Open Source Summit Japan
Jul 19, 2019 16:00~16:40

FUJITSU
shaping tomorrow with you

Evolving NVDIMM for Enterprise Grade

QI Fuli

Linux Development Division

Fujitsu Limited

Who am I

- QI Fuli
  - Software Engineer at Fujitsu Ltd
  - PhD Student at University of Tokyo
  - Working on Persistent Memory
  - Email: qi.fuli@fujitsu.com

Contents

FUJITSU

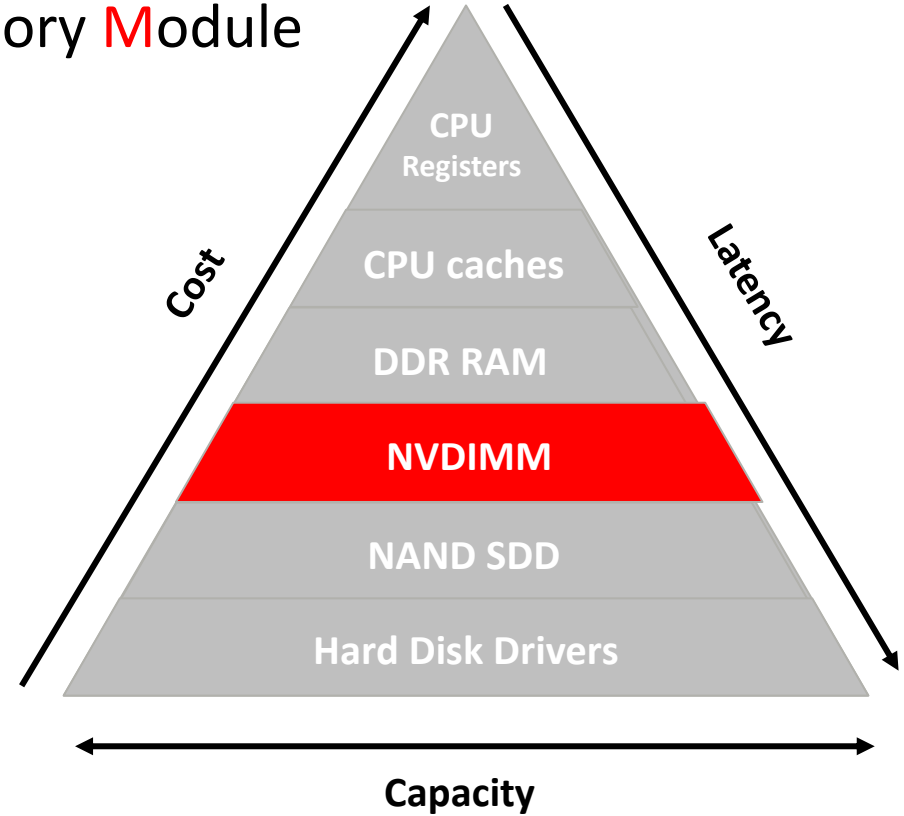
- Introduction of NVDIMM
- Monitor the SMART events from NVDIMM
- Distinguish and replace faulty NVDIMM
- Future work & Summary

FUJITSU

Introduction of NVDIMM

NVDIMM Overview

- Non-Volatile Dual In-line Memory Module
  - a type of random-access memory
  - NVDIMM retains its data even if electrical power is removed
- Use case
  - In-Memory Database, etc.



Concepts [1]

FUJITSU

- Interleave set
  - Two or more NVDIMMs create an N-Way interleave set to provide stripes read/write operations for increased throughput
- Namespace
  - Defines a contiguously-addressed range of Non-Volatile Memory
- Region
  - A group of one or more NVDIMMs, or an interleaved set, that can be divided up into one or more Namespaces

[1] https://docs.pmem.io/ndctl-users-guide/concepts

Concepts [1]

FUJITSU

Type

- Defines the way in which the persistent memory associated with a Namespace or Region can be accessed
- PMEM: Direct access to the media via load/store operations. (DAX supported)
- BLK: Direct access to the media via Apertures. (DAX is not supported)

Mode

- Defines which NVDIMM software feature are enabled for a given Namespace.
- Namespace Modes include fsdax, devdax, sector, and raw.

