



# IP-Based Drive Management Specification

**Version 1.0**

*ABSTRACT:* This document describes the management features and functions of a storage device class known as IP Based Drives. It includes a taxonomy covering the scope of involved device capabilities.

This document has been released and approved by the SNIA. The SNIA believes that the ideas, methodologies and technologies described in this document accurately represent the SNIA goals and are appropriate for widespread distribution. Suggestions for revision should be directed to <http://www.snia.org/feedback/>.

**SNIA Technical Position**

January 10, 2017

## USAGE

The SNIA hereby grants permission for individuals to use this document for personal use only, and for corporations and other business entities to use this document for internal use only (including internal copying, distribution, and display) provided that:

1. Any text, diagram, chart, table or definition reproduced shall be reproduced in its entirety with no alteration, and,
2. Any document, printed or electronic, in which material from this document (or any portion hereof) is reproduced shall acknowledge the SNIA copyright on that material, and shall credit the SNIA for granting permission for its reuse.

Other than as explicitly provided above, you may not make any commercial use of this document, sell any or this entire document, or distribute this document to third parties. All rights not explicitly granted are expressly reserved to SNIA.

Permission to use this document for purposes other than those enumerated above may be requested by e-mailing [tcmd@snia.org](mailto:tcmd@snia.org). Please include the identity of the requesting individual and/or company and a brief description of the purpose, nature, and scope of the requested use.

Copyright (c) 2014, 2015, 2016, 2017 The Storage Networking Industry Association.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- \* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- \* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- \* Neither the name of The Storage Networking Industry Association (SNIA) nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

## **DISCLAIMER**

The information contained in this publication is subject to change without notice. The SNIA makes no warranty of any kind with regard to this document, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. The SNIA shall not be liable for errors contained herein or for incidental or consequential damages in connection with the furnishing, performance, or use of this document.

Suggestions for revisions should be directed to <http://www.snia.org/feedback/>.

Copyright © 2014-2017 TWG. All rights reserved. All other trademarks or registered trademarks are the property of their respective owners.

## Revision History

Revision	Date	Sections	Originator	Comments
0.0.1	Dec. 22, 2015	All	Mark Carlson, James Espy	First draft combining contents of an earlier requirements white paper and management spec draft.
0.002	Dec. 23, 2015	Various spots	James Espy	Updates to contributor list and various editor notes.
0.003	Jan 18, 2016	Various spots	James Espy	Updates from review comments.
0.004	Jan 18, 2016	2.1 References 4.3 Acronyms 6 Management	James Espy	Acronyms additions. Section 6 organization update.
0.005	Feb 8, 2016	7 Use Cases 5.2 Electrical	James Espy	Removed section 7. Modified table in section 5.2.
0.006	March 14, 2016	4.2 Definitions	James Espy	Modified (added) definitions
0.007	March 22, 2016	Sections 5 and 6	James Espy	Selected text updates. Pictorial updates. Moved Redfish manageability example to Annex.
0.008	May 10, 2016		James Espy	Abstract and introduction updates. Table 1 References update; 5.2 updates; 5.5 Taxonomy depiction update; 6 and 7 updates;
0.009	May 11, 2016		Mark and Jim	Multiple changes of cleanup and change accepting
0.010	May 24, 2016	Many	James Espy	Consistent use of IP-Based Drive throughout document
0.011/12	Aug 05, 2016	Many	TWG team	Format updates;
0.013/14	Sept 27, 2016	Many	TWG team, James Espy, Mike Raineri	Format updates; Alternate Redfish proposal
.015	Sept 27, 2016	Figure 4	Mark Carlson	Cleanup with "This Specification"
016	Oct 4, 2016	Many	TWG team	Cleanup
016a	Oct 11, 2016	Many	TWG team	Alternate Redfish proposal accepted. Annex A deleted. Other editing.
016b	Oct 11, 2016	Many	TWG team	Accepted changes and notes for tech writer.
Version 1.0	Jan. 10, 2017	All (Editorial)	TWG team	Reformatted as a SNIA Technical Position

## Intended Audience

This document is intended for use by individuals and companies engaged in developing storage systems utilizing IP-Based Drive devices and/or related sub-systems.

