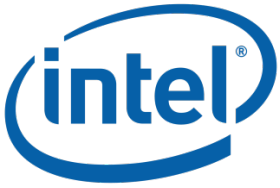


Intel® Rack Scale Design

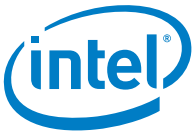
Architecture Specification
Software v2.2

December 19, 2017

Revision 001

`

Document Number: 336861-001



Contents

Contents

Figures

Tables

Revision History

Revision	Description	Date
001	Initial Release	December 19, 2017

§

1 Introduction

1.1 Scope

This document contains information about the Intel® Rack Scale Design (Intel® RSD) v2.2 solution as a reference implementation using Intel Platform technology. This reference implementation showcases Intel's vision for an industry leading Rack architecture that is modular and extensible, catering to the needs of the cloud and other server segments. The key value proposition for Intel® RSD systems is to reduce TCO, to achieve hyper-scale agility and improve data center operations.

This document provides the requirements, recommendations and/or optional criteria in order for your system to be considered Intel® RSD conformant. Rack Scale API conformance is described in the Intel® RSD PSME, PODM and RMM specifications.

1.2 Intended audience

The intended audiences of this document are:

- Hardware vendors (for example, OEMs/ODMs) who will build Intel® RSD platforms or Intel® RSD Platform components that are integrated into the Intel® RSD Platform.
- Software vendors (for example, ISVs/IBVs) who will implement the Intel® RSD API on Intel® RSD Platform hardware components and on supporting components.

1.3 Conventions

The key words/phrases "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119, Table

1.4 Acronym definitions

Table 1 contains definitions for the acronyms used in this document.

Table 1. Intel® RSD Acronym Definitions

Term	Definition
ACM	Authenticated Code Module. A security feature that is used to establish trust-level before executing the code. The processor or hardware first measures a digitally signed module (the ACM) that was provided by the chipset manufacturer. The ACM then measures the first BIOS module, and the first BIOS module then measures the next module, and so on.
AFU	Accelerator Function Unit. An FPGA that uses the AFU bitstream to provide end user functionality.
API	Application program interface. A set of routines, protocols, and tools for building software applications. API defines operations, inputs and outputs.
Authentication Server	An authentication server is an application that facilitates authentication of an entity that attempts to access a network. An authentication server can reside in a dedicated computer, an Ethernet switch or a network access server.
BIOS	Basic Input/Output System. Firmware that initializes and tests Compute/Storage Module hardware components, and loads a boot loader or an operating system from a mass memory device. The BIOS supports UEFI interface.
BMC	Baseboard Management Controller. A specialized service processor that monitors the physical state of a computer and provides services to monitor and control certain Compute/Storage Module operations. The BMC supports the Intelligent Platform Management Interface (IPMI) or Redfish.
CPP	Control Plane Processor. A hardware element that runs the Network Agent.
DHCP	Distributed Host Configuration Protocol. A standard network protocol used to dynamically distribute network configuration parameters, like IP addresses.

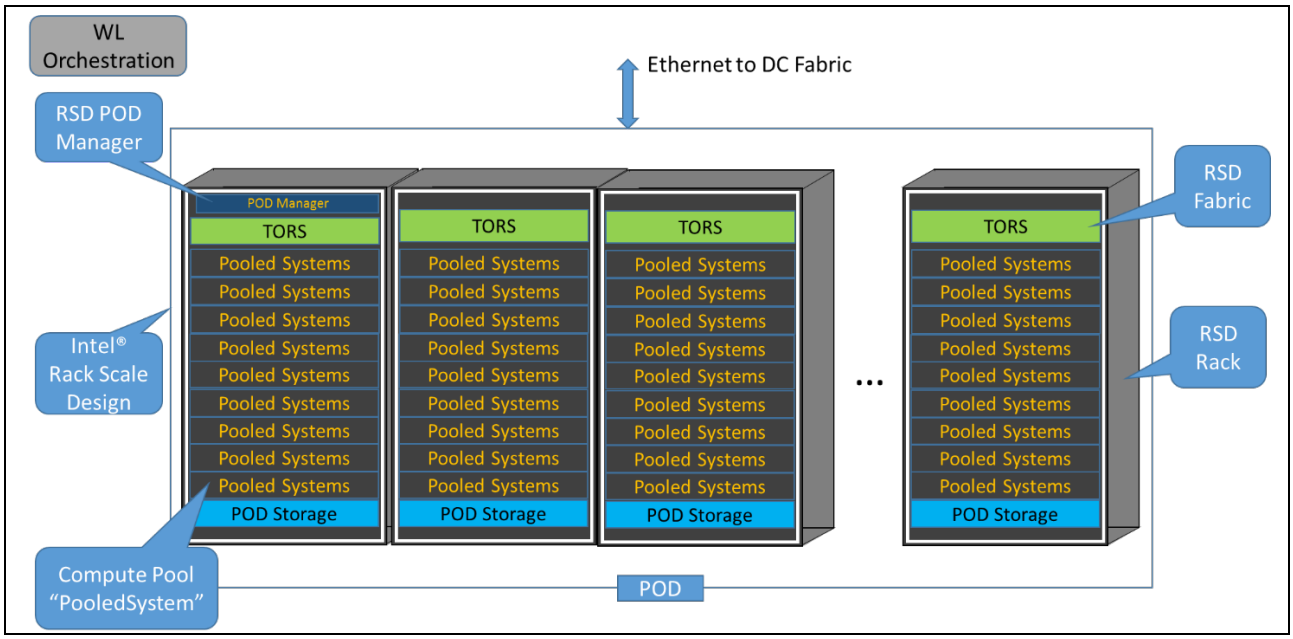
Term	Definition
DMC	Drawer Management Controller. A physical controller that manages the Drawer where the PSME functionality is normally implemented.
EORS	End-Of-Row Switch. In a Pod configuration that includes multiple Drawers, this switch in each Drawer connects the Drawer to all other Drawers in the Rack.
FPGA	A field-programmable gate array (FPGA) is an integrated circuit designed to be configured after manufacturing.
HA	High Availability. RSD performance requirements may dictate the use of High Availability (HA) designs as a redundancy feature. If a Rack supports HA RMM, then (by definition) there is more than one RMM present in the Rack. If the primary RMM fails, then the secondary RMM takes over immediately.
HII	Human Interface Infrastructure. An interface definition for the BIOS from the Unified Extensible Firmware Interface (UEFI) Forum
IE	Innovation Engine. The physical controller in the Module or Blade.
Intel TXT	Intel Trusted eXecution Technology.
iPXE	An open-source implementation of the PXE client firmware and boot loader.
JBOD	Just a Bunch Of Disks. A collection of storage devices consolidated in one chassis for easy serviceability.
NB	North Bound. Used to identify a direction of information transfer between the top and bottom layers of the software. From the PSME point of view, a transfer from the PSME to the PODM is considered a NB transfer.
NTP	Network Time Protocol. An Internet protocol used to synchronize the clocks of computers to a common time reference.
Intel ME	Intel Management Engine. A physical hardware resource that gives access to hardware features at the baseboard level below the OS.
MM	Module Manager. Firmware that runs on BMC/Intel ME/IE
MMC	Module Management Controller. The controller that manages the Blades in the module.
PM	Persistent Memory. Byte-accessible persistent memory that is produced by either DDR-based NVDIMM or high-density NVM technology.
PNC	Pooled Node Controller. A physical system element that provides connectivity and access control between the CPUs and the pool of NVMe storage, FPGA and Accelerators.
Pod	A logical and/or physical collection of Racks within a shared infrastructure management domain.
PODM	Pod Manager. The software that manages logical groupings of functionality across all infrastructure in a Pod.
PSME	Pooled System Management Engine. System management software that runs on the DMC and is responsible for the configuration of pooled Storage Modules by the PNC, the network (SDN), the Compute Modules, and the switches.
PTAS	Power Thermal Aware Scheduling. Software that manages computation within a data center in order to reduce thermal gradients, hotspots, and cooling fluctuations across the data center.
PXE	Preboot eXecution Environment. A specification that allows devices to boot over a network.
RAS	Reliability, Availability and Serviceability
REST	REpresentational State Transfer
RMC	Rack Management Controller. A physical system element that provides Rack management features.
RMM	Rack Management Module. A physical system element that is responsible for managing the Rack, which normally assigns IDs for the instances of PSME in the Rack, and manages the Rack power and cooling.
RMON	Remote network MONitoring. An open standard network monitoring specification.
RSD	Rack Scale Design
RSD API	Refers to PODM REST API, RMM REST API, PSME REST API combinations.
RSM	RSD Server Manager. A logical element (management software) that interfaces with the PODM to manage the RSD Platform.
RTM	Root of Trust for Measurement. A security feature; a component that can be trusted to reliably measure and report to the Root of Trust.
SB	South Bound. Used to identify a direction of information transfer between the top and bottom layers of the software. From the PODM point of view, a transfer from the PODM to the PSME is considered a SB transfer.
SDC	Software-Defined Data Center (an alternative term for SDI used in industry)
SDI	Software-Defined Infrastructure