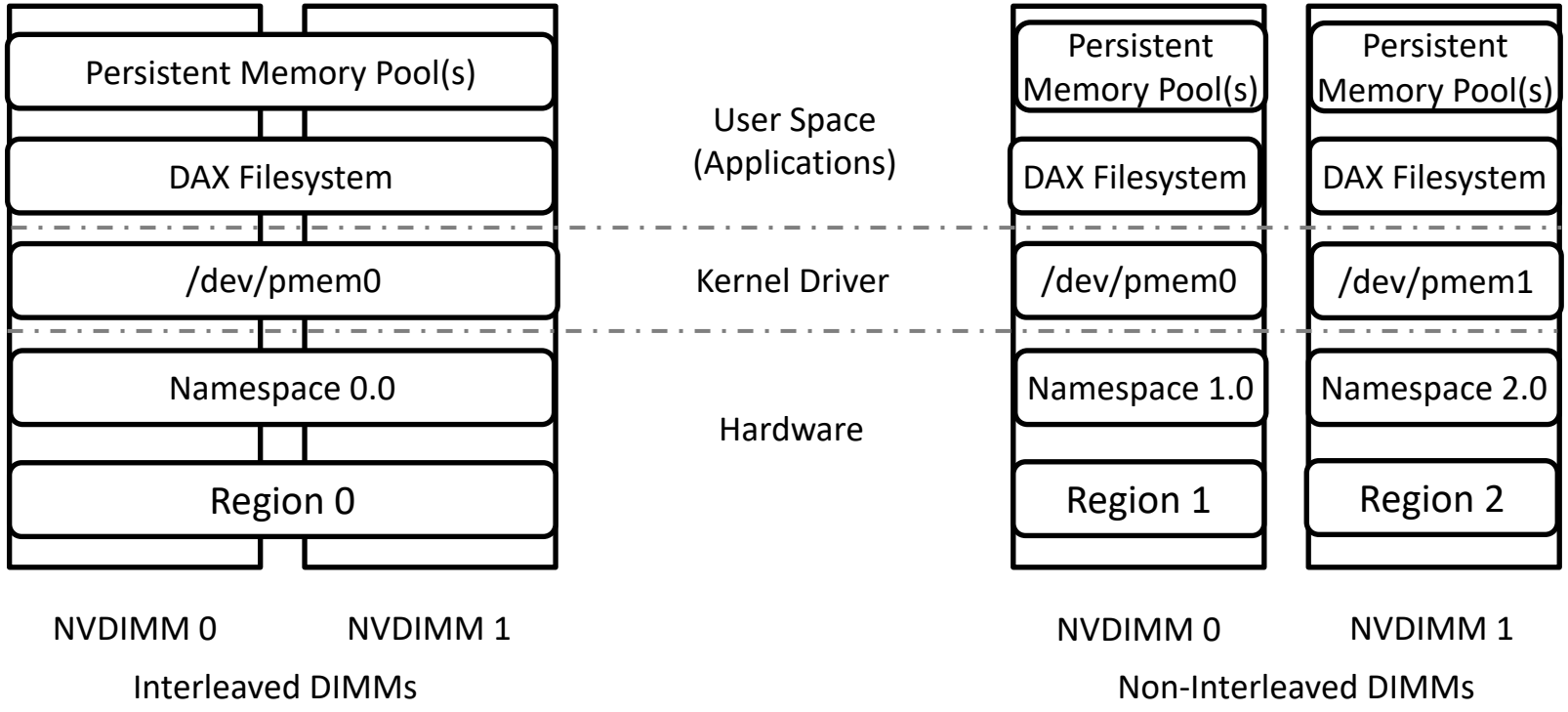
[1] https://docs.pmem.io/ndctl-users-guide/concepts

Namespace Modes

FUJITSU

- fsdax
  - filesystem provide direct access to Persistent Memory to applications
- devdax
  - creates a character device instead of a block device
  - intended for applications that mmap() the entire capacity
- sector
  - to help software that doesn't understand that sectors might end up with a mix of old and new data if power loss occurs while writes were underway
- raw
  - a memory disk that does not support DAX

Configuration options



Non-Volatile Device Control (NDCTL)

FUJITSU

- A utility for managing the Linux LIBNVDIMM Kernel subsystem
- Working with various NVDIMMs from different vendors
- Operations supported by ndctl
  - Provisioning capacity
  - Enumerating Devices
  - Enabling and Disabling NVDIMMs, Regions, and Namespaces
  - Managing NVDIMM Labels

FUJITSU

Monitor the SMART events from NVDIMM

Why NVDIMM must be monitored?