## Changes to the Specification

Each publication of this document is uniquely identified by a three-level identifier, comprised of a version number, a release number and an update number. The current identifier for this document is listed on the title page of this document. Future publications of this document are subject to specific constraints on the scope of change that is permissible from one publication to the next and the degree of interoperability and backward compatibility that should be assumed between products designed to different publications of this standard. The SNIA has defined three levels of change to a specification:

- **Major Revision:** A major revision of the specification represents a substantial change to the underlying scope or architecture of the specification. A major revision results in an increase in the version number of the version identifier (e.g., from version 1.x.x to version 2.x.x). There is no assurance of interoperability or backward compatibility between releases with different version numbers.
- **Minor Revision:** A minor revision of the specification represents a technical change to existing content or an adjustment to the scope of the specification. A minor revision results in an increase in the release number of the specification's identifier (e.g., from x.1.x to x.2.x). Minor revisions with the same version number preserve interoperability and backward compatibility.
- **Update:** An update to the specification is limited to minor corrections or clarifications of existing specification content. An update will result in an increase in the third component of the release identifier (e.g., from x.x.1 to x.x.2). Updates with the same version and minor release levels preserve interoperability and backward compatibility.

## About SNIA

The Storage Networking Industry Association (SNIA) is a non-profit organization made up of member companies spanning information technology. A globally recognized and trusted authority, SNIA's mission is to lead the storage industry in developing and promoting vendor-neutral architectures, standards and educational services that facilitate the efficient management, movement and security of information. For additional information, visit the SNIA web site at [www.snia.org](http://www.snia.org).

## Acknowledgements

The SNIA Object Storage Technical Working Group (TWG), which developed and reviewed this document, would like to recognize the significant contributions made by the following members:

Mark Carlson.....	Toshiba
David Slik.....	NetApp
James Espy .....	Dell-EMC
Michael Raineri .....	Dell-EMC
Paul Suhler .....	WDC
Bill Martin .....	Samsung
Don Heins .....	Seagate

# Contents

<b>1</b>	<b>Introduction .....</b>	<b>9</b>
<b>2</b>	<b>Normative References .....</b>	<b>10</b>
<b>3</b>	<b>Scope .....</b>	<b>11</b>
<b>4</b>	<b>Definitions, Symbols, Abbreviations, and Conventions .....</b>	<b>12</b>
4.1	Overview .....	12
4.2	Definitions .....	12
4.3	Acronyms and Abbreviated Terms .....	13
4.4	Keywords .....	14
4.5	Conventions .....	15
<b>5</b>	<b>IP-Based Drive Characteristics and Requirements .....</b>	<b>16</b>
5.1	Physical Layer – Form Factor.....	16
5.2	Electrical .....	16
5.3	Link Layer .....	16
5.4	Environmental .....	17
5.5	Taxonomy .....	17
<b>6</b>	<b>IP-Based Drive Management.....</b>	<b>20</b>
6.1	Overview .....	20
6.2	Establish Physical Link.....	20
6.3	Assign IP Address.....	20
6.4	Discover Basic Services.....	20
6.5	Redfish Manageability .....	21

## List of Tables and Figures

Table 1 - References.....	10
Table 2 - Utilization of I2C and Power Disable between chassis and drives based on type.....	16
Table 3 - Tabular taxonomy .....	17
Figure 1 – Traditional Drive and Kinetic KV Drive.....	18
Figure 2 – Pre-configured and User-configured.....	18
Figure 3 - Interposers.....	19
Figure 4 - Taxonomy Composite Depiction .....	19



# 1 Introduction

The growing popularity of object based storage has resulted in the development of Ethernet-connected storage devices, herein referred to as IP-Based Drives and subsystems supporting object interfaces and in some cases the ability to run localized applications. This document specifies management features of these devices.

A number of storage solutions, as part of open source and other projects, are architected to scale out by incrementally adding and removing functional storage nodes. Example projects include Hadoop HDFS, Ceph, and Swift (OpenStack object storage).

Typical scale-out storage nodes consist of relatively inexpensive enclosures with IP network connectivity, CPU, Memory and Direct Attached Storage (DAS). While inexpensive to deploy, these solutions become harder to manage over time. Power and space requirements of data centers are difficult to meet with this type of solution. IP-Based Drive vendors look to enable solutions that better meet data center requirements by re-partitioning solutions into drive-based storage nodes and creating points of interoperability.

Some of the aspects in this document may be met by various designs that fall outside of a typical drive form factor and yet may still interoperate at some level, such as the object protocol. Examples are an interposer that has two different interfaces (e.g. converting Ethernet to SATA) or an entire enclosure comprised of standard storage devices fronted by an object interface processing unit that pools the individual drives.

