



**SDC** 18

September 24-27, 2018  
Santa Clara, CA

[www.storagedeveloper.org](http://www.storagedeveloper.org)

# Key Value Storage Standardization Progress

**Bill Martin**

**Samsung**

# Disclaimer

This presentation and/or accompanying oral statements by Samsung representatives collectively, the “Presentation”) is intended to provide information concerning the SSD and memory industry and Samsung Electronics Co., Ltd. and certain affiliates (collectively, “Samsung”). While Samsung strives to provide information that is accurate and up-to-date, this Presentation may nonetheless contain inaccuracies or omissions. As a consequence, Samsung does not in any way guarantee the accuracy or completeness of the information provided in this Presentation.

This Presentation may include forward-looking statements, including, but not limited to, statements about any matter that is not a historical fact; statements regarding Samsung’s intentions, beliefs or current expectations concerning, among other things, market prospects, technological developments, growth, strategies, and the industry in which Samsung operates; and statements regarding products or features that are still in development. By their nature, forward-looking statements involve risks and uncertainties, because they relate to events and depend on circumstances that may or may not occur in the future. Samsung cautions you that forward looking statements are not guarantees of future performance and that the actual developments of Samsung, the market, or industry in which Samsung operates may differ materially from those made or suggested by the forward-looking statements in this Presentation. In addition, even if such forward-looking statements are shown to be accurate, those developments may not be indicative of developments in future periods.

# What is Key Value Storage?

- ❑ Storing an object (value) associated with a key
- ❑ A New paradigm that is different than block storage
- ❑ Different than Object Storage



# How is Key Value different than Object Storage

- ❑ Object storage
  - ❑ A solution or platform
    - ❑ A cluster of storage in general, Global namespace, Extensive higher layer features: global namespace, resiliency, etc.
  - ❑ A device (e.g., Kinetic)
    - ❑ No storage native interface (e.g., object(REST) interface)
  - ❑ Searchable based on the value in the object
  - ❑ Performs operations based on the object key and object(value)
- ❑ Key Value Storage
  - ❑ A device
  - ❑ Native key and value interface(e.g., store(), retrieve(), delete(), etc. with key and value instead of LBA)
  - ❑ Searches and other operations on the value are done outside of the storage element

# Why Key Value Storage

- ❑ Solid State Storage maps from an address to a physical location already
- ❑ This is a different mapping but what SSS already does
- ❑ This removes the triple mapping that occurs in current systems
  - ❑ Mapping from key to filesystem
    - ❑ Done in the host
  - ❑ Mapping from filesystem to Logical Block Address
    - ❑ Done in the host
  - ❑ Mapping from Logical Block Address to Physical address
    - ❑ Done in the storage device

# Where is Key Value Storage being standardized

- ❑ SNIA
  - ❑ Key Value Storage API
- ❑ NVMe
  - ❑ Key Value Command set

# SNIA KVS API

- ❑ Current revision 0.16 has been released for public review
  - ❑ <http://www.snia.org/publicreview>
- ❑ What is included
  - ❑ Keyspace management
  - ❑ Store
  - ❑ Retrieve
  - ❑ Exist
  - ❑ List
  - ❑ Delete
  - ❑ Group operations

# Keyspace management

- ❑ Multiple Keyspaces coexist on a single key Value device
- ❑ Each Keyspace has its own characteristics
  - ❑ Capacity
  - ❑ Key ordering
  - ❑ Device characteristics are global



# Store

- ❑ Specifies
  - ❑ Keyspace
  - ❑ Key
  - ❑ Value
  - ❑ Options
    - ❑ Compression
    - ❑ Overwrite
    - ❑ Update only

# Retrieve

- ❑ Specifies
  - ❑ Keyspace
  - ❑ Key
  - ❑ Options
    - ❑ Decompress
    - ❑ Delete after retrieve

# List

- ❑ Returns a list of keys
- ❑ Has a starting point
- ❑ Has a size of what is to be returned
  - ❑ Based on the key size determines number of keys to return

# Exist

- ❑ Returns success if value exists
- ❑ May be performed on a list of values

# Group operations

- ❑ Allows operations to be performed on a group of keys specified by
  - ❑ A set of bits within the key
  - ❑ A mask of those bits
- ❑ Supported operations
  - ❑ List
  - ❑ Delete

# SNIA KV API Status

- ❑ Being developed in the Object Drive TWG
- ❑ Meeting weekly
- ❑ Currently revision 0.16 available for public review
- ❑ Aiming for release as a specification end of this year

# NVMe Command Set

- ❑ In architectural development
- ❑ A new command set
- ❑ Operates over
  - ❑ PCIe
  - ❑ NVMe-oF

# Architecture overview

- ❑ A single controller supports either
  - ❑ Block Commands
  - ❑ Keyvalue Commands
  - ❑ NOT both
- ❑ A NVM Subsystem may have
  - ❑ A Block Command controller; and
  - ❑ A Keyvalue Controller
- ❑ New specification that references
  - ❑ NVMe Base specification
  - ❑ NVMe-oF specification
  - ❑ NVMe-PCIe Specification



# Keyspace

- ❑ Comparable to Namespace
- ❑ Allows separate spaces on a controller to use the same keys without overlap
- ❑ Allows partitioning of the controller resources

# Store Command

- ❑ Keys up to 16 bytes are carried in the command
- ❑ Keys greater than 16 bytes are passed as part of the DATAPTR
  - ❑ May be SGL or PRP
- ❑ Value is pointed to by DATAPTR
  - ❑ May be SGL or PRP

# Retrieve Command

- ❑ Keys up to 16 bytes are carried in the command
- ❑ Keys greater than 16 bytes are passed as part of the DATAPTR
  - ❑ May be SGL or PRP
- ❑ Value is returned to location pointed to by DATAPTR
  - ❑ May be SGL or PRP

# List Command

- ❑ Returns a list of keys within the key space
- ❑ Starts at the key specified by the command
- ❑ Keys up to 16 bytes are carried in the command
- ❑ Keys greater than 16 bytes are passed as part of the DATAPTR
  - ❑ May be SGL or PRP

# Exist Command

- ❑ Returns success if key exists in the key space
- ❑ Keys up to 16 bytes are carried in the command
- ❑ Keys greater than 16 bytes are passed as part of the DATAPTR
  - ❑ May be SGL or PRP

# NVMe KV Status

- ❑ Provisionally approved by NVMe Board
- ❑ Developing architecture
- ❑ Weekly subgroup meetings
- ❑ Pushing for member review around end of 2018

# Thank You



**SDC** 18

September 24-27, 2018  
Santa Clara, CA

[www.storagedeveloper.org](http://www.storagedeveloper.org)

**Title of Presentation**

**Presenter Name**

**Presenter Company**



- Subhead

  - Example 1

  - Example 2

- Subhead