FUJITSU

- NVDIMM has a life span
  - Blocks of NVDIMM will be broken due to endurance problem
  - NVDIMM consumes spare block to rescue broken block
  - When spare decreases to 0, it can not rescue broken block
  - NVDIMM has no feature such as mirroring to save its data
- Backup and replacement is needed before no spares
- Users should know the best time to backup and replace

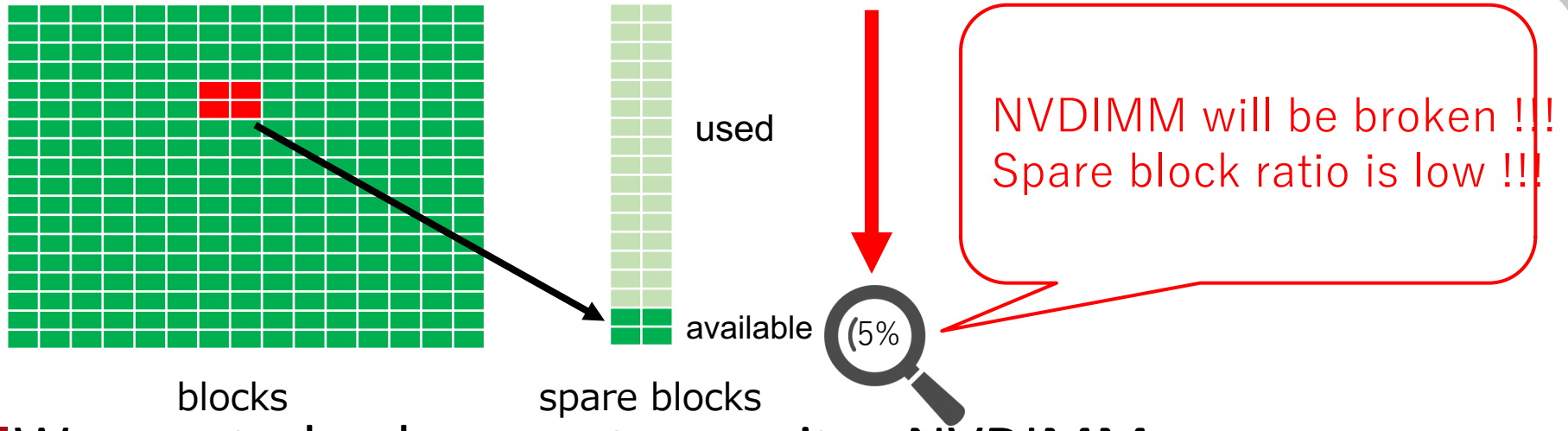
Why NVDIMM must be monitored?

FUJITSU

- NVDIMM has a life span
  - Blocks of NVDIMM will be broken due to endurance problem
- Backup and replacement is needed before no spares
- Users should know the best time to backup and replace

Status of NVDIMM needs to be monitored and be notified before NVDIMM becomes faulty

Ndctl monitor



- We created a daemon to monitor NVDIMM
  - monitor health status
  - notify critical status

https://pmem.io/ndctl/ndctl-monitor.html

Features of ndctl monitor (1/3)

FUJITSU

Monitoring

SMART event	trigger	effect
spares-remaining	spare block value goes below the spare threshold limit	NVDIMM becomes broken; data will be lost; etc.
health-status	normal health status of NVDIMM changes	
unclean-shutdown	last shutdown to be recorded at the next boot	saving data target fail on NVDIMM; etc.
media-temperature	temperature value of NVDIMM goes above threshold limit	server is getting too hot; may need remediation; a specific fan fails; etc.
controller-temperature	controller temperature value goes above threshold limit	

Features of ndctl monitor (2/3)

FUJITSU

- Actions after monitoring
  - Log the output notification
    - Log destination
      - syslog
      - arbitrary file (set in the configuration file or set by command option)
    - Output format
      - JSON (can be analyzed by other log collectors, such as fluentd)
  - Kick other application (to be implemented)
    - When the monitor detects a SMART health event, the application will move the data in real time.

