

Fosdem 2019

bmc`lib`

A Baseboard Management Controller library

One library to rule them all?

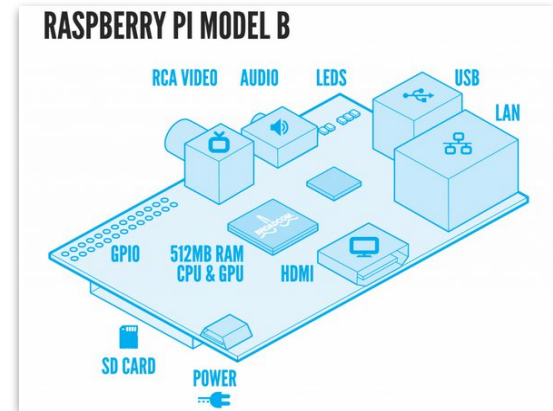
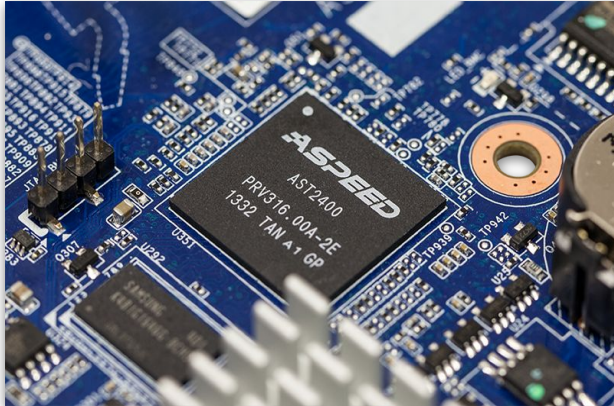
Juliano Martinez
Joel Rebello

Booking.com



Baseboard Management Controller

A BMC is a **system on chip** that integrates various computer components in a single die/package, roughly similar to the Broadcom SoC found on a Raspberry PI, except, the BMC runs within a server/chassis/switch/JBOD/JBOF with its own OS/Firmware



Various flavours, single function

Asset lifecycle management

- Out of Band access to servers/switches/JBODs, etc
- Last resort to power cycle, reboot, hard reset
- IPMI/VNC/iKVM/Serial console access
- Inventory information
- Hardware logs
- Root of trust

BMCs




**Hewlett Packard
Enterprise**

HPE iLO Advanced
Remote Management
(iLO2 - iLO3 - iLO4 - iLO5)



