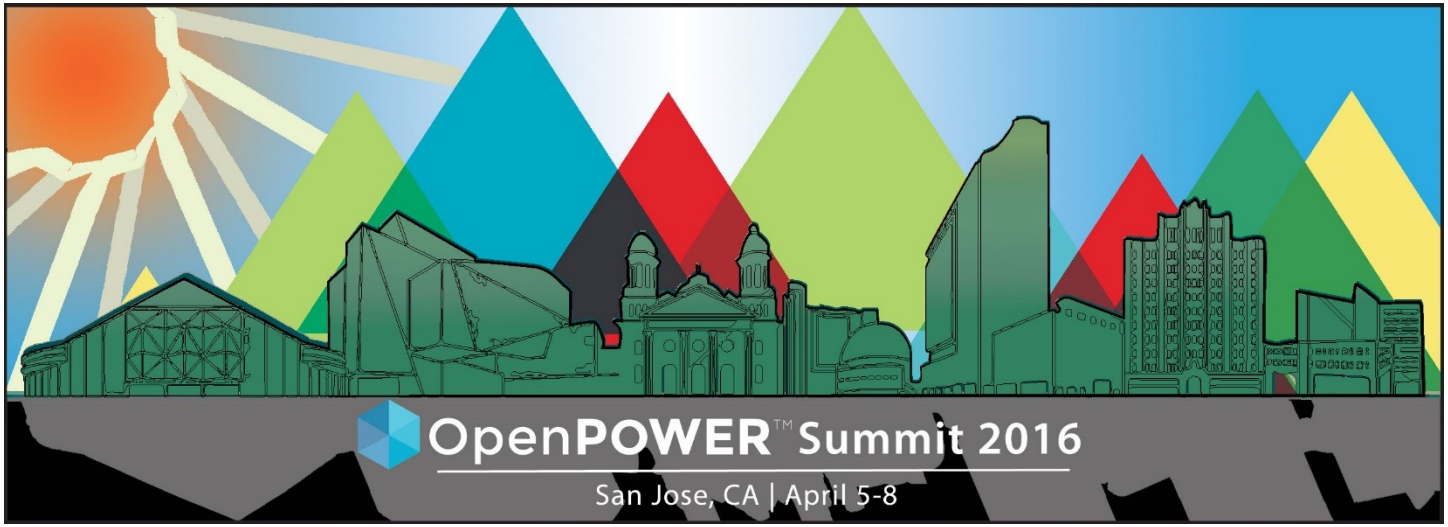
OpenPOWER™

Programming On-Chip Components To Retrieve Sensor Data

Shilpasri G Bhat <shilpa.bhat@linux.vnet.ibm.com>
Linux Kernel Developer, IBM Linux Technology Center
IBM India Systems and Technology Labs

Revolutionizing the Datacenter



Join the Conversation #OpenPOWERSummit

Agenda

- Introduction
- Overview of existing interfaces to read sensors
- OCC In-band Sensors
- Latency comparison involved to read sensor
- Work flow of OCC In-band Sensors
- Use cases and future enhancements
- Exploiting the sensors via Linux interfaces
- Build steps for OCC In-band Sensors