Term	Definition
SDN	Software-Defined Network
SDS	Software-Defined Storage
SLA	Service Level Agreement
SOL	Serial-Over-LAN. A mechanism that allows a serial port's input/output to be redirected over IP.
SRIS	Separate Refclk Independent SSC. A reference clock forwarding system.
SSH	Secure SHell. An encrypted network protocol (versions: SSH-1, and SSH-2) that can be used to secure a remote login and other network services.
TLS	Transport Layer Security. A security protocol that provides connection security.
TPM	Trusted Platform Module. An international standard for a secure microprocessor that is dedicated to hardware security.
TORS	Top-Of-Rack Switch. A physical switch in each rack that connects the racks together to handle NB/SB traffic flow.

1.5 Intel® RSD platform overview

Figure 1 illustrates the various elements of the Intel® RSD Platform. In this scenario, the Pod consists of several Racks with resources that allow for Pod scaling. These resources (compute resources labeled Pooled System and storage resources labeled Pod Storage) are contained in several Racks. At the top of each Rack there is an interconnecting fabric (labeled TORS) that provides the Intel® RSD fabric connection among Racks. One Rack (on the left in this diagram) contains the Pod manager software that conducts the workload orchestration (WL Orchestration) and coordinates the activity in the Pod. The Pod manager must not shut down on rack-wide resets or power downs. This could be done by separate power to Pod manager with highly available power such as battery backup, redundant power or redundant Pod managers.

Figure 1. Intel® RSD 2.2 Pod Block Diagram



1.5.1 Major changes in Intel® RSD v2.1

The major changes from the Intel® RSD v1.2 definition to the Intel® RSD v2.2 definition are the addition of:

- PCIe* Direct Attach Pooled I/O
- Software components requirements

1.5.2 Major changes in Intel® RSD v2.2

The major changes from the Intel® RSD v2.1 definition to the Intel® RSD v2.2 definition are the addition of:

- TPM discovery
- FPGA discovery
- Deep discovery of the platform components and capabilities
- Network management services
- In-band BIOS/firmware update blocking
- Telemetry

1.5.3 Intel® RSD terminology definitions

Intel® RSD terminology conventions used in this document are defined in Table 2.

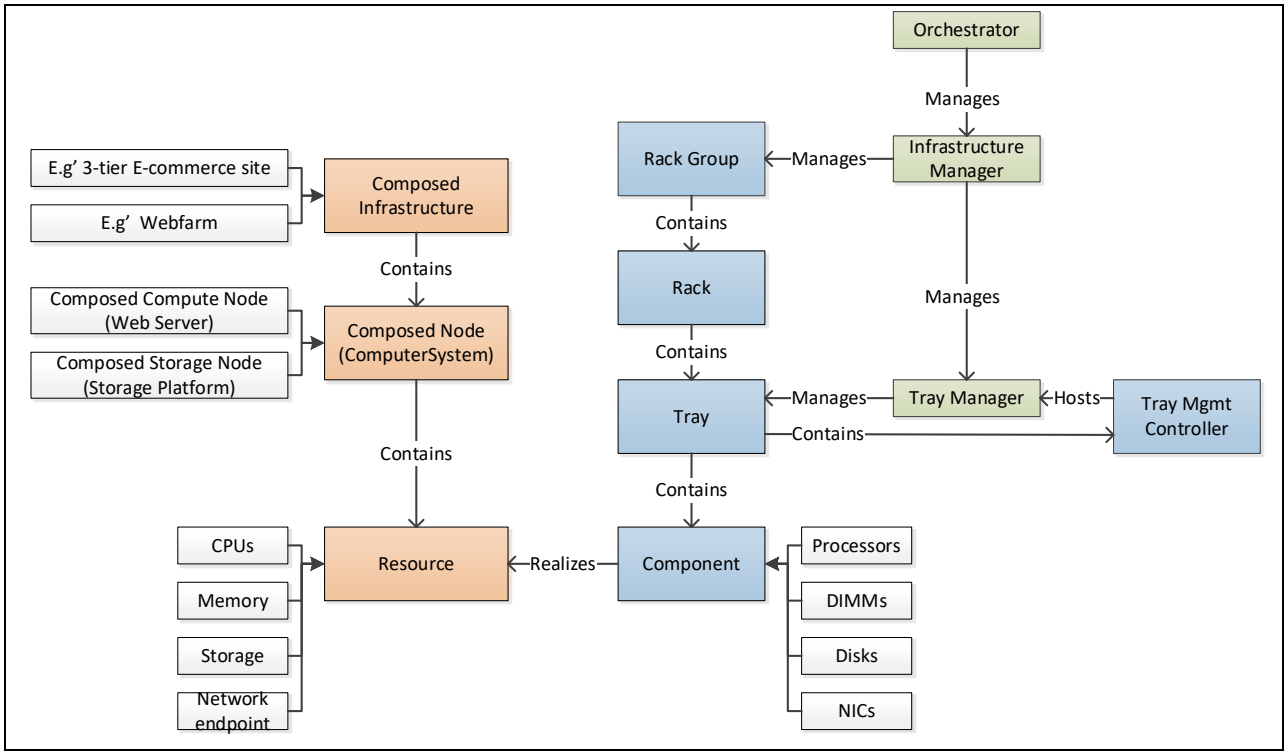
Table 2. Architecture terminology definitions

Term	Definition
Pod	A physical and/or virtual collection of Racks, Drawers, Modules, and Blades.
Rack	A physical element in a Pod that holds Pod resources.
Drawer	A physical element in a rack that acts as a container for Modules, Blades, or other system resources. Drawers provide the underlying compute resources or storage resources from which Composed Nodes are created.
Compute Drawer	A type of Drawer that holds compute resources for a Pod.
Storage Drawer	A type of Drawer that holds storage resources for a Pod.
Module	A physical element in a Drawer that is a grouping of compute or storage hardware resources that represent a field replaceable unit (FRU) or that can be reset together. (Note: The terms Module and Blade are used interchangeably because the physical elements can overlap in degenerate cases.)
Compute/Storage Module	A physical compilation of compute or storage hardware resources that represent a field replaceable unit (FRU) or that can be reset together.
Blade	A physical element in a Drawer that is a grouping of compute or storage resources. (Note: The terms Module and Blade are used interchangeably because the physical elements can overlap in degenerate cases.)
Compute Blade	A type of Blade server that provides compute resources within a Compute Drawer.
Storage Blade	A type of Blade server that provides additional storage resources within a Storage Drawer.
Network Agent	A logical agent that manages the RSD-compliant switch.
Node	A logical system element of the RSD architecture that contains resources (CPU components, memory components, or switches) that are attached to a computer or a network, and that can perform basic RSD functions.
Composed Node	A logical compilation of physical system resources, composed by the PODM. PODM creates Composed Nodes within the rack by communicating with the PSME to request the allocation of resources based on the user input.
Platform	A physical element of computer system architecture that includes a microprocessor, instruction set architecture design, microarchitecture design, logic design, and implementation that allows users to develop, run, and manage applications without adding the complexity of building and maintaining an infrastructure. Systems designed around the x86 microprocessor architecture are considered a Platform.
Rack Manager	A logical system element (firmware) that runs on the RMC.
In-Band vs. Out-Of-Band control	A protocol control system. In-band designs pass control data on the same connection as the main data (examples of In-band control include HTTP and SMTP). Out-of-band designs separate control data from the main data (an example of Out-of-band control is FTP).

1.5.4 Logical representation of Intel® RSD V2.2 Pod

Figure 2 illustrates a logical representation of the RSD Pod.

