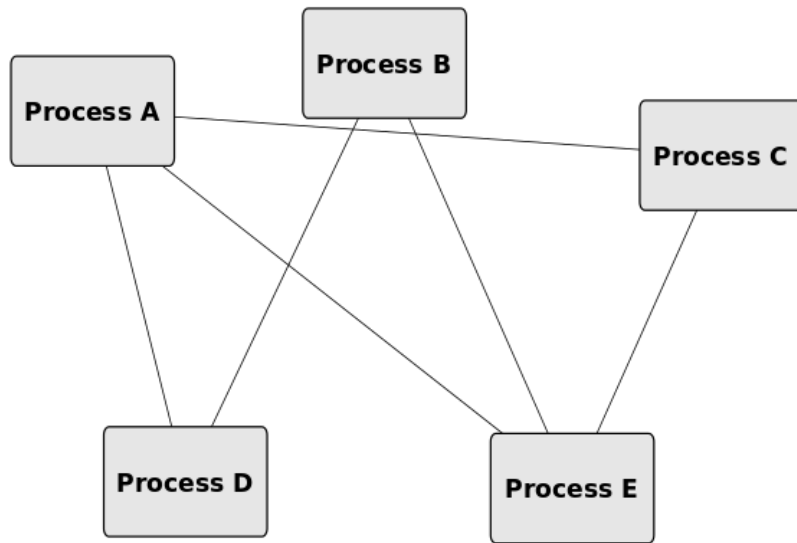
- Fully upstreamed U-Boot and Kernel
  - ASPEED AST2400 and AST2500

Struggling to keep up with latest technologies?

- U-Boot 2016
- Linux 4.10
- Yocto 2.1

# Without D-Bus

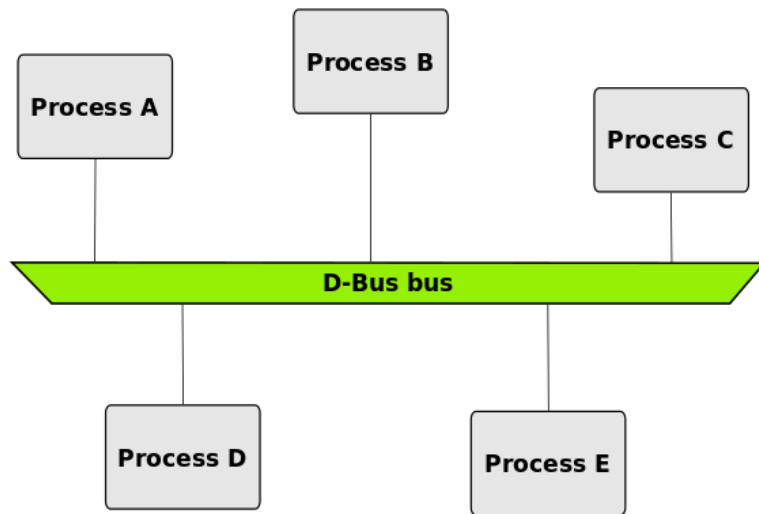
- Code all IPC yourself
- Startup / Recover logic
- Multi-company collaboration?



© 2015 Javier Cantero - this work is under the Creative Commons Attribution ShareAlike 4.0 license

# With D-Bus

- Rapid community collaboration
- Rapid Prototyping
  - Bindings for C/C++, Python, etc
- Compete IPC integration with systemd



© 2015 Javier Cantero - this work is under the Creative Commons Attribution ShareAlike 4.0 license

# OpenBMC Dbus Interface Examples

Attach an interface, get the benefits in Dbus and REST

`xyz.openbmc_project.Sensor.Threshold.Warning`

- Broadcast notification when value trips
- “Min”, “Max”

`xyz.openbmc_project.Inventory.Item`

- VPD Properties
- “Present”

# Making ODM development easier

## In 2 Ways...

---

### XML Integration

- Manifest for your hardware
- Define wiring
- Define environmental limits
- IPMI entity IDs

# XML Integration

```
<targetInstance>
  <id>MAX31785.hwmon2</id>
  <type>unit-hwmon-feature</type>
  ...
  <attribute>
    <id>HWMON_FEATURE</id>
    <default>
      <field><id>HWMON_NAME</id><value>fan1</value></field>
      <field><id>DESCRIPTIVE_NAME</id><value>fan0</value></field>
      <field><id>WARN_LOW</id><value>1000</value></field>
      <field><id>WARN_HIGH</id><value>80000</value></field>
      <field><id>CRIT_LOW</id><value></value></field>
      <field><id>CRIT_HIGH</id><value></value></field>
    </default>
  </attribute>
```

# Making ODM development easier

## In 2 Ways...

---

### XML Integration

- Manifest for your hardware
- Define wiring
- Define environmental limits
- IPMI entity IDs

### YAML definitions for REST

- Human readable properties
- Documentation driven code

# YAML, Code and Schema Documented

**description:** >

Implement to provide event/error entry attributes.