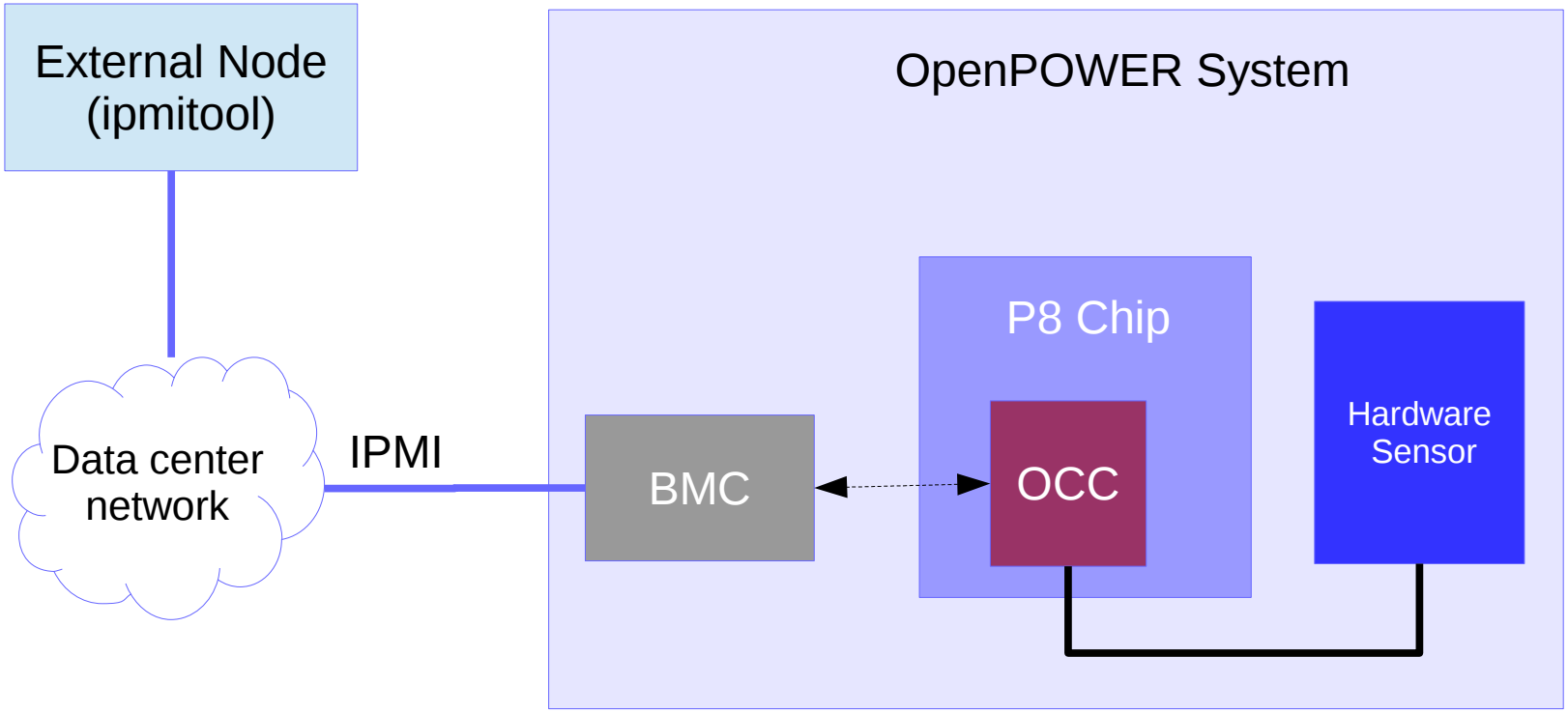
Introduction

- OpenPOWER provides a platform to program on-chip components present in POWER8 chip
- POWER8 has below on-chip programmable components
  - OCC *(On-Chip-Controller)* : Power/Thermal management
  - GPE *(General Purpose Engine)* : Used by OCC
  - SLW *(Sleep-Winkle Engine)* : Used to restore core logic after power management idle instruction
  - SBE *(Self Boot Engine)* : Used for chip initialization and to load and start Hostboot firmware

Existing interfaces for reading sensors (1)

- Out-of-band via IPMI
  - ipmitool raw <netfn> <cmd> [data] -H <bmc_ip> -U <user> -P <password>
  - ipmitool sdr -H <bmc_ip> -U <user> -P <password> -I <interface>
  - ipmitool sensor -H <bmc_ip> -U user -P <password> -I <interface>