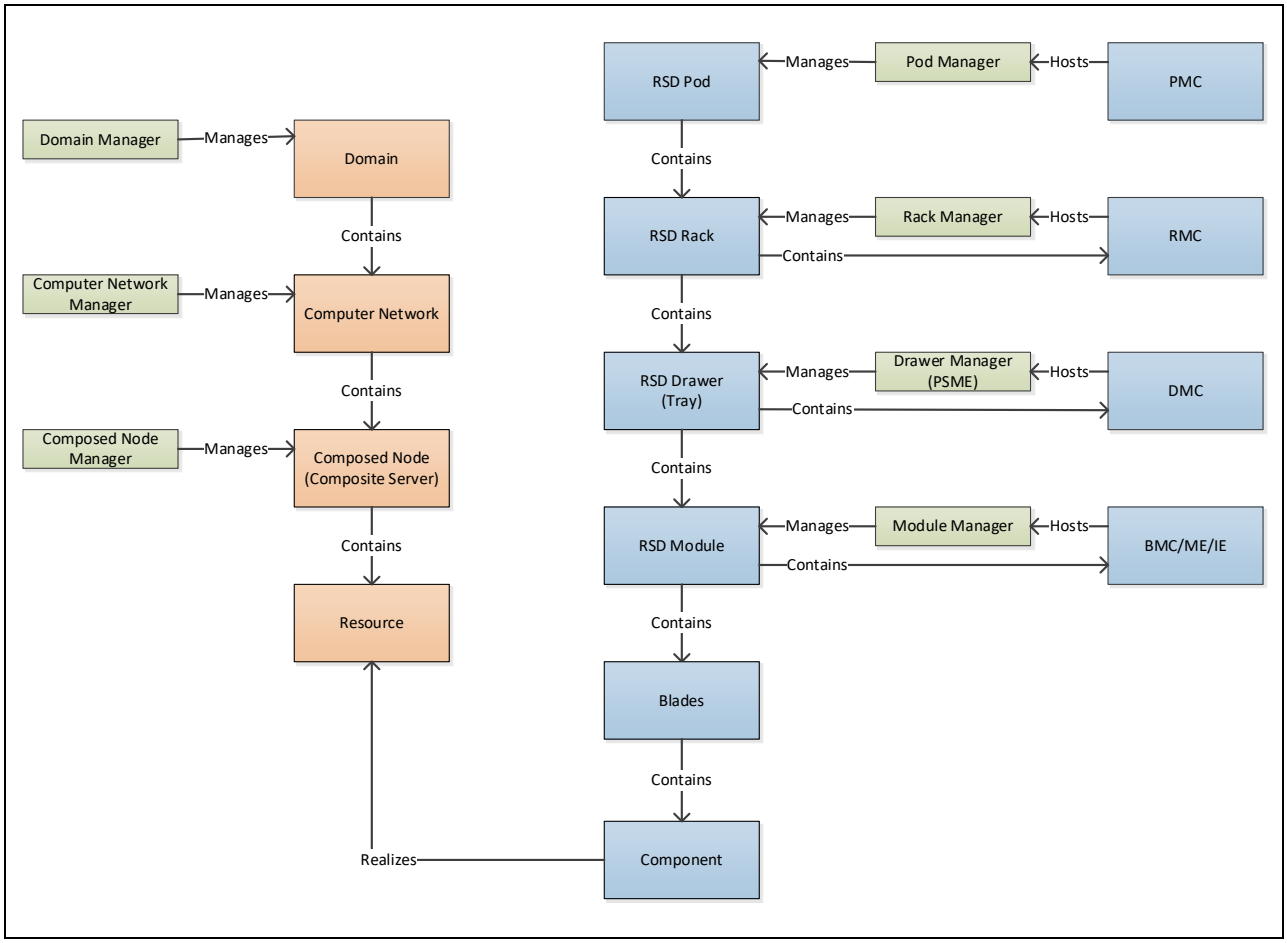
Figure 2 Logical View of the Intel® RSD V2.2 Pod



1.5.5 Management elements of Intel® RSD V2.2 Pod

Figure 3 illustrates the management elements of an RSD Pod.

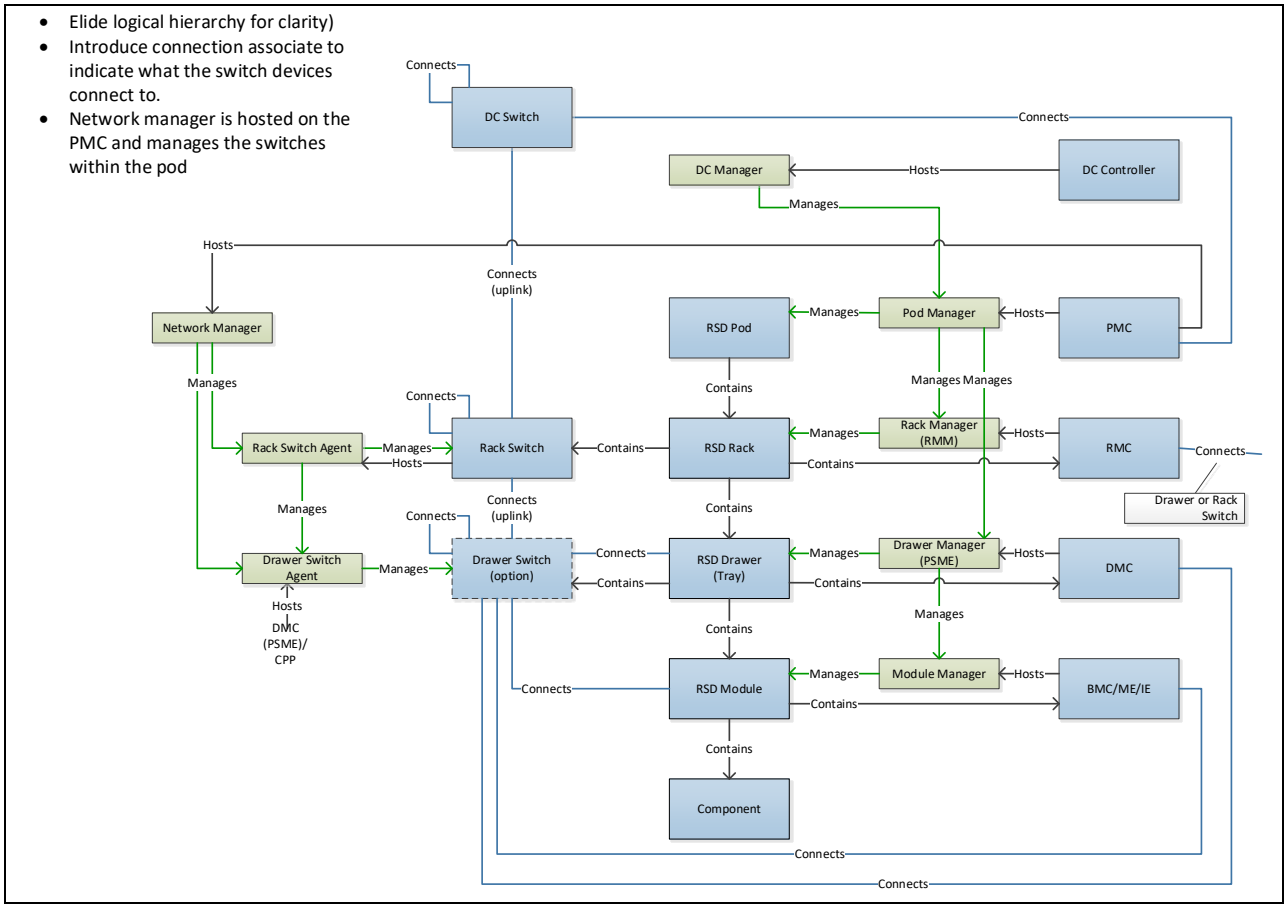
Figure 3. Management elements of a logical Intel® RSDv2.2 Pod



1.5.6 North Bound management hierachy of Intel® RSD v2.2 Pod

Figure 4 illustrates North Bound management hierarchy of an RSD Pod.

