Gen Z Queuing

Diving Into the Deep End of the Pool

Ralph Weber T10/T13 Representative – Ralph.Weber@wdc.com – 214-912-1373
Queuing Not Trusted

✔ Not used at all

✔ Used only with limited queue depths

“Sending too many commands in a queue simply allows the HDD to run amuck”

“We have service agreements to meet.”
“We must manage devices, and particularly HDD queues to make this happen.”

Photo credit: Palo Cech
On the Flip Side

Drive vendors see a long history of queues improving throughput

Gut instinct says …

+ HDDs know things
  ➔ Where the heads are
  ➔ When sector will be under the head

+ HDD can react faster than host can send instructions

Photo credit: Pixabay
Squaring the Circle

星级 How to match
+ HDD queue management
✓ Host expectations and …
    ultimately Customer needs

✗ Another long/checkered history
  ➡ New attempt every 2-5 years
  ➡ Standards defunct before ink dries
  ➡ No joy in Mudville … not yet
Towards More Useful Queues

- Offload Host queue goals to HDD
- Generalize ... support multiple Host types
- Consider HDD issues, but ... don’t be driven by them
- Build a synergy between
  - What Host needs
  - What HDD can do

Photo credit: Pixabay
New “Language”

→ Communicate Host expectations to HDD Queue Manager

📖 It's not a **language**, really!

- More constrained than C++ *(than JCL, come to that)*

- Control Values & Response Actions *(more than any previous attempt)*

- Lots of Reserved Space for added features in the future
Benefits

- Avoid per-customer point solutions
- Put good ideas in standard controls
  - Revising standards, if needs be

- ROI-based desire to:
  - Stop depending on Firmware Builds
  - Start relying on Control Values

Icon by: prettycons
Benefits

Time to Market

✗ No Wait for new firmware from dual-source vendors
✓ Adjust some Control Values wash … rinse … repeat

Reliability

✓ Many organizations exercising same firmware
“Language” Overview
Shoehorning Complicated Behaviors into 3 Bits
Loading Host Queuing Rules into HDD

“Language” sets the meaning for each of 7 priority classes

Command Priority Management Service

Assign Priority Class

New Command

Mode Page SATA Log

Priority Class Lookup Establishes HDD Queuing Behavior
“Language” Elements
One Elements Instance per Priority Class

- Cornerstone
- Requested Latency
- Add Teeth to Request
- Latency Miss Action
- Basic Latency Target Controls
- Non-Conformance Criteria & Action
- Load Shedding
- Predictive Miss Action
- Terminate Disaster in the Making
- Adapt to Latency Misses

Gen Z Queuing
The **Other Shoe Drops**

This is a Number Right __Minus__ Number Wrong Quiz

- Device must confess its limitations
  - Log number of commands that missed
    - Latency Misses
    - Non-Conformance Incidents
    - Preemptive Misses
  - With **Index** as a qualifier

- Priority Class Definitions are complex
  Feedback from the device is essential
“Language” Definition v0.7 Issues

🌟 Current Actions Assume

🔹 Miss Policy == Failure
🔹 Miss Policy == Rush Delivery

Proposed on OCP Storage Reflector w/ no responses

🌟 Should Load Sheading Actions Include Unit Attention Conditions

🌟 What Verbs Are Missing from the “Language”

Photo credit: Palo Cech
Discussion
Backup Slides
“Recent” Sightings

- March ‘19 – T10 approves Command Duration Limits (after a year of haggling)

- June ‘19 – T13 begins its discussions
  - See sow's ear
  - Presses for silk purse
  - Focus turns to OCP Storage discussion in August ‘19

- September ‘19 – More Innovation

- October ‘19 – “Need Input!” - *Short Circuit* ©1986
  Back to OCP Storage
“Recent” Sightings
OCP Fast Fail Read

→ Approved: T10 March ‘19 & T13 October ‘19

★ Inactive Time Limit & Active Time Limit

★ Avoids Wasting HDD Resources when data is being obtained from elsewhere

📖 Fundamentally …

✗ About when something didn't work
✓ Not about making more useful queues

Photo credit: Dids
Beyond the Basics
Controls Overlooked by Previous Efforts

✔️ What is the price of failure? …
  ✔️ On just this command
  ✔️ On too many similar commands

✔️ What to do about impossible-to-achieve requests?

⭐ Define a multiple-choice list of actions for the device to take in each of the above case

✔️ Reminder: Device only converts LATENCY TARGET into an internal goal
  ✗ unless something goes wrong
  ✔️ for success, conversion is the only new overhead
Multiple-Choice List of Actions

- Keep on trying
- Throw in the towel (two or three choices)
- Try a different set of Latency Controls
- Turn off Latency Controls (for this command)
- ...
Latency Controls Index as a Technology

- Index selects a Latency Controls recipe
- Recipe is written in a work-in-progress language
- Management Software stores recipes in device
- I/O Stack picks the right recipe for each read or write

This is a major advance over any previous effort
Latency Controls Index + Recipe is a Bleeding Edge Technology

✓ Standards are Meant to do This Kind of Work
✓ Connect two ends of a wire with Intervening Tools that allow both ends to function efficiently
✓ Allow the same tools to work equally well for most consumers on either end of the wire
✓ Allow bugs found by one consumer to be fixed for all consumers

✓ Recipes designed to allow devices freedom to maximize resource utilization
  + More value out of existing hardware platforms

✗ There's no free lunch
✗ Optimal recipes aren't like low-hanging fruit (WD is looking for a partner)
☆ 20% – 50% throughput increases possible (without sacrificing key latency guarantees)