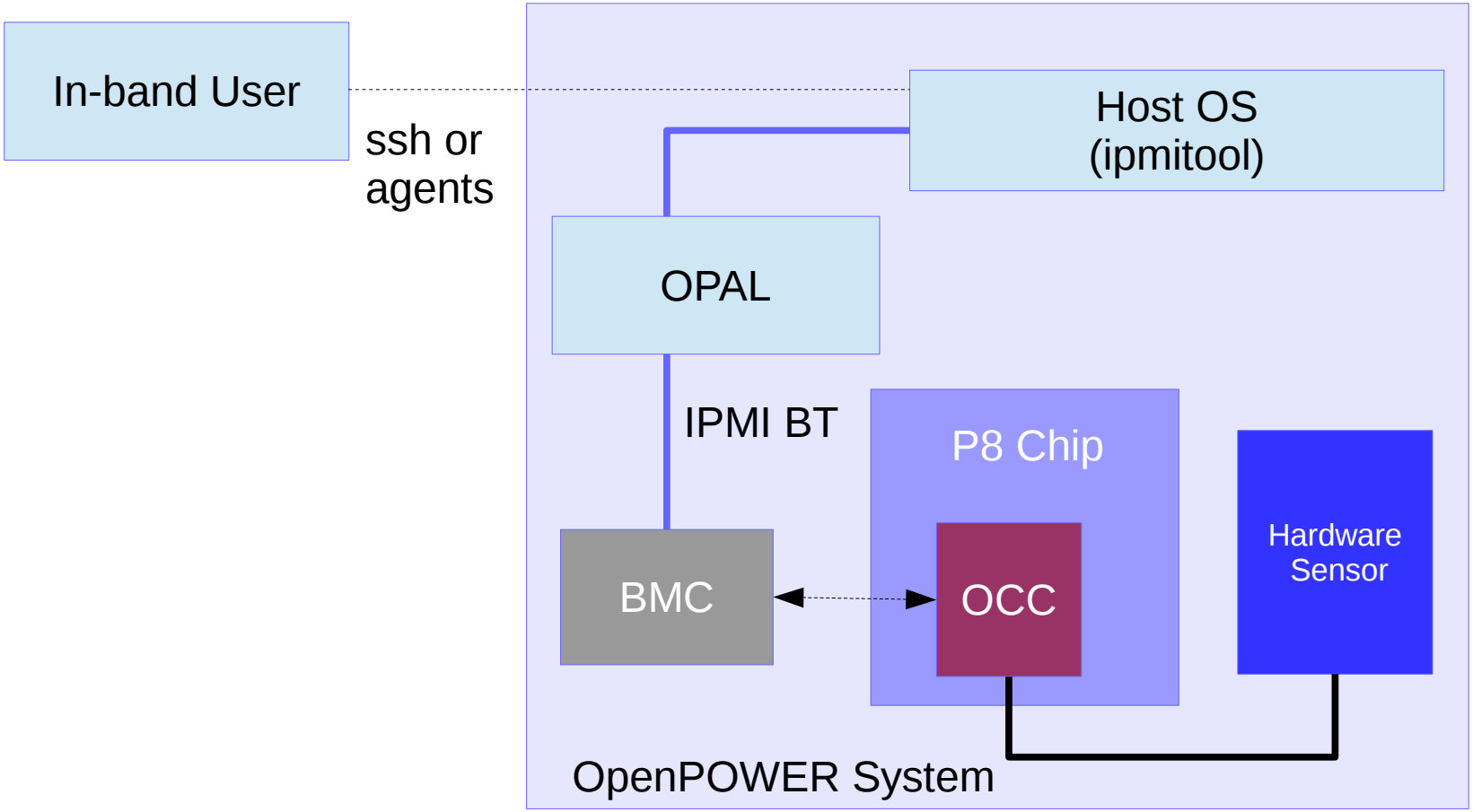
IPMI Out-Of-Band



Existing interfaces for reading sensors (2)

- In-band via IPMI
  - ipmitool raw <netfn> <cmd> [data]
  - ipmitool sdr
  - ipmitool sensor

IPMI In-band


In-band User
ssh or
agents
Host OS
(ipmitool)
OPAL
IPMI BT
P8 Chip
BMC
OCC
Hardware
Sensor
OpenPOWER System

Existing interfaces for reading sensors (3)

- In-band via XSCOM
  - XSCOM is a specialized scan communication interface in the chip pervasive logic to access specific latches from the processor cores.
  - Core and memory temperatures are read from Digital Thermal Sensors via XSCOM.
  - Exported as HWMON sensors