**Integrated Dell Remote
Access Controller 7**

 **Hewlett Packard
Enterprise**

HPE BladeSystem Onboard Administrator

DELL

CI ASSIS MANAGEMENT CONTROLLER

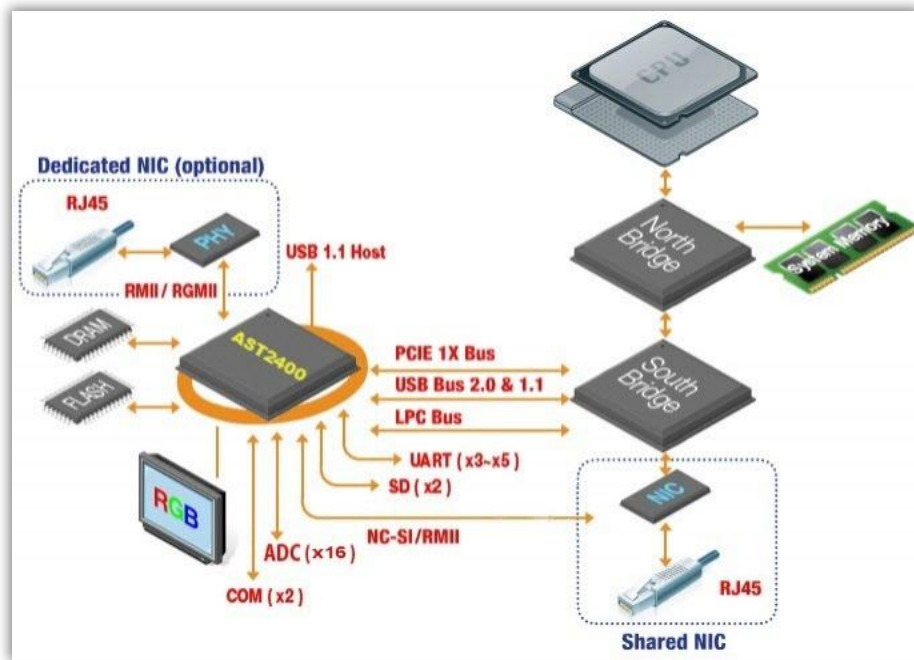


Quanta System Manager

Specs

Common SoC BMC – AST2400/AST2500

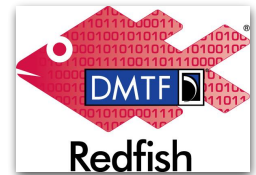
- 400MHz/800MHz ARM CPU
- 512 MB DDR3/DDR2 SDRAM
- PCIe VGA
- 10/100/1000 Mbps NIC (Dedicated/Shared)
- Web, IPMI, SSH (SMASH)



Standards

What are those?

- IPMI - common across all vendors, although shitty and insecure
- SSH - there's no standards (vendor specific implementations)
- Web interfaces - slow and buggy
- API - none or inconsistent implementations of Redfish[0]
 - Redfish is an odata based API, which is overkill[1] for a BMC device
 - Unreliable/buggy implementations across vendors
 - Every vendor seems to be heading in their own direction
 - Promising, maybe in the future this will be better



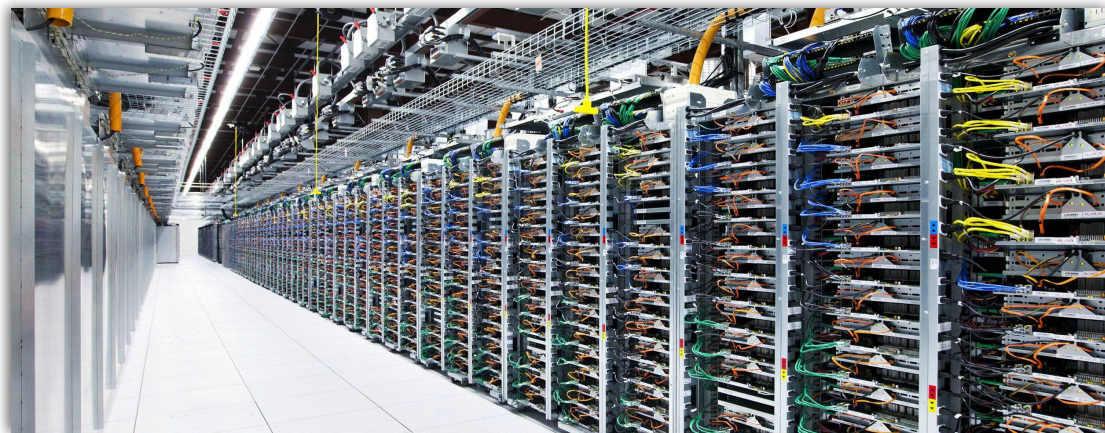
0. <https://www.dmtf.org/standards/redfish>

1. <http://docs.servicestack.net/why-not-odata>



The hardware

- 50K and growing set of bare metal servers and storage hardware.
- Hardware from multiple vendors, generations.



@scale

The challenges

- Four engineers, 50K servers ~ 12500 servers/engineer.
- Treat servers as light bulbs - **plug and play**.
- Reliably **provision**.
- Accurately **inventorize**.
- Manage BMC **configuration**.
- **Diagnose** hardware problems.
- All this done using the BMC, **without** manual intervention, at **scale**.





