History & Today

History:
* Concept introduction under Storage Track – Mar, 2017

Today:
* Provide overview on the project architecture
* Update the development status
* Share an use case in the data center environment
* Discuss future development
* Call for actions & participations

Disclaimer: This presentation is not a contribution to Open Compute Project
Project Overview

• A storage device vendor provided, user space library, conforming to OCP Health Chain Management API
• Targeting large storage device deployment environment
• Focusing on the storage device management, including health condition check, field configuration, decommission, and etc.
• Support all storage devices: HDD, SSD, etc. with different interfaces
High Level Matrix

- Vendor Factory Data
- System Integration Data
- Data Center Operation Data
- Device Decommission

Supply Chain Data Model

Supply Chain Data Management

Drive Health Management API

- Health Check
- Self Test
- Field Repair
- Device Decommission
- Other...

API Library

DRAFT
Do Not Distribute!
V1.0 API Functions

- `bool getFuncArray(uint64_t array[], uint32_t size)`  
  Ensures backward and forward compatibility

- `int64_t startHealthCheck(const std::string& devNode)`  
  Primary function/task to verify device health

- `int64_t startSelfTest(const std::string& devNode)`

- `int64_t startOfflineSelfTest(const std::string& devNode)`

- `int64_t startRecover(const std::string& devNode)`

- `int64_t startOfflineRecover(const std::string& devNode, uint32_t level = 0)`

- `int64_t startSecureWipe(const std::string& devNode)`

- `int64_t startOfflineSecureWipe(const std::string& devNode, uint32_t pattern)`

- `int64_t abortTask(const std::string& devNode, uint32_t timeout = 10)`

- `int64_t checkTaskStatus(const std::string& devNode, TaskStatus& status)`

- `std::string getErrorCodeMessage(int64_t errorCode)`

- `int64_t getWAF(const std::string & devNode, const std::string & file, double & result)`

- `int64_t getVendorLog(const std::string & devNode, const std::string & path, std::string & name, uint32_t level = 0)`

Seagate will upload the binary to the OCP Storage Github

---

OPEN. FOR BUSINESS.
Sample Implementation - Task Based

System Suspects HDD Failure

Reset or Power
Reset Device
&
Launch DHL

bool getFuncArray (uint64_t array[], uint32_t size);

int64_t startHealthCheck (const std::string & devNode);

SUCCESS?

No (Err Code)

Yes (0)

int64_t checkTaskStatus (const std::string & devNode, TaskStatus & status);

Sleep (5)

100%?

No

Yes

ErrCode==0?

Pass!
Drive Return
to Services

Yes

No

Fail!
Vendor diag
t through
ErrCode

Do Not Distribute!
Hyperscale Data Center Use Case for API

- Simplified and controlled deployment of vendor diagnostic tools in heterogeneous environments
- Vendor gets to diagnose device health based on SMART and vendor unique proprietary information
- Vendor receives failure information based on vendor algorithms
- Hope to create a robust feedback loop to improve vendor products, reduce failures - measured by AFR
- Removes making guesses based on SMART and applying it across multiple vendors and families of products
Future...

Ph 1: Passive
- Snap shot check
- Passive

Ph 2: Active
- Repair
- Security protocol enabling
- Machine Learning Integration

Ph 3: Real Time
- Real time monitor
- Prediction
Call For Actions

• Storage vendors? Join the API implementation?
• Operational professionals? Can this lib be useful?
• C/C++ developers? Interested in being part the dev effort?
• Storage professionals? Any other use case it shall cover?

We invite you to join the future feature discussion, and development!