OCC In-band Sensors

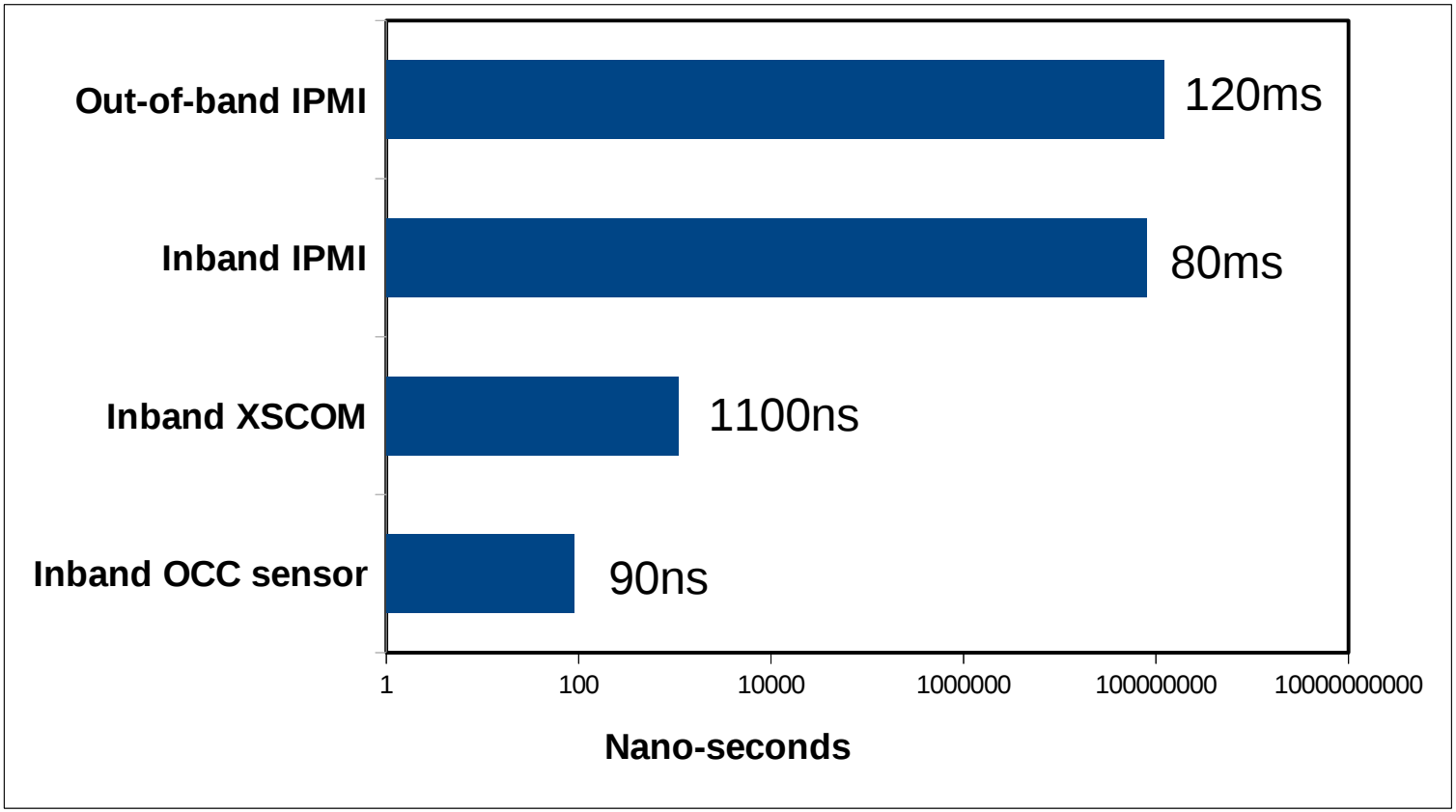
- Involves ***programming on-chip component OCC*** to enable instrumentation in host Linux kernel
- OCC In-band Sensors provide a mechanism to expose the ***platform sensor data*** in-band through standard Linux interfaces
  - Perf
  - HWMON/lm-sensors
- Enable workload profiling and faster instrumentation using fast-sensing of platform sensor data

Why In-band Sensors?