One library to abstract BMC interaction

Data collection support

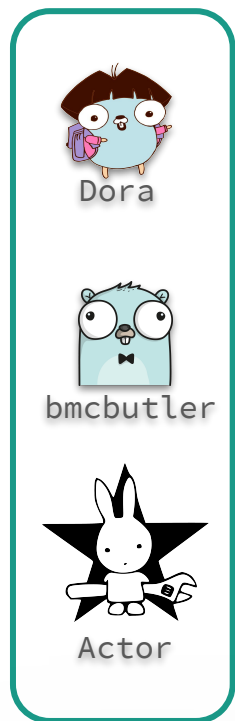
Hardware	Supported	Partially Supported
Dell M1000e	✓	
Dell iDRAC8	✓	
Dell iDRAC9	✓	
HP c7000	✓	
HP iLO3		✓
HP iLO4	✓	
HP iLO5	✓	
Supermicro X10	✓	

Configuration support

Hardware	User accounts	Syslog	NTP	Ldap	Ldap groups	BIOS	SSL
Dell M1000e	✓	✓	✓	✓	✓	-	
Dell iDRAC8	✓	✓	✓	✓	✓		
Dell iDRAC9	✓	✓	✓	✓	✓	✓	
HP c7000	✓	✓	✓	✓	✓	-	
HP iLO4	✓	✓	✓	✓	✓		
HP iLO5	✓	✓	✓	✓	✓		
Supermicro X10	✓	✓	✓	✓	✓		

<https://github.com/bmc-toolbox/bmclib>

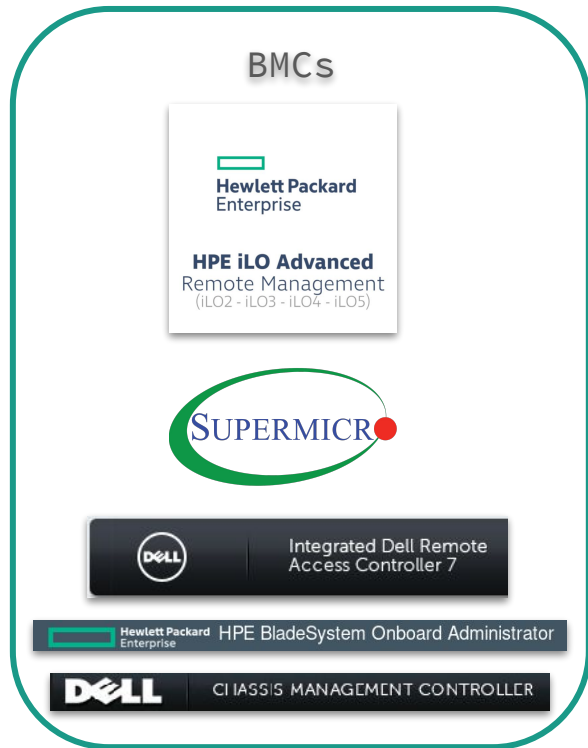
bmc-toolbox



bmclib



BMCs



bmcbutler

A BMC configuration management tool

```
syslog:  
  server: syslog.example.com  
  port: 514  
  enable: true #defaults to false  
  
user:  
  - name: Administrator  
    password: changeme  
    role: admin  
    enable: true  
  - name: foobar  
    password: changeme  
    role: user
```