This interface should be instantiated for the phosphor::logging namespace.

This interface is a collection of objects, therefore it is required to implement org.freedesktop.DBus.ObjectManager on the logging namespace root. Optionally, implement org.openbmc.Object.Delete to allow the deletion of individual entries.

**properties:**

- **name:** Id

**type:** uint32

**description:** >

The error event entry id number.

- **name:** Timestamp

**type:** uint64

**description:** >

Commit timestamp of the error event entry in milliseconds since 1970.



# Defined REST Schema

- /
- /list
- /enumerate
- /attr/<property>
- /action/<method>
- /schema

# Lets see an example

```
curl -b cjar -k https://bmc/xyz/openbmc_project/inventory/
```

```
{  
  "data": [  
    "/xyz/openbmc_project/inventory/system"  
  ],  
  "message": "200 OK",  
  "status": "ok"  
}
```

# Lets see an example

```
curl -b cjar -k
https://bmc/xyz/openbmc_project/Inventory/system/chassis/motherboard/cpu0

{
  "BuildDate": "",
  "Manufacturer": "IBM",
  "Model": "",
  "PartNumber": "01HL322",
  "Present": 0,
  "PrettyName": "PROCESSOR MODULE",
  "SerialNumber": "YA3933741577",
  "Version": "EC:10"
}
```

# Quick Test Drive?



# QEMU == No Hardware Required

```
Phosphor OpenBMC (Phosphor OpenBMC Project Reference Distro) 0.1.0 qemuarm tty1
qemuarm login:      Starting Network Service...
[ OK ] Started Phosphor OpenBMC host IPMI to DBUS example implementation.
[ OK ] Started Serial Getty on ttyAMA0.
[ OK ] Started Phosphor OpenBMC QEMU application example.
[ OK ] Started Phosphor OpenBMC fan management.
      Starting SSH Key Generation...
[ OK ] Started Serial Getty on ttyAMA1.
[ OK ] Reached target Login Prompts.
      Starting Login Service...

Phosphor OpenBMC (Phosphor OpenBMC Project Reference Distro) 0.1.0 qemuarm tty1
qemuarm login:
```

```
causten@fiji: ~/workspace
causten@fiji:~/workspace$ curl -b cjar -k https://localhost:2443/xyz/openbmc_project/
{
  "data": [
    "/xyz/openbmc_project/sensors",
    "/xyz/openbmc_project/inventory",
    "/xyz/openbmc_project/.ObjectMapper",
    "/xyz/openbmc_project/software",
    "/xyz/openbmc_project/logging"
  ],
  "message": "200 OK",
  "status": "ok"
}
causten@fiji:~/workspace$
causten@fiji:~/workspace$ ssh -p 2222 root@localhost
root@localhost's password:
root@palmetto:~#
```