- To reduce the latency involved in reading the platform sensor data
  - Low overhead to monitor.
- To export the platform sensor data in standard Linux interfaces to enable:
  - Profiling of applications
  - Exporting sensor data to in-band monitoring agents (OpenStack)

Latency Comparison

- Time taken to read sensor

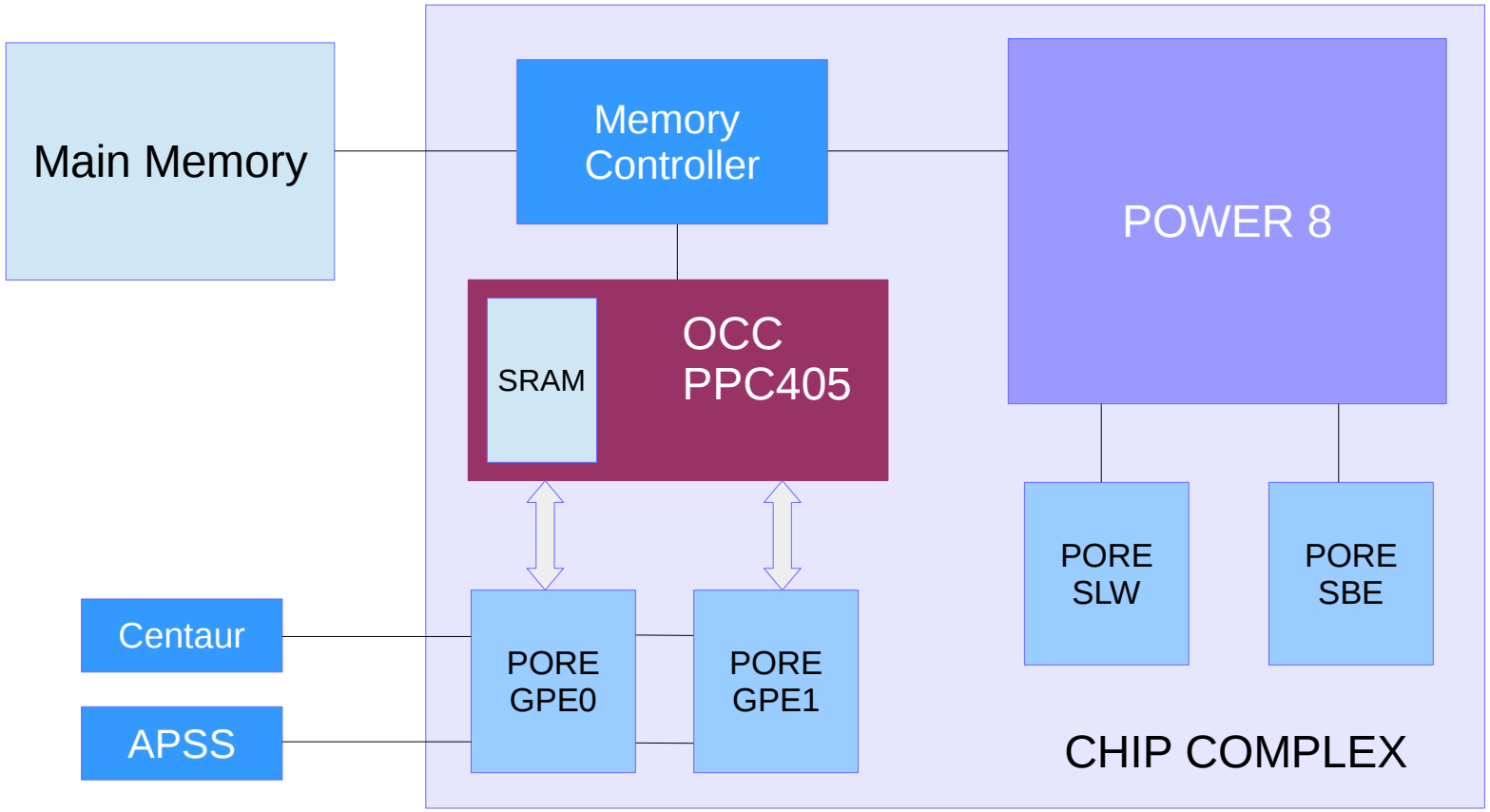


* IPMI includes network latency

What is OCC?