## 2 Normative References

Table 1 lists the standards, specifications and other documents related to this document.

**Table 1 - References**

Author/Owner	Title	Revision	URL
SNIA	TLS Specification for Storage Systems	1.0.1	<a href="http://www.snia.org/sites/default/files/TLSs_pec-v1.01_Technical_Position.pdf">http://www.snia.org/sites/default/files/TLSs_pec-v1.01_Technical_Position.pdf</a>
Open Compute Project	Storage Device with Ethernet Interface	4	<a href="http://www.opencompute.org/wiki/storage">http://www.opencompute.org/wiki/storage</a> <a href="http://files.opencompute.org/oc/public.php?service=files&amp;t=3f4d2fedf373ae0b29a2b86f0255da1b">http://files.opencompute.org/oc/public.php?service=files&amp;t=3f4d2fedf373ae0b29a2b86f0255da1b</a>
IEEE	IEEE 802.3cb		<a href="http://www.ieee802.org">http://www.ieee802.org</a>
	RFC 4171 - iSNS		<a href="https://www.ietf.org/rfc.html">https://www.ietf.org/rfc.html</a>
	RFC 791 - IPV4		<a href="https://www.ietf.org/rfc.html">https://www.ietf.org/rfc.html</a>
	RFC 2460 - IPV6		<a href="https://www.ietf.org/rfc.html">https://www.ietf.org/rfc.html</a>
DMTF	Redfish		<a href="http://www.dmtf.org/standards/redfish">http://www.dmtf.org/standards/redfish</a>
	RFC 2131 – DHCP		<a href="https://www.ietf.org/rfc.html">https://www.ietf.org/rfc.html</a>
	RFC 2132 - DHCP Options and BOOTP		<a href="https://www.ietf.org/rfc.html">https://www.ietf.org/rfc.html</a>
	RFC 1035 - DNS		<a href="https://www.ietf.org/rfc.html">https://www.ietf.org/rfc.html</a>
	RFC 6762 Multi-case Domain name Service		<a href="https://www.ietf.org/rfc.html">https://www.ietf.org/rfc.html</a>
	RFC 6763 – mDNS/DNS-SD		<a href="https://www.ietf.org/rfc.html">https://www.ietf.org/rfc.html</a>
	SAS-3		<a href="http://www.t10.org">http://www.t10.org</a>
SFF Committee	SFF-8201 – 2.5” Drive Form Factor Dimensions		<a href="http://www.sffcommittee.org/">http://www.sffcommittee.org/</a>
SFF Committee	SFF-8223 – 2.5” Drive Form Factor with Serial Connector		<a href="http://www.sffcommittee.org/">http://www.sffcommittee.org/</a>
SFF Committee	SFF-8301 – 3.5” Drive Form Factor Dimensions		<a href="http://www.sffcommittee.org/">http://www.sffcommittee.org/</a>
SFF Committee	SFF-8323 – 3.5” Drive Form Factor with Serial Connector		<a href="http://www.sffcommittee.org/">http://www.sffcommittee.org/</a>
SFF Committee	SFF-8601	0.7	<a href="http://www.sffcommittee.org/">http://www.sffcommittee.org/</a>
SFF Committee	SFF-8639		<a href="http://www.sffcommittee.org/">http://www.sffcommittee.org/</a>
SFF Committee	SFF-9639	1.0	<a href="http://www.sffcommittee.org/">http://www.sffcommittee.org/</a>

### 3 Scope

This document focuses on devices and interfaces within drive (e.g. HDD and SSD) form factors. Larger and smaller form factors and varying physical connectivity are not precluded.

Additionally, there is no bias or advocacy toward any particular data persistence technology. IP-Based Drive devices can utilize spinning media of virtually any recording technology along with flash and non-volatile solid state technology.

This document is focused on an in-band management API that can be used for scale-out management of IP-Based Drives using the IP protocol.

This document is divided into several sections. The first lists key definitions, conventions, etc. The second describes fundamental attributes of IP-Based Drives. The third details management functionality and represents the main content of this document. A separate *IP-Based Drive Mgmt Spec.zip* file is provided that shows examples.

## 4 Definitions, Symbols, Abbreviations, and Conventions

### 4.1 Overview

For the purposes of this document, the terms and definitions given in the SNIA 2016 Dictionary ([www.snia.org/education/dictionary](http://www.snia.org/education/dictionary)) apply. In cases where the current definitions in the SNIA dictionary conflict with those presented in this document, the definitions in this document shall be assumed to be correct.