BMC config



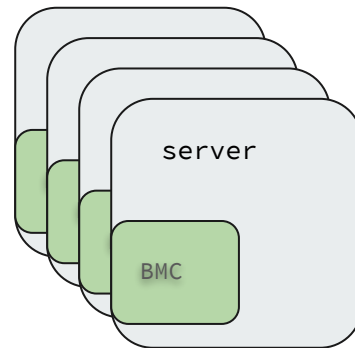
bmcbutler



bmclib

apply configuration

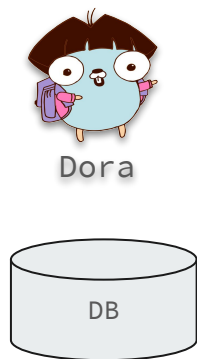
execute actions



go get github.com/bmc-toolbox/bmcbutler

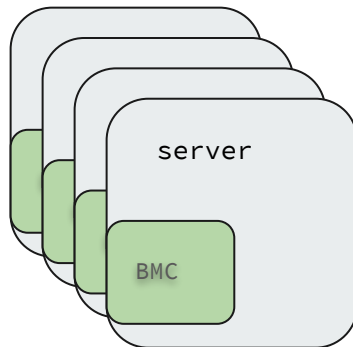
Dora

Asset inventory and explorer



bmclib

collect inventory

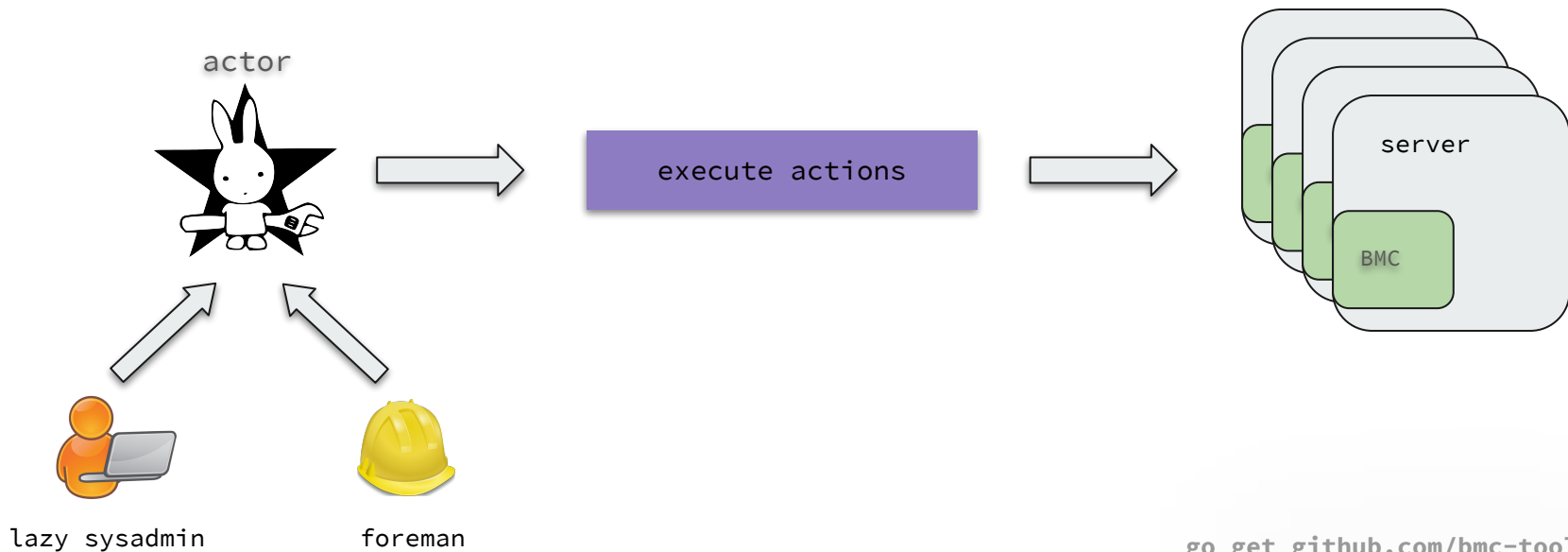


go get github.com/bmc-toolbox/dora

```
{
  "data": {
    "type": "blades",
    "id": "f1",
    "attributes": {
      "serial": "f1",
      "name": "fdi",
      "bios_version": "I36 10/17/2018",
      "bmc_type": "ilo4",
      "bmc_address": "f1",
      "bmc_version": "2.61 Jul 27 2018",
      "bmc_ssh_reachable": true,
      "bmc_web_reachable": true,
      "bmc_ipmi_reachable": true,
      "bmc_licence_type": "iLO Advanced",
      "bmc_licence_status": "Perpetual",
      "bmc_auth": true,
      "blade_position": 8,
      "model": "ProLiant BL460c Gen9",
      "temp_c": 23,
      "power_kw": 0.207,
      "power_state": "on",
      "status": "OK",
      "vendor": "HP",
      "processor": "intel(r) xeon(r) cpu e5-2620 v3",
      "processor_count": 2,
      "processor_core_count": 6,
      "processor_thread_count": 12,
      "memory_in_gb": 128,
      "updated_at": "2019-01-28T18:05:04.957145+01:00"
    },
    "relationships": { ... } // 4 items
  },
  "meta": {
    "author": "PSM Crew",
    "license": "APACHE2",
    "license-url": "https://www.apache.org/licenses/LICENSE-2.0",