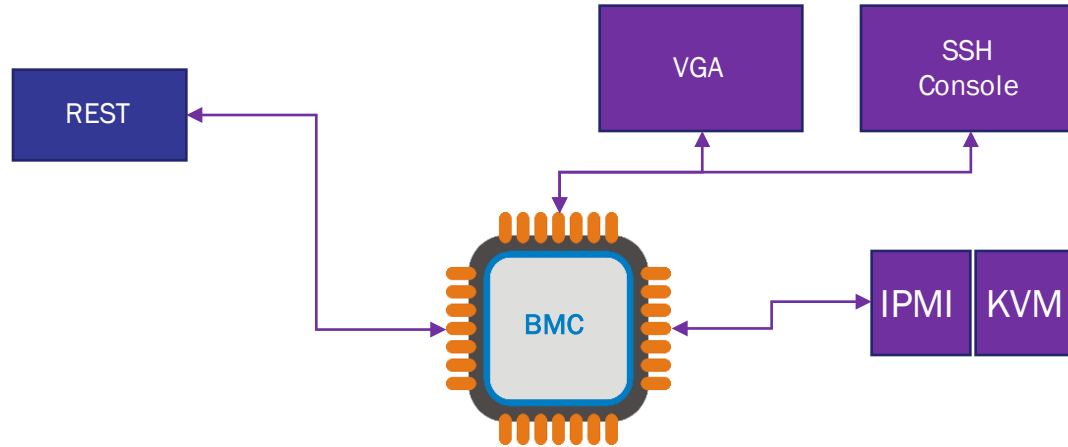
# Automated Build Verification

OpenPower continuous integration setup

S	W	Name ↓	Last Success	Last Failure	Last Duration
✓	☁	<a href="#">Intrinsics Build - buildroot</a>	9 days 0 hr - <a href="#">#12-3fc77e5f81906d</a>	13 days - <a href="#">#11-3fc77e5f81906d</a>	9 min 35 sec
✓	☁	<a href="#">Intrinsics Builds</a>	N/A	N/A	N/A
✓	☀	<a href="#">Linux Build - aspeed</a>	15 days - <a href="#">#4-35d2cfb27392c7</a>	N/A	7 min 21 sec
✓	☀	<a href="#">Linux Build - master branch</a>	8 hr 26 min - <a href="#">#155-e46b4e2b46e173</a>	N/A	6 min 53 sec
✓	☁	<a href="#">OpenBMC Build</a>	3 days 11 hr - <a href="#">#77-2fe86d90044af2</a>	17 days - <a href="#">#74-b182f7b07220bc</a>	1 hr 41 min
✓	☀	<a href="#">OpenBMC Build - Pull Request</a>	16 hr - <a href="#">#134-c9c000e1555fe3</a>	13 days - <a href="#">#125-331da0441b1cb8</a>	1 hr 3 min
✓	☀	<a href="#">OpenBMC Test - QEMU Boot</a>	15 hr - <a href="#">#183-</a>	10 days - <a href="#">#174-</a>	1 min 18 sec

- Dedicated Test Development Team
- Every commit... tested
- Enterprise quality

# Covered Interfaces -- 2016

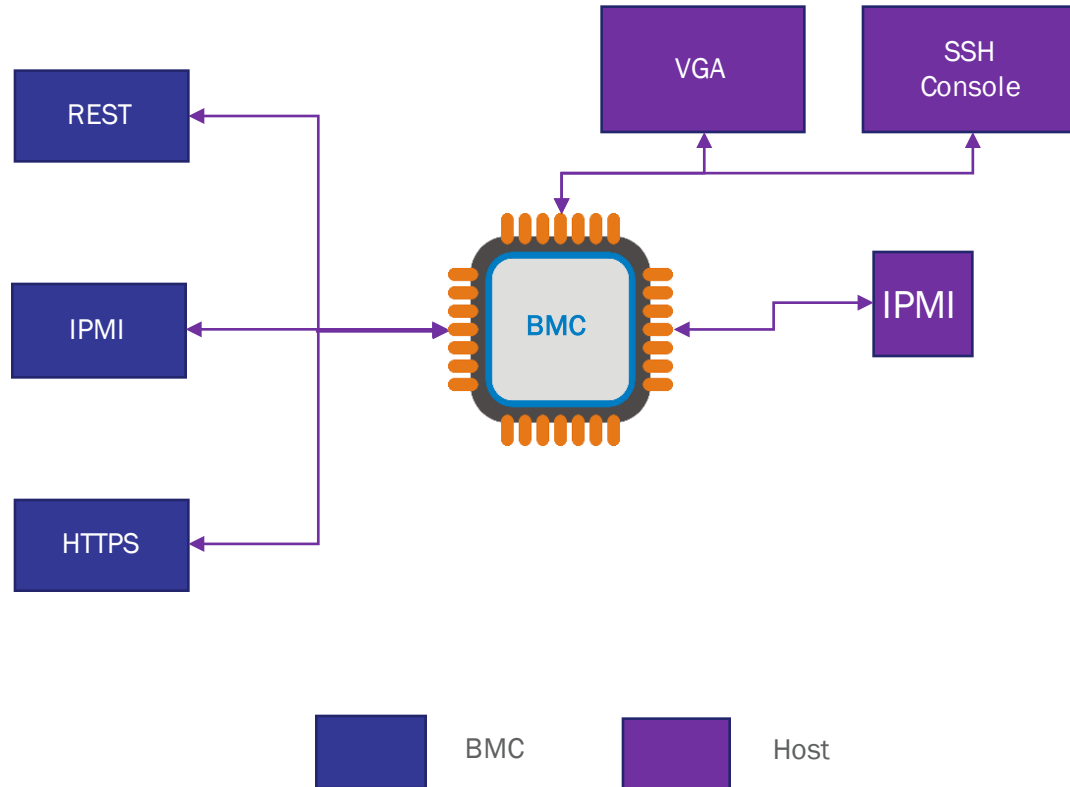


BMC



Host

# Covered Interfaces -- 2017





# Features Provided

- Power Control
- Remote Console
- IPMI
- Sensors
- LED Management
- Host Watchdog
- VPD Inventory
- Power and Cooling Management
- Event Logs
- Zeroconf discoverable
- Simulation

# Features in development

- Enhanced Code Update
- Full IPMI 2.0 Compliance with DCMI IPMI
- User Interface ← *looking for Sponsor Users*

# Features of the future... Join Us

- OpenCompute Redfish Compliance
- Remote KVM
- Remote USB
- Additional Board Support
- OpenStack Ironic Integration
- QEMU enhancements
- Quarantine Room for full Hardware CI
- Designs beyond the data center

# Thank You

Code:

- <https://github.com/openbmc>

Continuous Testing

- <https://openpower.xyz>

Contact

- Mail: [openbmc@lists.ozlabs.org](mailto:openbmc@lists.ozlabs.org)  
IRC: #openbmc on freenode.net
- Riot: [#openbmc:matrix.org](https://matrix.org/#openbmc)





**OPEN**  
Compute Project