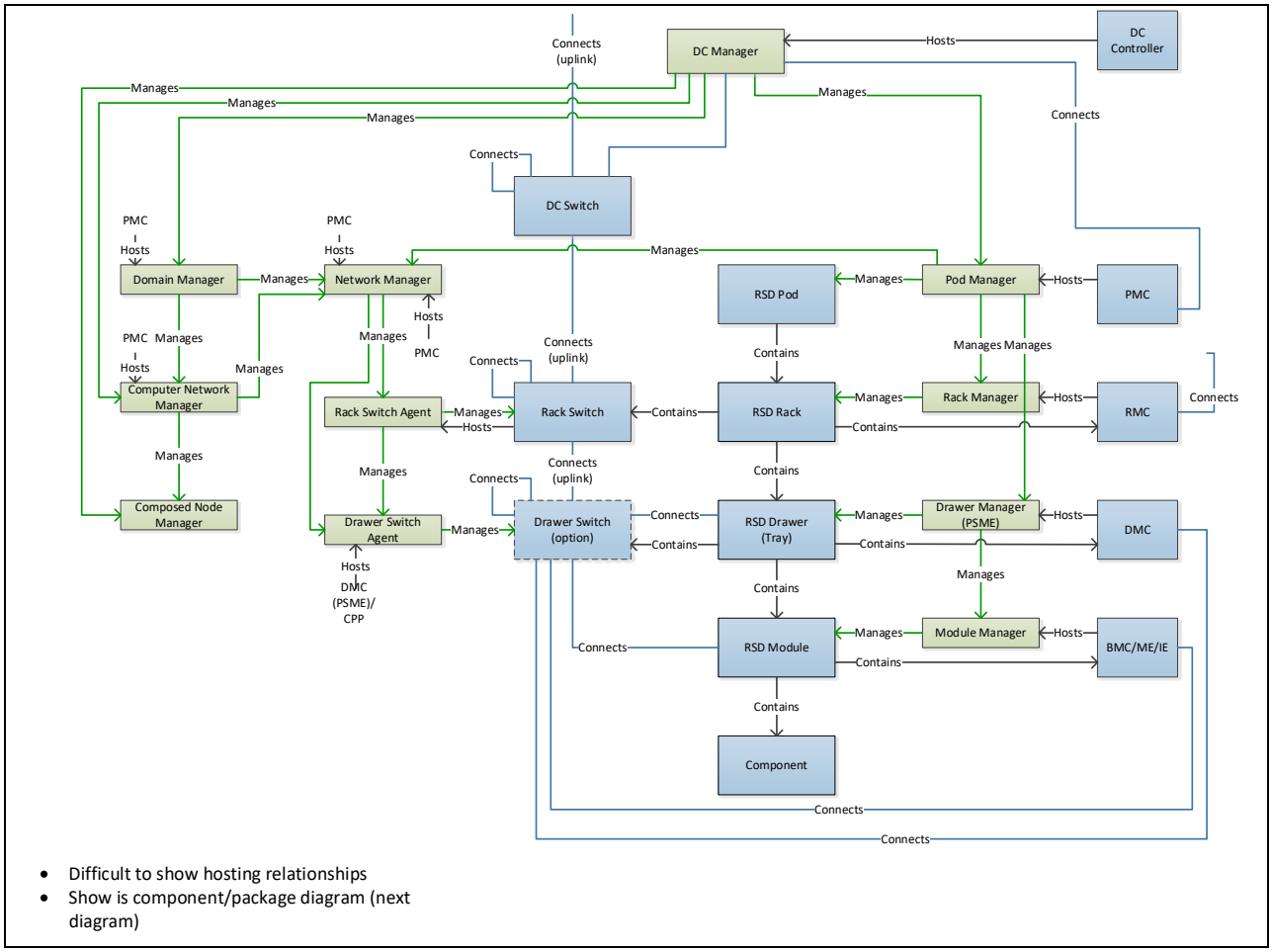
Figure 4. North Bound management hierarchy of Intel® RSD v2.2 Pod



1.5.7 Full management hierachy of Intel® RSD v2.2 Pod

Figure 5 illustrates the full management hierarchy of an RSD Pod.

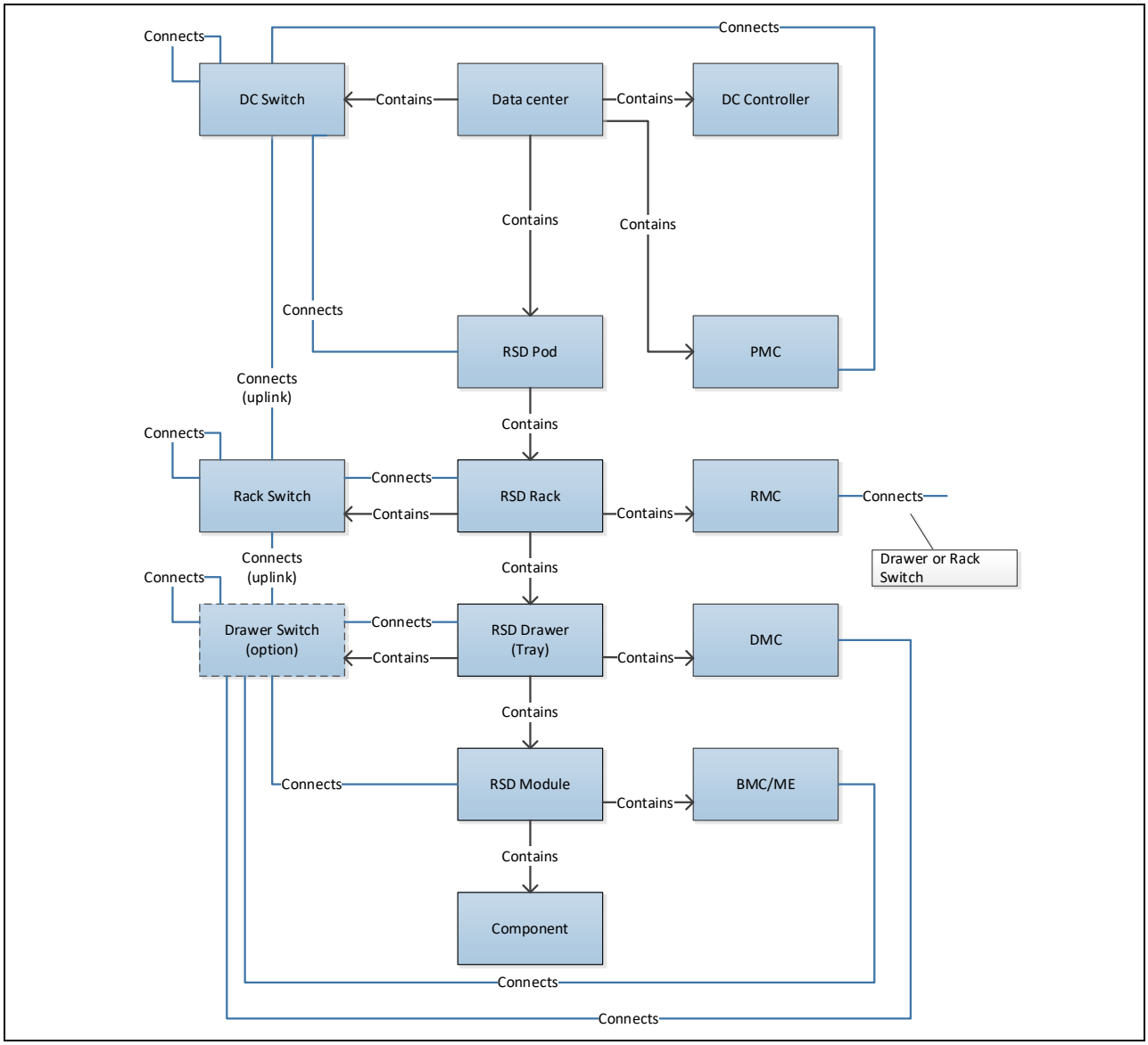
Figure 5. Full management hierarchy of Intel® RSD v2.2 Pod



1.5.8 Full physical containment hierachy of Intel® RSD v2.2 Pod

Figure 6 illustrates the full physical containment hierarchy of an RSD Pod.