ISO and IEC maintain terminological databases for use in standardization at the following addresses:

- IEC Electropedia: available at <http://www.electropedia.org/>
- ISO Online browsing platform: available at <http://www.iso.org/obp>

### 4.2 Definitions

#### 4.2.1

##### **key value storage**

Type of object storage interface where a key is used to address the associated object

#### 4.2.2

##### **object**

Encapsulation of data and associated meta-data

#### 4.2.3

##### **object drive**

Storage element that directly provides object services

#### 4.2.4

##### **object service**

Object-level access to storage

#### 4.2.5

##### **object storage**

Storage device that provides object services. Object storage includes DSaaS

### 4.3 Acronyms and Abbreviated Terms

API	Application Programming Interface
ATA	Advanced Technology Attachment
CDMI	Cloud Data Management Interface
DAS	Direct Attached Storage
DHCP	Dynamic Host Configuration Protocol
DMTF	Distributed Management Task Force
DNS	Domain Name Service
DSaaS	Data Storage as a Service
GbE	Gigabit Ethernet
HDFS	Hadoop Distributed File System
HDD	Hard Disk Drive
HTTP	HyperText Transfer Protocol
I2C	Inter-Integrated Circuit
IEC	International Electrotechnical Commission
INCITS	International Committee on Information Technology Standards
IP	Internet Protocol (Alt. Intellectual Property)
ISO	International Standards Organization
LLDP	Link Level Discovery Protocol
KV	Key Value
NTP	Network Time Protocol
NVM	Non-Volatile Memory
NVMe	Non-Volatile Memory Express
OCP	Open Compute Project
OS	Operating System
PCIe	Peripheral Component Interface Express
PMR	Perpendicular Magnetic Recording
RFC	Request For Comment
SAS	Serial Attached SCSI
SATA	Serial ATA Interface
SCSI	Small Computer Systems Interface
SFF	Small Form Factor
SNIA	Storage Network Industry Association
SMR	Shingled Magnetic Recording
SSD	Solid State Drive
SSDP	Simple Service Discovery Protocol

SW	Software
T10	INCITS SCSI Technical Committee
T13	INCITS AT Attachment Technical Committee
TCP	Transmission Control Protocol
TLS	Transport Level Specification
TWG	Technical Working Group
URL	Uniform Resource Locator

## 4.4 Keywords

### 4.4.1

#### **expected**

A keyword used to describe the behavior of the hardware or software in the design models presumed by this standard. Other hardware and software design models may also be implemented.

#### **mandatory**

A keyword indicating an item that is required to be implemented as defined in this document to claim compliance with this document.

#### **may**

A keyword that indicates flexibility of choice with no implied preference.

#### **obsolete**

A keyword indicating that an item was defined in prior revisions to this document but has been removed from this revision.

#### **optional**

A keyword that describes features that are not required to be operational. However, if any optional feature is operational, it shall be implemented as defined in this document.

#### **prohibited**

A keyword used to describe a feature or behavior that is not allowed to be present.

#### **required**

A keyword used to describe a behavior that shall be implemented.

#### **reserved**

A keyword referring to bits, bytes, words, fields, and code values that are set aside for future standardization.

Note 1 to entry: A reserved bit, byte, word or field shall be set to zero, or in accordance with a future extension to this standard.

Note 2 to entry: Recipients are not required to check reserved bits, bytes, words or fields for zero values; receipt of reserved code values in defined fields shall be reported as an error.

#### **shall**

A keyword indicating a mandatory requirement.

#### **should**

A keyword indicating flexibility of choice with a preferred alternative; equivalent to the phrase “it is recommended”.

## 4.5 Conventions

Certain words and terms used in this document have a specific meaning beyond their normal English meaning. These words and terms are defined either in 4.2 or in the text where they first appear.

Numbers that are not immediately followed by lower-case b or h are decimal values.

Numbers immediately followed by lower-case b (xxb) are binary values.

Numbers immediately followed by lower-case h (xxh) are hexadecimal values.

Hexadecimal digits that are alphabetic characters are upper case (i.e., ABCDEF, not abcdef).