    "service": "Dora"
  }
}
```

Actor

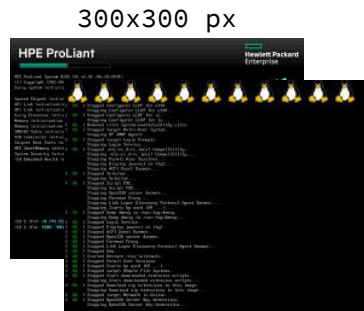
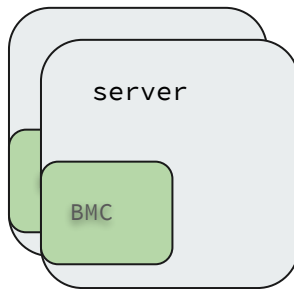
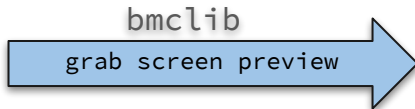
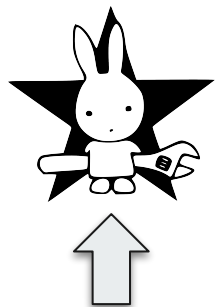
Consistent Web API interface to BMCs



go get github.com/bmc-toolbox/actor

bmclib in action

bare metal state identification

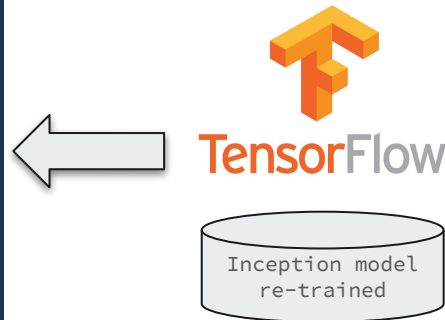


Hey Actor,
Get me the the state of
server-foobar?



Lazy sysadmin

```
"filename": "10.213.36.175-idrac8-1547652945.png",  
"labels": [  
  {  
    "label": "os install",  
    "probability": 0.7486801  
  },  
  {  
    "label": "pxeboot",  
    "probability": 0.16349028  
  },  
  {  
    "label": "os",  
    "probability": 0.05537151  
  },  
  {  
    "label": "os boot",  
    "probability": 0.018453646  
  },  
  {  
    "label": "post",  
    "probability": 0.0064438563  
  }  
]
```



Takeaways

- `bmclib` **abstracts** various vendor BMCs into a single API - so we can focus on building tools to manage them
- If you are looking to **inventorize, configure, update** server BMCs, check out the *bmc-toolbox*
- Asking what you need and get what you expect for orchestration is a must
- BMCs are a fundamental part of a servers lifecycle, it's time they got more attention
- You can help! create an issue/PR - If you work with bare-metal servers and would like support for your BMC
- Avoid vendor lock-in and have defined requirements to require