- On-Chip Controller (OCC)
  - Embedded PPC405 with 512K SRAM
  - Takes care of power management per chip
  - Has two co-processors to offload work
    - PORE-GPE0 (Power On Reset Engine - General Purpose Engine 0)
    - PORE-GPE1 (Power On Reset Engine – General Purpose Engine 1)
  - ***Periodically reads and stores many platform sensor data*** (power, thermal, memory-bandwidth, cpu utilization, cpu frequency etc)
    - Total System Power is read every 250us
    - Core temperature is read every 2ms

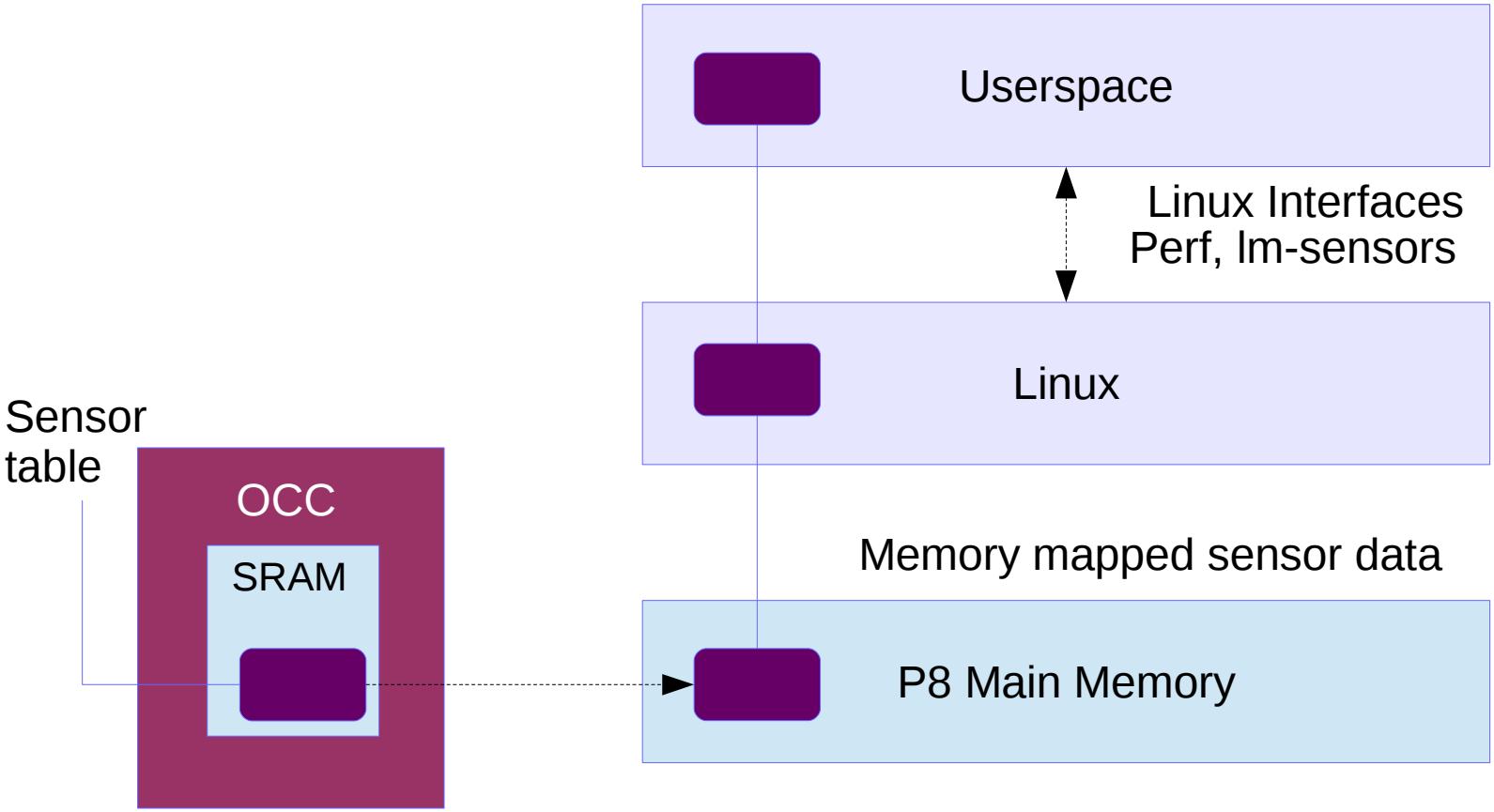
On-Chip-Controller (OCC)



Work Flow

- Program OCC to copy relevant sensors from OCC SRAM --> Main memory in a regular interval.
  - Create a sensor table to pick the relevant sensor data
  - Periodically queue work on 'Block Copy Update Engine (BCUE)' which is a SRAM to mainstore copy thread to copy the sensor table
  - Copy the sensor table to OCC-OPAL-shared-memory-region
    - The shared memory region is pre-defined and pre-allocated by Hostboot

How does inband-sensor work?