Features of ndctl monitor (3/3)

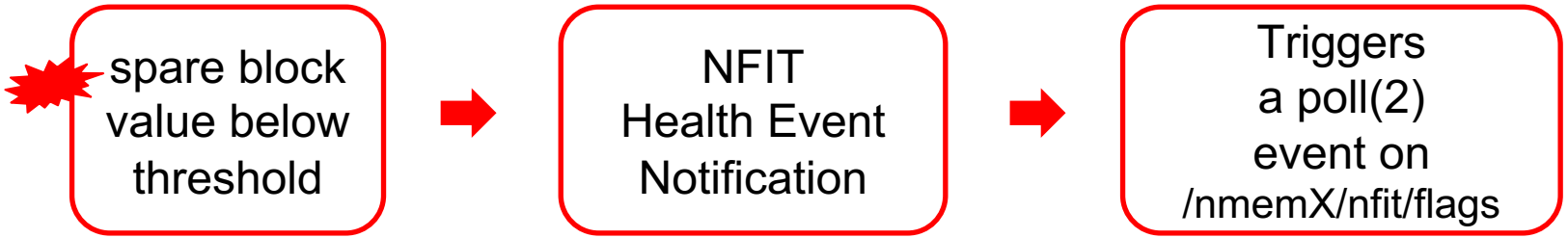
FUJITSU

- Filtering
  - objects to monitor can be filtered
    - all NVDIMMs are monitored by default
    - objects can be filtered by namespace, region, bus, and dimm
      - backup the data of applications which are running on a certain namespace
        ⇨ filter by namespace
      - allowing to monitor for any media error on the bus
        ⇨ filter by bus
  - SMART health events to monitor can be filtered