Hexadecimal numbers may be separated into groups of four digits by spaces. If the number is not a multiple of four digits, the first group may have fewer than four digits (e.g., AB CDEF 1234 5678h)

Storage capacities shall be reported in base-10. IO transfer sizes and offsets shall be reported in base-2. The associated units and abbreviations used in this document are:

- A kilobyte (KB) is equal to 1,000 ( $10^3$ ) bytes.
- A megabyte (MB) is equal to 1,000,000 ( $10^6$ ) bytes.
- A gigabyte (GB) is equal to 1,000,000,000 ( $10^9$ ) bytes.
- A terabyte (TB) is equal to 1,000,000,000,000 ( $10^{12}$ ) bytes.
- A petabyte (PB) is equal to 1,000,000,000,000,000 ( $10^{15}$ ) bytes.
- An exabyte (EB) is equal to 1,000,000,000,000,000,000 ( $10^{18}$ ) bytes.
- A kibibyte (KiB) is equal to  $2^{10}$  bytes.
- A mebibyte (MiB) is equal to  $2^{20}$  bytes.
- A gibibyte (GiB) is equal to  $2^{30}$  bytes.
- A tebibyte (TiB) is equal to  $2^{40}$  bytes.
- A pebibyte (PiB) is equal to  $2^{50}$  bytes.
- An exibyte (EiB) is equal to  $2^{60}$  bytes.

## 5 IP-Based Drive Characteristics and Requirements

### 5.1 Physical Layer – Form Factor

Drives may be delivered in different standard physical dimensions. Example form factors as defined in the references noted in Table 1 include:

- SFF-8201 2.5” Drive Form Factor Dimensions;
- SFF-8223 2.5” Drive Form Factor with Serial Connector;
- SFF-8301 3.5” Drive Form Factor Dimensions; or
- SFF-8323 3.5” Drive Form Factor with Serial Connector.

### 5.2 Electrical

This document is for Ethernet connected media. Other media are possible and would not be outside the definition of an IP-Based Drive.

A standard for the modified pin-out of SFF-8639 (Serial Attachment 12 Gb/s 6X Unshielded Connector) is documented in SFF-9639.

IP-Based Drives shall implement the pin out in the SFF-9639 (overlay of SFF-8639) SNIA Ethernet Drive.

The utilization of I<sup>2</sup>C and Power Disable between chassis and drives based on type is shown in Table 2.

**Table 2 - Utilization of I<sup>2</sup>C and Power Disable between chassis and drives based on type**

	SNIA Ethernet Drive	OCP Kinetic Drive
SNIA Ethernet Drive chassis	Drive and/or chassis may support Power Disable and/or I <sup>2</sup> C	Drive supports I <sup>2</sup> C; chassis may support I <sup>2</sup> C
OCP Kinetic Drive chassis	Chassis supports I <sup>2</sup> C; Drive may support I <sup>2</sup> C	Drive and chassis support I <sup>2</sup> C

### 5.3 Link Layer

An IP-Based Drive shall support IPv4 (see RFC 791) and should support IPv6 (see RFC 2460).

An IP-Based Drive shall support operation at 1 Gb/s, and may support 2.5 Gb/s, 5 Gb/s, or higher in any combination. If more than one speed is supported, then an IP-Based Drive shall perform speed auto-negotiation at power-on (see SFF-8601 and IEEE 802.3cb). The goal of speed negotiation is to achieve the highest speed possible by both sides of a connection.

The SFF 8601 specification defines the speed negotiation process for Ethernet drives, and the LLDP-based port auto-negotiation protocol for Ethernet drives that do not support IEEE 802.3 CL73 auto-negotiation.

IP-Based Drives:

- Shall implement SFF-8601;
- May implement the OCP I<sup>2</sup>C management functionality defined in “[Storage device with Ethernet Interface](#)” v4; and
- May implement manual link speed assignment (run at other than negotiated speed).



## 5.4 Environmental

This document specifies a discovery and reporting mechanism to provide information on device environmental characteristics (e.g. power and temperature).