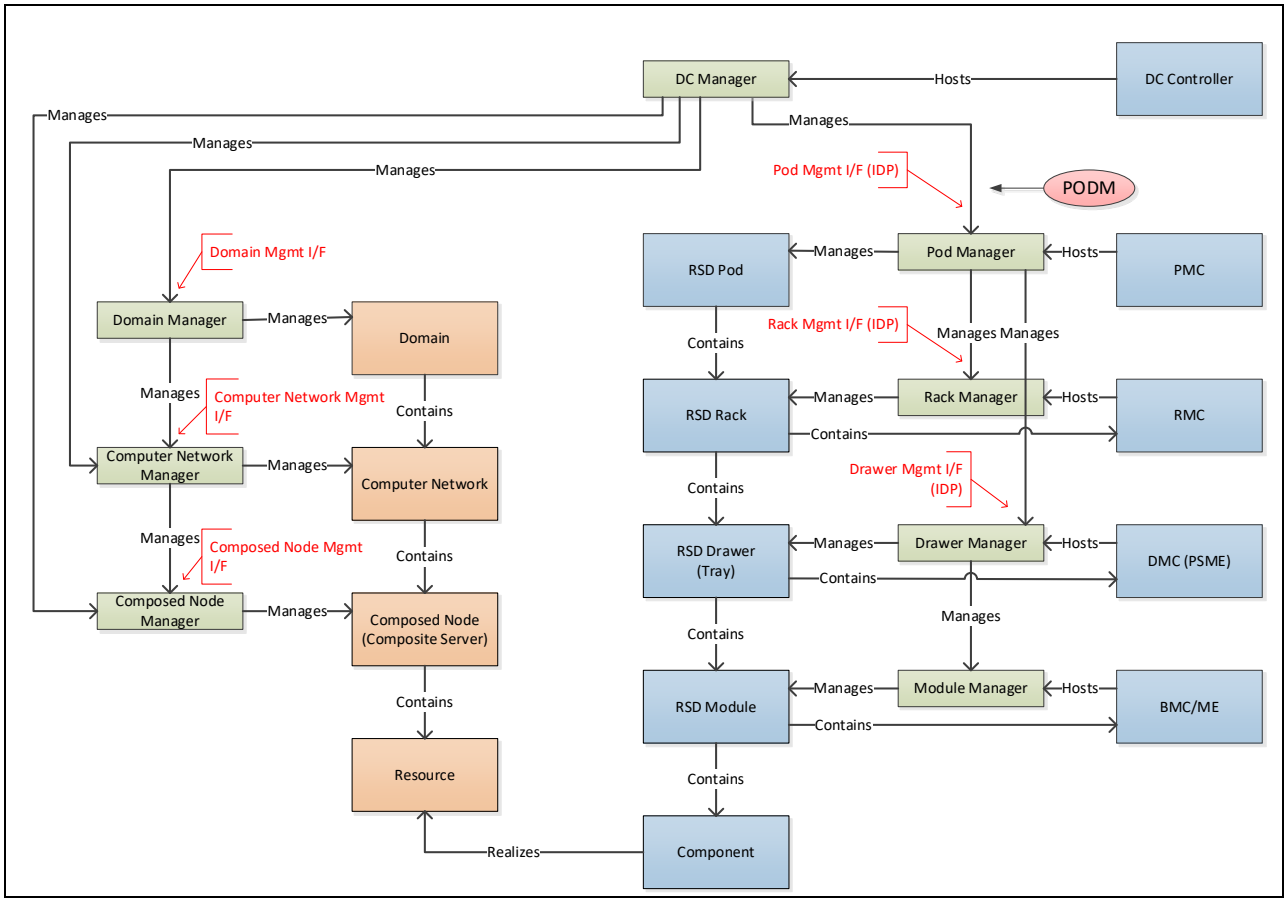
Figure 6. Full physical containment hierarchy of Intel® RSD v2.2 Pod



1.5.9 Interfaces of Intel® RSD v2.2 Pod

Figure 7 illustrates the interfaces of an RSD Pod.

Figure 7 Interfaces of Intel® RSD v2.2 Pod



1.5.10 Mapping Pod software to Pod architecture layers

Figure 8 illustrates the five physical layers for a typical Pod built on the Intel® RSD v2.2 architecture and how the software and firmware span different levels of the architecture.

The Pooled System Management Engine (PSME) is responsible for Drawer identification management, as well as supporting the PSME RSD API and communicating with the Baseboard Management Controller (BMC) to perform Compute/Storage Module-level management.

1.5.11 Intel® RSD container options

Even though Figure 3 through Figure 9 illustrates RSD architecture with container hierarchy, including an RSD Pod containing an RSD Rack, an RSD Rack containing an RSD Drawer, an RSD Drawer containing an RSD Module, an RSD Module containing an RSD Blade – the RSD Drawer, RSD Module and RSD Blade are optional components. For example, if an RSD Module implements PSME API, the RSD Pod manager can interface with the RSD Module without having an RSD Drawer.

Figure 8. Physical layers and Software Architecture in Intel® RSD with Physical Rack/RMM

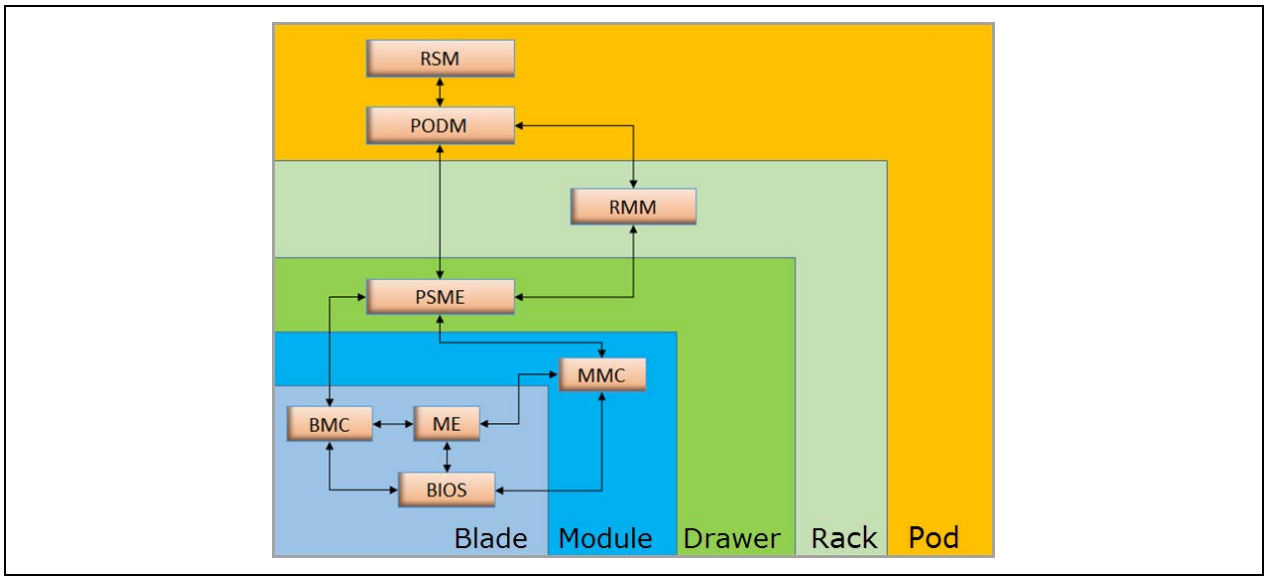
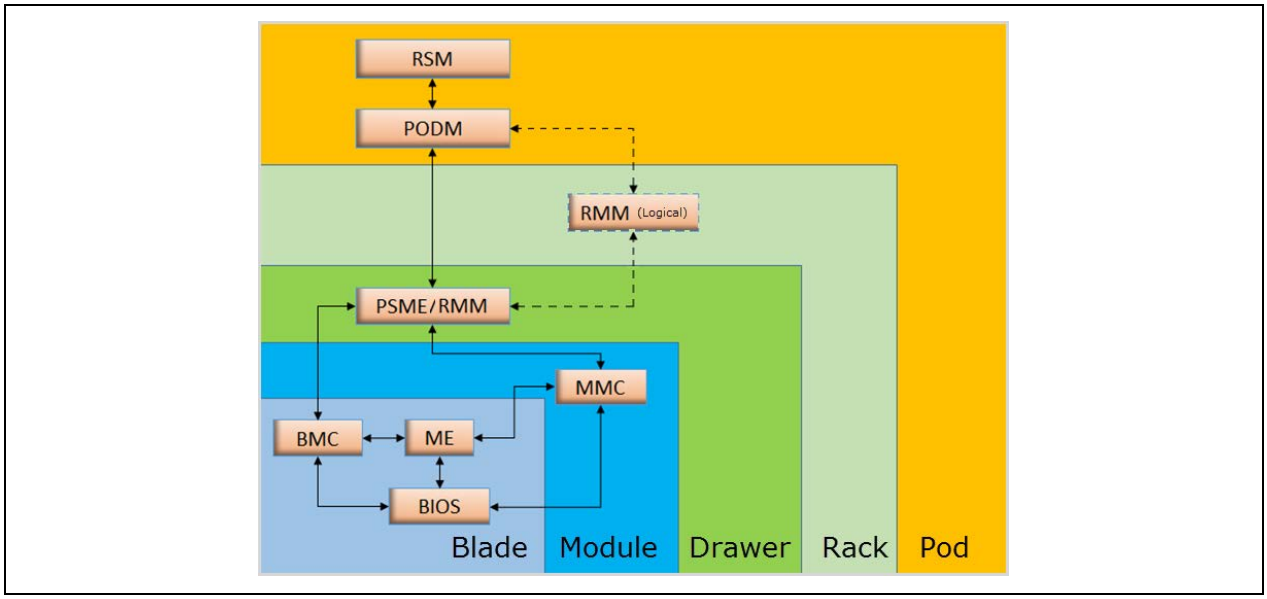