ACPI supports for NVDIMM

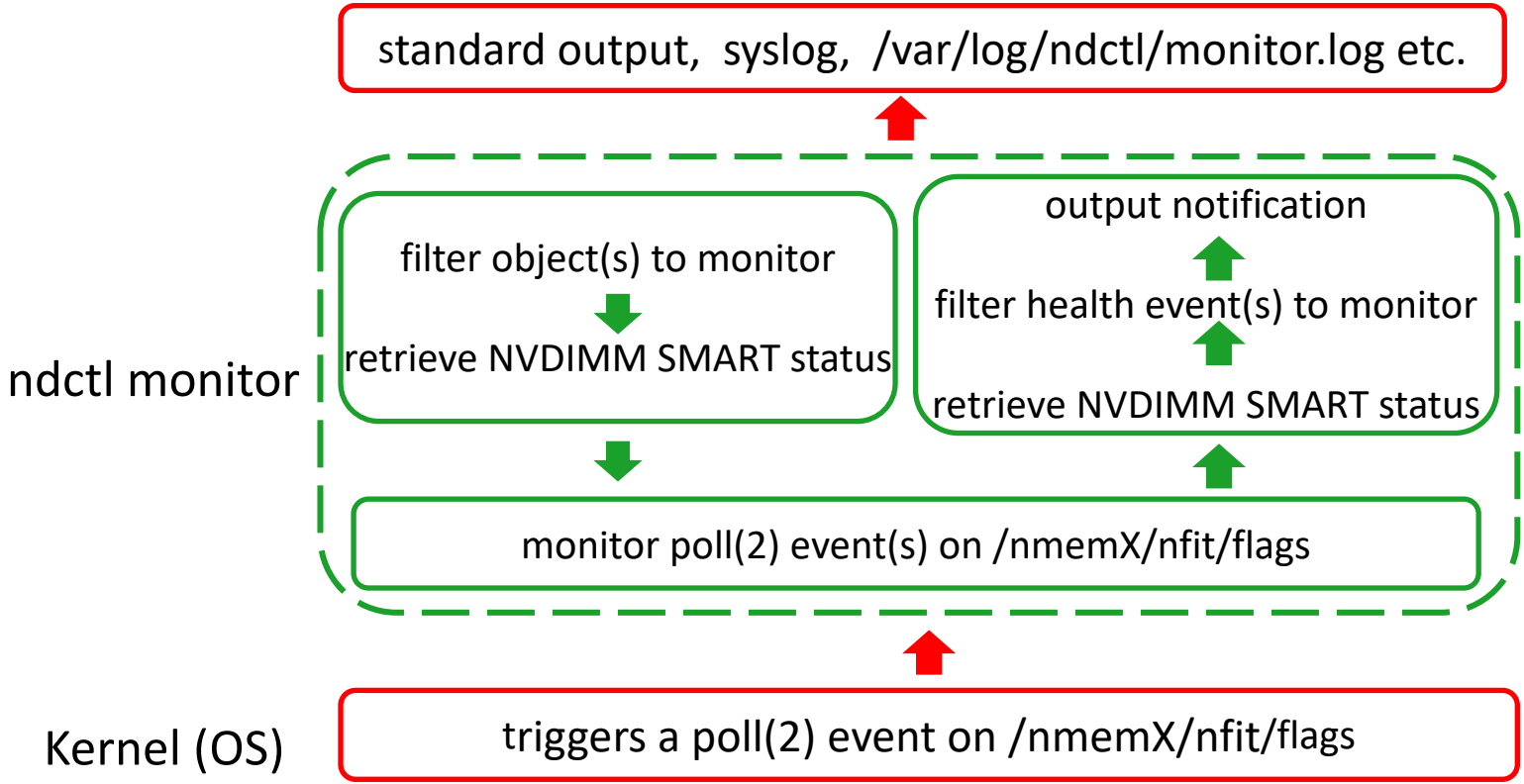
FUJITSU

- NVDIMM has SMART Health Info like SSD/HDD
  - SMART Health Info can be retrieved via a *_DSM*
  - SMART Health Info includes spare block value and spare threshold limit
- ACPI 6.1 adds “NFIT Health Event Notification”
  - a notification will be sent when the spare block value goes below the spare threshold limit
  - a poll(2) event will be triggered when the notification is received



Architecture of ndctl monitor

FUJITSU



Sample of output notification

FUJITSU


```
"timestamp":"1537323709.432459532",
  "pid":28189,
  "event":{ "dimm-spares-remaining":true },
  "dimm":{
    "dev":"nmem1", "health":{
      "health_state":"non-critical",
      "temperature_celsius":23,
      "controller_temperature_celsius":25,
      "spares_percentage":4,
                          …
      "spares_threshold":5,
      "shutdown_state":"clean“
   }}
```


SMART Health event

current spare block value

spare threshold limit

How to start ndctl monitor

FUJITSU

- Linux distribution support systemd

```
# systemctl start ndctl-monitor.service
```

- Linux distribution does not support systemd

```
# ndctl monitor --daemon
```

- Users could run monitor as a one shot command

```
# ndctl monitor
```

Distinguish and replace faulty NVDIMM(s)

Replacing faulty NVDIMM is complex

FUJITSU

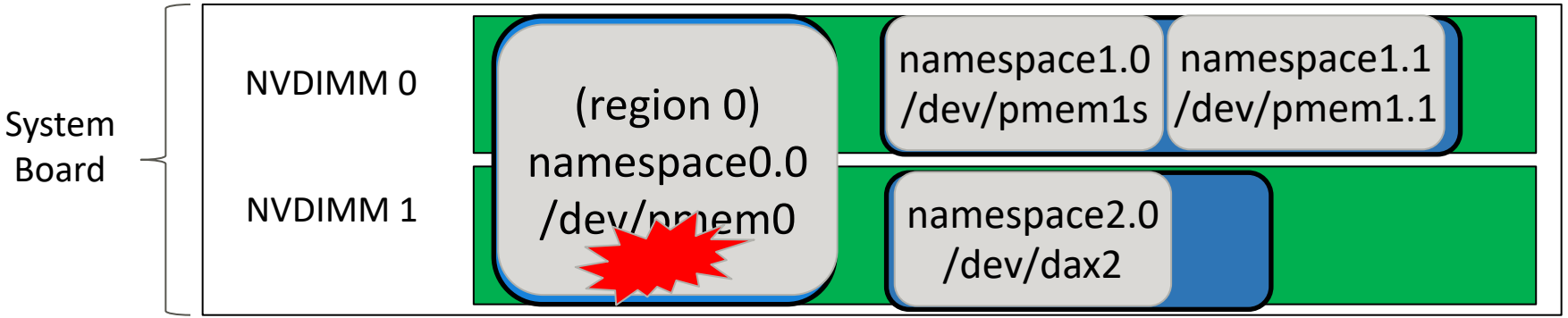
- Replacing NVDIMM is more difficult than SSD/HDD
  - Hardware does NOT support hot-replace for NVDIMM *
  - No hardware mirroring for NVDIMM
  - Software RAID does NOT work with Filesystem DAX or Device DAX
  - The specification of NVDIMM (region and namespace) causes additional complexity