## 5.5 Taxonomy

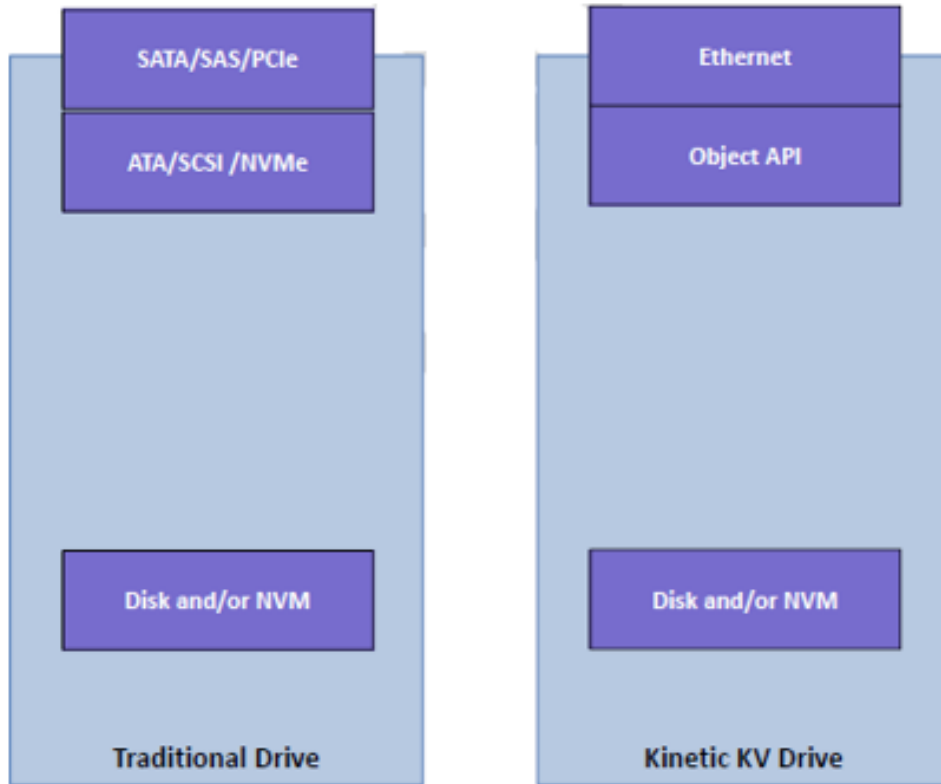
An IP-Based Drive may be defined in many forms. This taxonomy lists several forms with the idea that innovation could expand this list over time.

A tabular taxonomy is shown in Table 3.