Figure 9 illustrates an alternative system configuration that does not include the separate instance of the RMM in the Rack. In this case, one of the PSME instances in the Pod can be used to provide the RMM services.

Figure 9 Physical layers and Software Architecture in Intel® RSD with Logical Rack/RMM



1.6 Intel® RSD platform hardware components

The Intel® RSD Platform consists of the following hardware components:

- RSD Pod
  - Where PODM (Pod Manager) resides
- RSD Rack (logical or physical)
- RSD Drawer
- RSD Module/Blade

- Pooled system management engine (PSME)
- RSD Compute/Storage Module
  - A Compute Module or a Storage Module
- Power supply (either independent per Drawer or shared between Drawers)
- Cooling (either independent per Drawer or shared between Drawers, using fans or other means)
- Pooled Node Controller (PNC)

1.7 Intel® RSD platform software components

The Intel® RSD Platform consists of the following software components:

- Pod Manager
- Rack Management Module (RMM) software
- PSME Software (optional if BMC provides Redfish interface for compute nodes)
- Network switch agent software
- BMC Firmware
- BIOS
- Pooled Node Controller (PNC) Firmware

1.8 Intel® RSD API

The RSD API information is outlined in the Intel® RSD API specifications, Table 3, Intel® RSD Reference Documents.

1.9 References to more information

To find more information about related aspects of Intel® RSD, refer to the documents listed in Table 3.

Note: The following documents are placeholders for final documents.

Table 3. Intel® RSD Reference Documents

Doc ID	Title	Location
336811	Intel® Rack Scale Design (RSD) Conformance and Software Reference Kit Getting Started Guide v2.2, Revision 001	*http://www.intel.com/intelRSD*
336814	Intel® Rack Scale Design Pod Manager (PDOM) Release Notes, Software v2.2, Revision 001	
336815	Intel® Rack Scale Design Pod Manager (PDOM) User Guide, Software v2.2, Revision 001	
336816	Intel® Rack Scale Design PSME Release Notes, Software v2.2, Revision 001	
336810	Intel® Rack Scale Design PSME User Guide, Software v2.2, Revision 001	
336855	Intel® Rack Scale Design PSME REST API Specification, Software v2.2, Revision 001	
336856	Intel® Rack Scale Design Storage Services API Specification, Software v2.2, Revision 001	
336857	Intel® Rack Scale Design Pod Manager REST API Specification, Software v2.2, Revision 001	
336858	Intel® Rack Scale Design Rack Management Module (RMM) API Specification, Software v2.2, Revision 001	
336859	Intel® Rack Scale Design Generic Assets Management Interface API Specification, Software v2.2, Revision 001	
336860	Intel® Rack Scale Design Firmware Extension Specification, Software v2.2, Revision 001	
336862	Intel® RSD v2.2 Solid State Drive (SSD) Technical Advisory	
RFC2119	Key words for use in RFCs to Indicate Requirement Levels, March 1997	*https://www.ietf.org/rfc/rfc2119.txt*

Doc ID	Title	Location
SDP0266	Scalable Platforms Management API Specification v1.1.0	*https://www.dmtf.org/sites/default/files/standards/documents/DSP0266_1.1.0.pdf*
DSP8010	Redfish Schema v2016.3	*https://www.dmtf.org/sites/default/files/standards/documents/DSP8010_2016.3.zip*

1.10 Notes and Symbol Convention

- Symbol and note convention are similar to typographical conventions used in CIMI specification.
- Notation used in JSON serialization description:
- Mandatory in italics indicate data types instead of literal Mandatory.
- Characters are appended to items to indicate cardinality:
  - "?" (0 or 1)
  - "*" (0 or more)
  - "+" (1 or more)
- • Vertical bars, "|", denote choice. For example, "a|b" means a choice between "a" and "b".
- • Parentheses, "(" and ")", are used to indicate the scope of the operators "?", "*", "+" and "|".
- • Ellipses (i.e., "...") indicate points of extensibility.

Note: The lack of ellipses does not mean no extensibility point exists; rather it is just not explicitly called out.

§

2 Intel® RSD Platform Requirements Summary

This section provides a summary of the RSD Platform design requirements. The requirements are listed in Table 4. The columns labeled Section, RSD Validation Criteria, and Compliance contain links to specific topics in this document. Note the RSD version column contains the initial version of the RSD when the specified feature is introduced.

Table 4. Intel® RSD platform architecture requirements summary

Section	RSD validation criteria description	Compliance	Type	RSD version
3	**Intel® RSD Platform** General Guidelines			
3.2	Generic Intel® RSD platform requirements			
3.2.1	Rack must have one or more logical Pooled System Management Engine software (PSME)	Required	Software	1.2
3.2.2	Shared or highly efficient power	Required	Hardware	1.2
3.2.3	Shared or highly efficient cooling	Recommended	Hardware	1.2
3.2.4	JBOD support	Optional	Hardware	1.2
3.2.5	Compute module with local boot drive	Optional	Hardware	1.2
3.2.6	At least one Intel® RSD compute module in Pod	Required	Hardware	1.2
3.2.7	Compute module serviceability independence	Required	Hardware	1.2
3.2.8	Ethernet-based fabric	Required	Hardware	1.2
3.2.9	At least one Ethernet switch in the Pod	Required	Hardware	1.2
3.2.10	Network switch support for network software agent	Required	Hardware	1.2
3.2.11	PODM support PNC	Recommended	Hardware	2.1
3.2.12	Platform support for FPGA	Recommended	Hardware	2.2
3.2.13	Hot-pluggable modules in Intel® RSD drawers	Required	Hardware	1.2
3.2.14	Backward-compatibility for Intel® RSD v2.1 PODM	Required	Software	2.1
3.2.15	Backward-compatibility for Intel® RSD v2.1 drawer	Required	Software	2.1
3.2.16	Intel® RSD v2.2 and Intel® RSD v2.1 coexistence support within a Rack	Required	Software	2.1
3.2.17	PODM-to-PSME communication channel protection	Required	Software	1.2
3.2.18	PODM-to-RMM communication channel protection	Required	Hardware	1.2
3.2.19	PSME-to-RMM communication channel protection	Required	Hardware	1.2
3.2.20	In-band re-configuration of the private network	Required	Hardware	2.2
3.2.21	User-maintained backup copy of data	Recommended	Software	2.1
3.3	Intel® RSD components location identification support			
3.3.1	Field replaceable units identification and location information	Required	Software	1.2
3.3.2	Connectivity identification	Required	Software	2.1
3.4	Intel® RSD fabric and network configuration			
3.4.1	OOB management network and In-band data network separation	Required	Hardware	1.2
3.4.2	Secure NTP access availability	Required	Software	2.1
3.4.3	Secure DHCP server availability if DHCP discovery is used	Recommended	Software	2.1
3.4.4	Secure DNS support	Recommended	Software	2.1
3.5	Intel® RSD platform configuration and provisioning			
3.5.1	Serial over LAN (SOL) or KVM support for compute modules	Required	Software	2.1
3.5.2	FW and SW updates signed and checked	Required	Software	2.2
3.5.3	PODM or admin control of updates	Recommended	Software	2.2
3.6	Intel® RSD platform security			
3.6.2	Composed node volatile memory clearing	Required	Software	2.1
3.6.3	User to archive data before decomposing a node	Recommended	Software	2.1