* by then, distinguish feature was under developing

Replacing faulty NVDIMM is complex

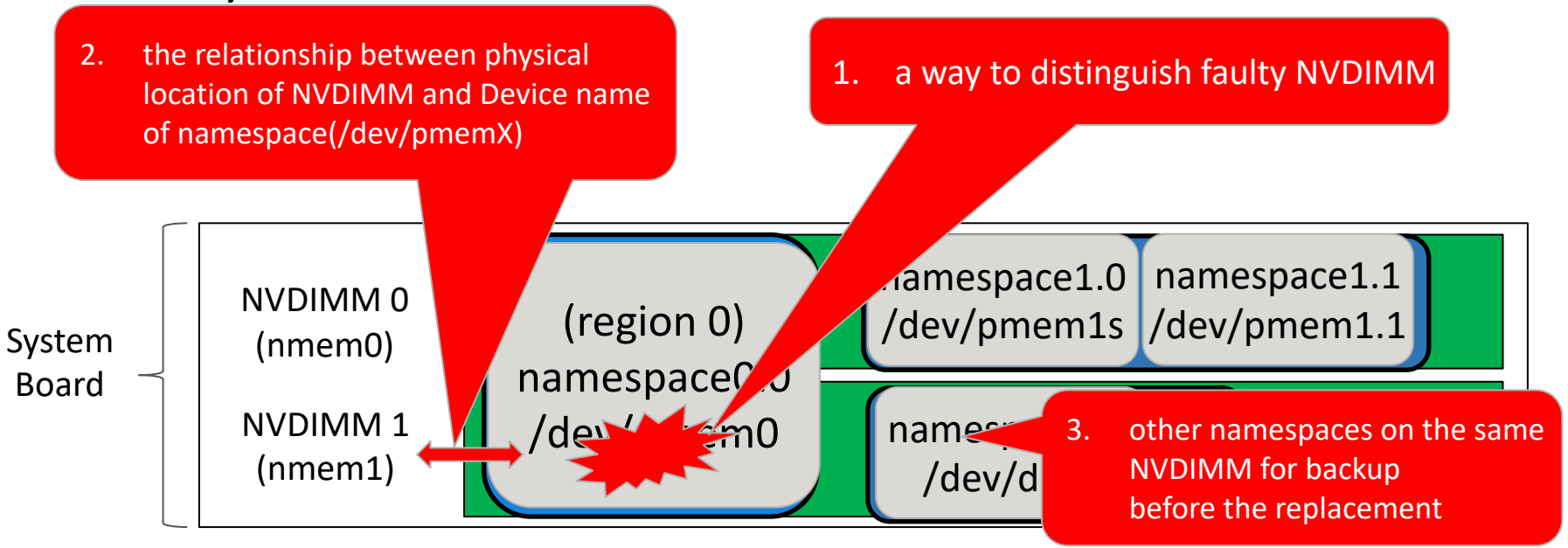
FUJITSU

- If a block on Region 0 is faulty, and users need to replace the faulty NVDIMM, what information is necessary?