Consuming sensor data in host

- OPAL memory maps the sensor table from the shared memory region
  - Abstracts the sensor information as device tree entry (name, type, unit, size, address)
- Sensors copied to main memory can be consumed in the host in the below format
  - Perf
  - Lm-sensors
  - Sysfs files

Use Cases and Future Enhancements

- Provisioning using Memory Bandwidth metric in OpenStack

  https://www-304.ibm.com/partnerworld/wps/servlet/download/DownloadServlet?id=xlxSBGVx2TFiPCA$cnt&attachmentName=cloud_optimization_using_ibm_power_platform.pdf&token=MTQ1NzMzOTI4MTY5Mg==&locale=en_ALL_ZZ

- Highly suitable for profiling HPC applications with fast-path for platform sensor data
- Correlate with workload behavior and to auto-tune the workload
- In-band AMESTER

Backup Slides

Linux Perf Interface (event list)

perf list

```
occ_power/chip_energy/                    [Kernel PMU event]
occ_power/system_energy/                  [Kernel PMU event]
occ_power/system_power/                   [Kernel PMU event]
```

Linux Perf Interface

```
# perf stat -a -e occ_power/system_power/
./ebizzy -s 4096 -S 10 -t 64  > out

  Performance counter stats for 'system wide':

                559 Watts occ_power/system_power/


      10.003012034 seconds time elapsed
```

Linux Lm-sensor Interface

```
# cat /sys/class/hwmon/hwmon0/power1_input
417000000      --> Unit in Micro Watts
# sensors | grep power
  power1:        424.00 W
```

Linux Sysfs Interface

```
# cd /sys/devices/system/cpu/occ_sensors/
# ls

chip0  system

# ls *

chip0:

chip-energy  chip-mbw  core1-temp  core2-temp  core3-temp
core4-temp  core5-temp  core6-temp  core7-temp  core8-temp
power power-memory  power-vcs  power-vdd

system:

ambient-temperature  count  fan-power  fan-speed  gpu-
power  io-power  power  storage-power  system-energy
```

How to enable In-band Sensors? (1)

- In-band OCC sensor requires the below firmware changes:
  - Skiboot changes to export this memory mapped sensor data to kernel as device tree entries

    `https://github.com/shilpasri/skiboot/tree/inband-sensors`
  - OCC changes to write the sensor data periodically to main memory

    `https://github.com/shilpasri/occ/tree/inband-sensors`

How to enable In-band Sensors? (2)

- Host Linux requirements to enable In-band sensors:
  - Platform HWMON driver 'ibmpowernv.ko' (which can be enabled by setting the kernel config option CONFIG_SENSORS_IBMPOWERNV)
  - Custom kernel module to export the sensors as sysfs files

    `https://github.com/shilpasri/inband_sensors.git`

How to build the custom PNOR image?

```
# git clone --recursive https://github.com/shilpasri/op-build.git
# cd op-build
# git checkout -b inband_sensors origin/inband_sensors
# . op-build-env
# op-build palmetto_defconfig && op-build --> Palmetto
# op-build habanero_defconfig && op-build --> Habanero
# op-build firestone_defconfig && op-build --> Firestone
```

Acknowledgments

- Vaidyanathan Srinivasan
- Madhavan Srinivasan
- Dipankar Sarma
- Todd Rosedahl
- Nilesh Joshi
- Shreyas Prabhu
- Gautham Shenoy
- Akshay Adiga
- Stewart Smith

THANK YOU

Legal Statements

- Copyright International Business Machines Corporation 2016.
- This work represents the view of the authors and does not necessarily represent the view of IBM.
- IBM, IBM logo, ibm.com are trademarks of International Business Machines Corporation in the United States, other countries, or both.
- Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.
- Other company, product, and service names may be trademarks or service marks of others.
- References in this publication to IBM products or services do not imply that IBM intends to make them available in all countries in which IBM operates.
- INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you. This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.