Section	RSD validation criteria description	Compliance	Type	RSD version
3.7	Intel® RSD power and cooling			
3.7.1	Power monitoring support	Required	Software	2.1
3.7.2	Power budgeting support	Recommended	Software	1.2
3.7.3	Cooling failure reporting	Required	Software	2.2
4	**Intel® RSD API**			
4.1	Intel® RSD API interface			
4.1.1	Intel® RSD API compliance	Required	Software	1.2
4.1.2	Intel® RSD API support for access control and secure communication channel	Required	Software	2.1
5	**Module Design Guidelines**			
5.1	Module reset, power, and performance			
5.1.1	Module power on/off support	Required	Hardware	2.1
5.1.2	Module reset support	Required	Hardware	2.1
5.1.3	Power monitoring support	Required	Hardware	2.1
5.1.4	Power budgeting support	Recommended	Software	1.2
5.2	Module features			
5.2.1	Expose TPM capabilities if TPM present	Required	Software	2.2
5.2.2	Expose SMBIOS information to PSME	Required	Software	2.2
5.2.3	Expose FPGA capabilities if FPGA is present	Required	Software	2.2
5.2.4	BIOS/firmware support for PNC if PNC supported	Required	Software	2.1
5.2.5	Minimum 10GbE NIC per module for data plane	Recommended	Hardware	2.1
5.3	Module firmware update			
5.3.1	Module in-band firmware update blocking	Required	Software	2.2
5.3.2	Firmware update authentication	Required	Software	2.2
5.3.3	Module configuration default support	Recommended	Software	2.2
5.4	Module configuration information			
5.4.1	Module processor information (out-of-band)	Required	Software	2.2
5.4.2	Module memory information (out-of-band)	Required	Software	2.2
5.4.3	Module storage information (out-of-band)	Recommended	Software	2.2
5.4.4	Compute module remote OS boot support	Required	Software	2.1
5.4.5	Compute module iPXE support	Recommended	Software	2.1
5.4.6	Compute module iSCSI support	Recommended	Software	2.1
5.4.7	Compute module OS boot from local storage	Recommended	Software	2.1
5.4.8	Module security support information	Recommended	Software	2.2
5.5	Reliability availability and serviceability (RAS) support			
5.5.1	Out-of-band health event support	Required	Software	2.2
5.5.2	Error persistence over reboot	Recommended	Software	2.2
5.5.3	Reporting health and performance information	Required	Software	2.2
6	**PCIe* Direct Attach Pooled I/O Design Guidelines**			
6.2	System topology and mapping			
6.2.1	Enumeration of components in the system must be deterministic and persistent across power or initialization cycles	Required	Software	2.1
6.2.2	PSME exclusive management link to PNC	Required	Hardware	2.1
6.2.3	Expose and enumerate PNC devices in a pooled system	Required	Software	2.1
6.2.4	Expose PSME mapping of management connections to PNCs	Required	Software	2.1
6.2.5	Assignment of primary PSME for PNC	Required	Software	2.1
6.2.6	Expose and enumerate PNC upstream ports	Required	Software	2.1
6.2.7	Expose and enumerate PNC downstream ports	Required	Software	2.1

Section	RSD validation criteria description	Compliance	Type	RSD version
6.2.8	Expose data path cross-connections between multiple PNC	Recommended	Software	2.1
6.2.9	Expose and enumerate device slots of the IO pooled system	Required	Software	2.1
6.2.10	Expose mapping of device slot connectivity to PNC downstream ports	Required	Software	2.1
6.2.11	Compute module to PNC upstream port connection ID mapping	Required	Software	2.1
6.2.12	Expose the connection presence of each upstream port	Optional	Software	2.1
6.3	I/O device discovery support			
6.3.1	Expose the presence of an I/O device	Required	Software	2.1
6.3.2	Discovery of device type and capability	Required	Software	2.1
6.3.3	PSME configuration of I/O device support if sharing of IO device supported	Required	Software	2.1
6.4	I/O device assignment to compute module			
6.4.1	Full assignment of a device PCIe function to a single compute node	Required	Software	2.1
6.4.2	Assignment of single PCIe function to multiple upstream ports	Optional	Software	2.1
6.4.3	Dynamic assignment of a device shall not affect other device connectivity	Required	Software	2.1
6.4.4	Dynamic release of a device shall not affect other device connectivity	Required	Software	2.1
6.4.5	Devices with data storage must secure data upon release	Recommended	Software	2.1
6.4.6	I/O resources must be in an unassigned state prior to assignment to a compute node	Required	Software	2.1
6.5	Adding or removing devices from the I/O pool			
6.5.1	Physical hot add support of devices to the IO pool	Required	Software	2.1
6.5.2	Managed removal of device from the I/O pool support	Required	Software	2.1
6.5.3	Surprise removal of a device from the I/O pool support	Required	Software	2.1
6.5.4	Surprise disconnect of the IO pool shall be supported	Required	Software	2.1
6.5.5	Notification of devices added or removed from the I/O pool	Required	Software	2.1
6.6	Error handling and telemetry			
6.6.1	Down port containment support for all PNC downstream ports	Required	Hardware	2.1
6.6.2	Fault and service indicators for I/O devices	Recommended	Hardware	2.1
6.6.3	PNC trap of PCIe error events detected on the PCIe link	Recommended	Software	2.1
6.6.4	Expose PNC, device and I/O pooled system telemetry	Recommended	Software	2.1
6.7	Pooled I/O system support			
6.7.1	Device serviceability while system powered on	Required	Hardware	2.1
6.7.2	Pooled system enclosure management support	Recommended	Software	2.1
6.7.3	AUX power to cable connector	Optional	Hardware	2.1
6.7.4	Exposing cable electrical parameters for cable signal drive support	Optional	Software	2.1
6.8	Compute module requirements for I/O pooled systems			
6.8.1	Independent PCIe* domain per compute module connection	Required	Software	2.1
6.8.2	Down port containment support for all connected ports	Required	Hardware	2.1
6.8.3	BIOS shall allocate memory space for all potential I/O devices	Required	Software	2.1
6.8.4	Compute module visibility of IO device controlled by the PSME	Required	Software	2.1
6.8.5	Compute module connection identification	Recommended	Software	2.1
6.8.6	Compute module managing the assigned I/O device	Optional	Software	2.1
6.8.7	Compute module managing the I/O pool system is not allowed	Required	Hardware	2.1
7	**PMSE Design Guidelines**			
7.2	PSME configuration management			
7.2.1	PSME API compliance	Required	Software	1.2

Section	RSD validation criteria description	Compliance	Type	RSD version
7.2.2	PSME authentication credential	Required	Software	2.1
7.2.3	PSME time sync mechanism	Required	Software	2.1
7.2.4	PSME telemetry requirements	Required	Software	2.2
7.2.5	Serial-over-LAN and KVM credential change with user changes	Recommended	Software	2.2
7.2.6	PSME support for power and thermal capping	Recommended	Software	2.2
7.3	PSME software update			
7.3.1	PSME remote software update	Required	Software	2.2
7.4	PSME reliability, availability and serviceability support			
7.4.1	Drawer event reporting	Required	Software	2.1
7.4.2	Drawer (PSME) hot add only when RMM is present and running	Recommended	Software	2.1
8	**RMM Design Guidelines**			
8.2	RMM reset (power on)			
8.2.1	RMM boot and PSME ID assignment	Required	Software	2.1
8.2.2	RMM assigns PSME ID if PSME not configured	Required	Software	2.1
8.2.3	PSME enters “PSME ID Not Configured” state	Required	Software	2.1
8.3	RMM general support			
8.3.1	RMM event handling	Required	Software	2.1
8.4	RMM power and cooling support			
8.4.1	Rack power monitoring support by RMM if shared power is used	Required	Software	2.1
8.4.2	Rack power budgeting support by RMM if shared power is used	Recommended	Software	2.1
9	**Pod Manager (PODM) Design Guidelines**			
9.2	PODM configuration management			
9.2.1	PODM powered independent of rack power	Required	Software	2.1
9.2.2	PODM REST API compliance	Required	Software	2.1
9.2.3	Secure communication channel for management network	Required	Hardware	1.2
9.2.4	PODM authentication certificate	Required	Software	2.1
9.2.5	PODM timestamp support	Required	Software	2.1
9.2.6	Only one active PODM per pod	Required	Software	2.1
9.2.7	PODM to allow addition of new drawers only when RMM is alive	Required	Software	2.1
10	**Network Switch Design Guidelines**			
10.1.1	Module-to-port mapping configuration file	Required	Software	2.1
10.1.2	Switch PSME support for base network services	Required	Software	2.1
10.1.3	Device discovery and switch configuration reporting	Required	Software	2.1
10.1.4	Switch functionality change event generation	Required	Software	2.1
11	**Telemetry**			
11.3.1	PSME API support for telemetry	Required	Software	2.2
11.3.2	Pod manager SB and NB API support for telemetry	Required	Software	2.2
11.3.3	Support for in-band telemetry collection	Recommended	Software	2.2
11.3.4	Support for correlation of IB and OOB telemetry data	Recommended	Software	2.2