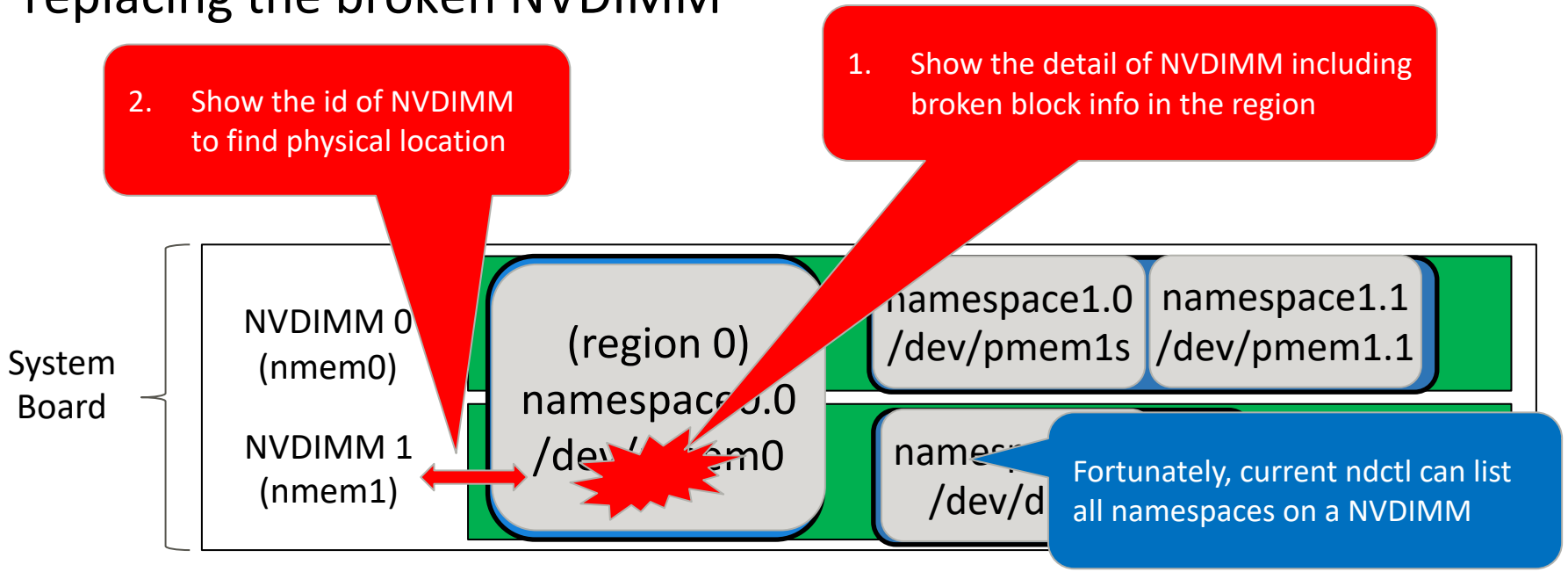
Distinguish the faulty NVDIMM

- To replace the faulty NVDIMM, at least the following information is necessary.



What Fujitsu has developed

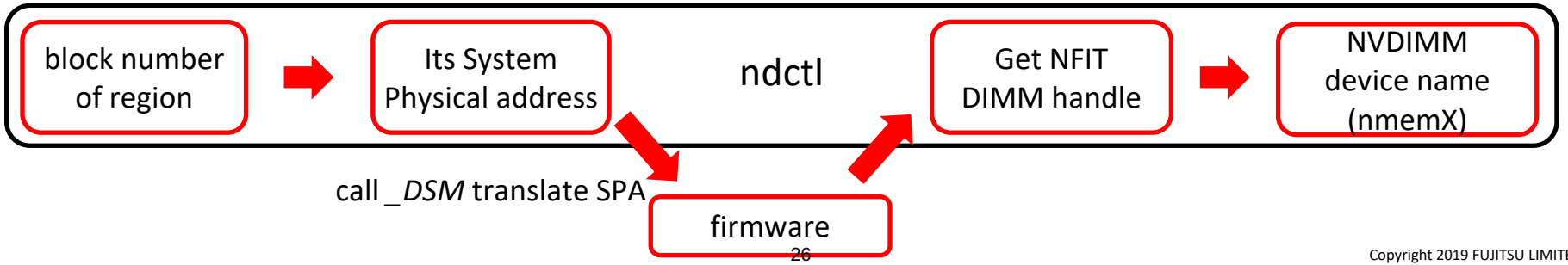
- Two features of ndctl to show the important information for replacing the broken NVDIMM