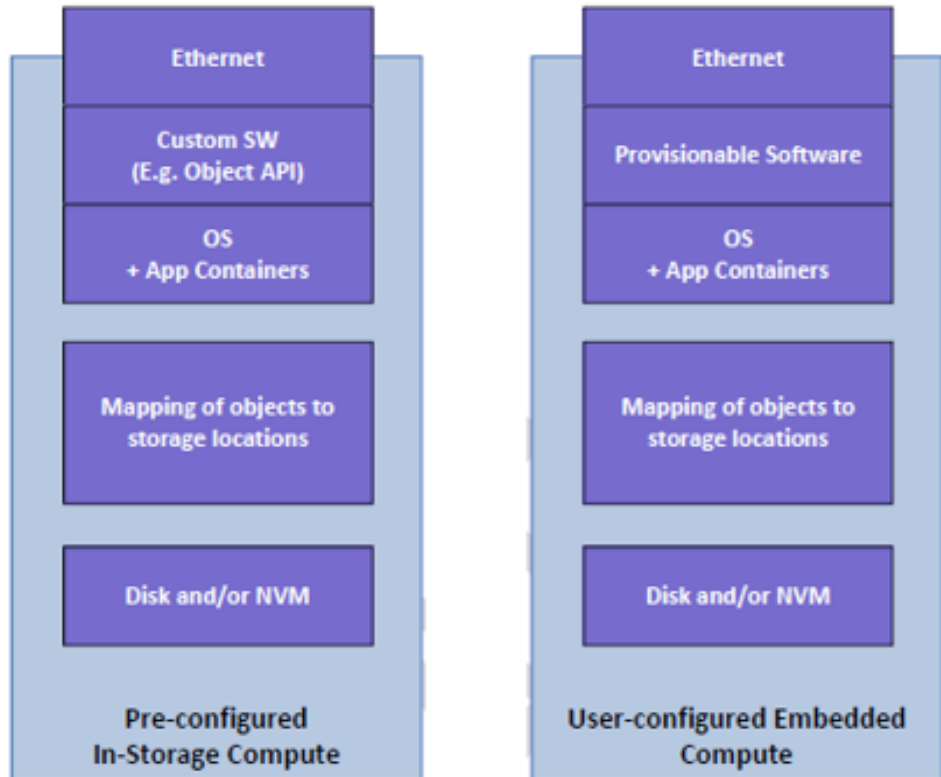
**Table 3 - Tabular taxonomy**

Drive Type	Host Connect	Protocol Examples	Management Defined In...	Comments
Traditional	SAS SATA NVMe	T10 SCSI/block T13 SATA/block	T10/T13/ NVMeexpress	Low level interface. Complex host stack. Host sees block interface (PMR, SMR ...). Lossless SAS/SATA w/limited routability. Typically single host/client.
Key-Value (KV)	GbE	e.g., Kinetic/KV	This standard and the appropriate object protocol documentation	High level KV interface. Simplified host stack. Host sees object interface (drive abstracted). Lossy connectivity (TCP/IP), full routability. Multi-host/client. Failover for availability.
Pre-Configured In-Storage Compute	GbE	Object/other e.g., Kinetic/Ceph	This standard and the appropriate object protocol documentation	Custom apps (e.g. object) installed at factory. Apps run in an embedded environment. Lossy connectivity (TCP/IP), full routability. Multi-host/client. Failover for availability.
User-Configured Embedded Compute	GbE	Object/other e.g., Kinetic/Ceph	This standard and the appropriate object protocol documentation	Apps installed anytime. Apps run in embedded environment. Lossy connectivity (TCP/IP), full routability. Multi-host/client. Failover for availability.
Interposer (addition to drive)	GbE	Object/other e.g., Kinetic/Ceph	This standard and the appropriate object protocol documentation	Allows standard block drives to be used as IP-Based Drives. Allows collections of standard block drives to be virtualized as IP-Based Drives. Lossy connectivity (TCP/IP), full routability. Multi-host/client. Failover for availability.

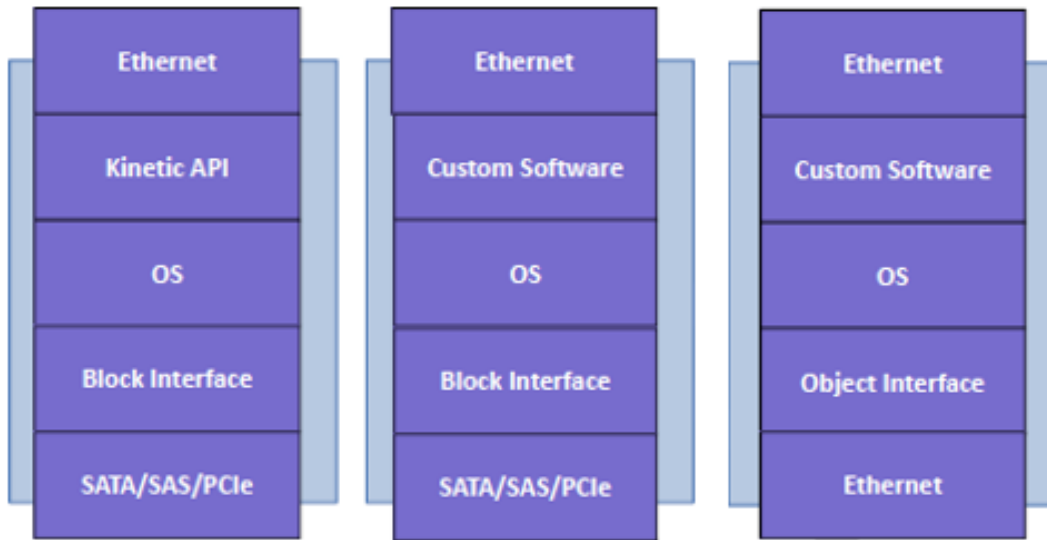
The taxonomy row items are depicted in Figure 1, Figure 2, and Figure 3, with a composite depiction in Figure 4.



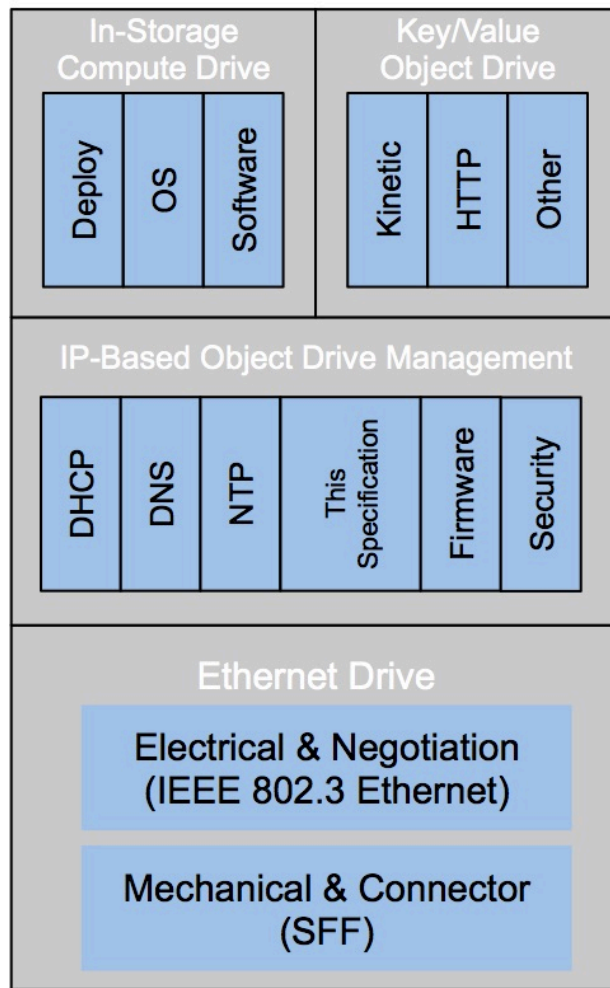
**Figure 1 – Traditional Drive and Kinetic KV Drive**



**Figure 2 – Pre-configured and User-configured**



**Figure 3 - Interposers**



**Figure 4 - Taxonomy Composite Depiction**

## 6 IP-Based Drive Management

This section describes management functionality at the device level. This can be expanded to include subsystem enclosures and multi-rack configurations.

### 6.1 Overview

There are two aspects to device discovery and management:

1. Management Discovery;
2. Service Discovery.

Management Discovery is described in this section. Service Discovery is the responsibility of the service (application) operating on the device and is beyond the scope of this document.

Management Discovery has the following structure and sequence further described in this section:

- Establish Physical link (Ethernet, LLDP);
- Assign IP Address (DHCP);
- Discover Basic Services (DNS-Name, NTP-Time);
- Provide Management Services (Redfish).

### 6.2 Establish Physical Link

It is assumed that a physical Ethernet link has been established using LLDP. See section 5.3.

### 6.3 Assign IP Address

IP-Based Drives shall support IPv4 and shall support IPv6.

If more than one port is implemented, such ports may be configured for redundancy or other purposes.

Link aggregation, if needed may be done through LLDP and as such is not part of this document.

### 6.4 Discover Basic Services

The discovery of IP-Based Drive basic services involves finding its access point on a network and perhaps its position in whatever enclosure it is located. It also involves determining the IP-Based Drive's capabilities. At a high level, if the drive serves up a native protocol (Key/Value or other) the type and protocol version are important (other lower level capabilities should be discovered through the protocol itself). If the drive is capable of hosting software in a general purpose computing environment, that capability should be discoverable as well. In this case, an interoperable means to load and manage the software on the drive is required.

Redfish references SSDP (Simple Service Discovery Protocol) for discovery. IP-Based Drives should use the Redfish optional discovery mechanism.

## 6.5 Redfish Manageability

The following standard Redfish services off the Redfish Service Root shall be implemented:

- Account Service
- Session Service
- Chassis Collection
- Manager Collection
- Computer System Collection

The following standard Redfish services off the Redfish Service Root should be implemented:

- Update Service

The Chassis resources shall report the “ChassisType” property as “IPBasedDrive”.

The Chassis resources should support the following properties: “Status”, “Manufacturer”, “Model”, “SKU”, “PartNumber”, “SerialNumber”, “AssetTag”, “IndicatorLED”.

The Manager resources shall contain an Ethernet Interface Collection. This is required to manage IP address settings of the manager of the IP Based Drive.

The Computer System resources shall contain an Ethernet Interface Collection. This is required to manage IP address settings of the IP-Based Drive. The Computer System resources should support the following properties: “Status”, “Manufacturer”, “Model”, “SKU”, “PartNumber”, “SerialNumber”, “AssetTag”, “IndicatorLED”.

The Redfish implementation should support the Redfish standard Drive entity.

The Drive resources should support the following properties: “Status”, “Manufacturer”, “Model”, “SKU”, “PartNumber”, “SerialNumber”, “AssetTag”, “IndicatorLED”, “BlockSizeBytes”.