To show faulty NVDIMM

- Ndctl did not display faulty NVDIMM information
  - It only listed bad block number in the region
  - When the region is interleaved, kernel/driver cannot distinguish NVDIMM.
  - "Translate SPA" is defined in _*DSM* of "NVDIMM root Device" at ACPI 6.2

    Firmware translates System Physical Address to "NFIT DIMM handle" and Physical address in the DIMM
- the translation in ndctl utility



To display bad block information

```
# ndctl list -DRHMu
   "regions":{
       "dev":"region0",
       "mappings":[
         { "dimm":"nmem1",},
         { "dimm":"nmem0",}
        ]},
       "badblock_count":1,
       "badblocks":[
         { "offset":65536,
           "length":1,
           "dimms":[ "nmem0“ ]}
       ]
```

command to show region info with dimm, health, and bad block info

region 0 is interleaved by nmem1 and nmem0

displays which NVDIMM has bad block in this region

To show NVDIMM device location

FUJITSU

- The physical location of a device can be confirmed with “dmidecode”
  - In addition, each device has SMBIOS handle, and shown in the command
- SMBIOS handle is “NVDIMM Region mapping structure” table in NFIT
- If ndctl utility can show it, the location can be found with the handle
- We made a patch that let ndctl show it as phys_id

Example of SMBIOS Handle of ndctl

FUJITSU

```
# ndctl list -Du

 {    "dev":"nmem1",
       "id":"XXXX-XX-XXXX-XXXXXXXX",
       "handle":"0x120",
       "phys_id":"0x1c“ },
 {    "dev":"nmem0",
       "id":"XXXX-XX-XXXX-XXXXXXXX",
       "handle":"0x20",
       "phys_id":"0x10",
       "flag_failed_flush":true,
       "flag_smart_event":true }
```

display DIMM information with human readable format

The nmem0 includes broken block in previous result

phys_id is SMBIOS handle of these DIMM

0x10 is faulty NVDIMM’s handle

To show the location of NVDIMM

FUJITSU

```
# dmidecode

Handle 0x0010, DMI type 17, 40 bytes
 Memory Device
        Array Handle: 0x0004
            :
            :
         Locator: DIMM-Location-example-Slot-A
            :
        Type Detail: Non-Volatile Registered (Buffered)
```

SMBIOS handle of NVDIMM

Locator: shows the place of NVDIMM

To find all namespaces on NVDIMM

FUJITSU

```
# ndctl list -N -d nmem0
 [
   { "dev":"namespace0.2",
     "mode":"sector",
     "size":67042312192,
          …
   },
   { "dev":"namespace0.0",
     "mode":"sector",
     "size":67042312192,
          …
   }
 ]
```

command to find
all namespaces on the nmem0

FUJITSU

Future work & Summary

Removing the “Experimental” of FS-DAX

FUJITSU

- In the upstream, Filesystem DAX is still marked as experimental.
- Removing the experimental in the upstream was discussed during LSF/MM summit 2019
  - The end of the DAX experiment [1]
  - DAX semantics [2]

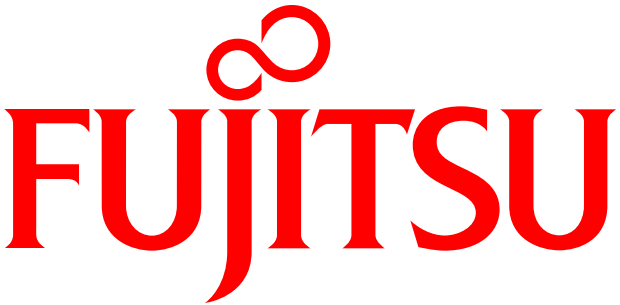
Fujitsu would like to remove the “experimental” of FS-DAX on XFS

[1] https://lwn.net/Articles/787233
[2] https://lwn.net/Articles/787973

Summary

FUJITSU

- Introduction of NVDIMM
- Monitor SMART events from NVDIMMs
  - Monitoring SMART status of NVDIMM is important
  - Ndctl monitor daemon
- Distinguish and replace faulty NVDIMM
  - Replacement of NVDIMM is complex
  - Adding some information in ndctl list for replacing NVDIMM



shaping tomorrow with you