§

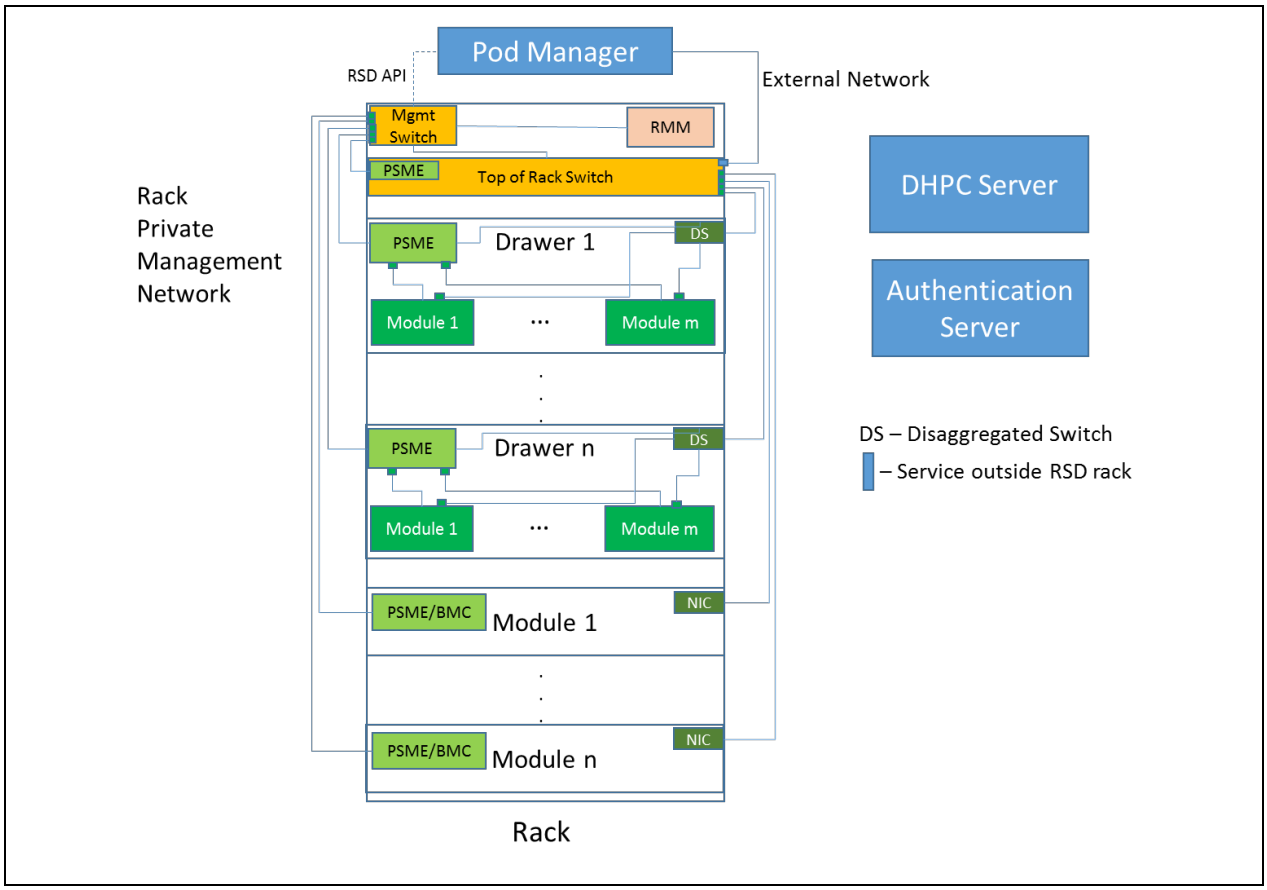
3 Intel® RSD Platform General Guidelines

This section describes the feature design guidelines for the Intel® RSD Platform. Subsequent sections describe the design guidelines for the RSD Platform subcomponents.

3.1 Intel® RSD platform power on flow

Figure 10 shows the private Rack-wide management network for the Intel® RSD API connections in a Pod.

Figure 10. Management plane in the Intel® RSD v2.2 Rack



When the power is applied to a Pod, all Racks in the Pod are powered on. Applying power to each Rack starts the RMM, PSME, and TORS resources in each Rack. Then the PODM can communicate with the RMM and PSMEs in the Racks to collect the hardware configuration information from the managed resources.

The Pod must step through a sequence of events to create one (or more) Composed Nodes. The event sequence is as follows:

1. Assumptions:
   - Pod has access to the data center NTP server
   - If host names are used for PODM, RMM, and PSME, then the data center provides access to the DHCP and DNS server
   - RMM and TORS need manual configuration
2. Prerequisites:

- If the DHCP and DNS are not available, provision PODM, RMM, and PSME with static IP addresses[1].
- Provision the RMM with PODM authentication credential and PODM host name or PODM IP address.
- Configure switches (such as TORS) to allow the RMM and PSME to connect to the data center OOB management network.
- The SSH may be used to provision the above steps.

3. Apply power to the Pod:
   - Power is applied to all the Racks in the Pod:
     - All the TORS are powered.
     - All RMC and Drawers are powered, and RMM firmware starts execution.
     - All the Drawers are powered, and PSME firmware starts execution.

Note: When power is applied to Drawers (in step 3), the sequence of power applied to the Modules, Blades, and PNC is implementation specific, as controlled by the PSME.

4. The RMM and PSME are assigned an IP address:
   - If the host name is used, then the IP address is obtained from the DHCP server. DNS maintains the latest host-name-to-IP address mapping.
   - If a static IP address is used, the DHCP server is not required.
5. The RMM and PODM create a TLS session with two-way authentication[2]. Although the hard requirement is that RMM authenticate PODM, RMM is acting as the TLS server.
6. PODM registers the RMM into its CMDB, with the status “RMM initialization started.”
7. PODM waits for the RMM to send a message indicating “RMM initialization completed.”
8. The RMM is ready to receive the queries from the PSME through the Rack-wide private network.
9. The RMM discovers the power supply and the fans, and starts managing them.
10. The RMM sends an “RMM initialization completed” message to PODM.
11. PSME gets the IP address from DHCP, Rack ID, PODM IP address, and PODM credentials from the RMM using the Rack-wide private network.
12. PSME and PODM create a TLS session with two-way authentication, and establish a secure communication channel with PODM.
13. PODM registers the PSME into its CMDB, with status “PSME initialization started.”
14. PSME discovers the Drawer inventory (such as Modules and Blades) by communicating with the MMC and BMC.
    - This step can happen in parallel while the PSME is establishing communication with RMM and PODM.
15. SSDP discovery complete denotes PSME initialization stage complete.
16. PODM requests a “Drawer inventory” from each PSME.
17. Repeat steps 11 to 16 for each PSME in the Rack.
18. At this point, the PODM is ready to create and assign Composed Nodes.

Note: Rack private management network could be physically separated or logically separated. This document uses physical separation as a base design to describe the requirements.

---

[1] Notice the PODM is on an in-band managment network, whereas RMM and PSME are on the OOB managment network. How IP addresses get allocated in these different networks is data center specific.

[2] Although the hard security requirement is for RMM/PSME to authenticate PODM, not the other way around, because RMM/PSME is the server for the TLS connection, it is necessary to do two-way authentication. For cases where RMM/PSME authentication is not required i.e., a datacenter does not want to incur the cost of provisioning certificates for all RMMs/PSMEs), then PODM could just accept a self-signed certificate from RMM/PSME.

3.2 Generic Intel® RSD platform requirements

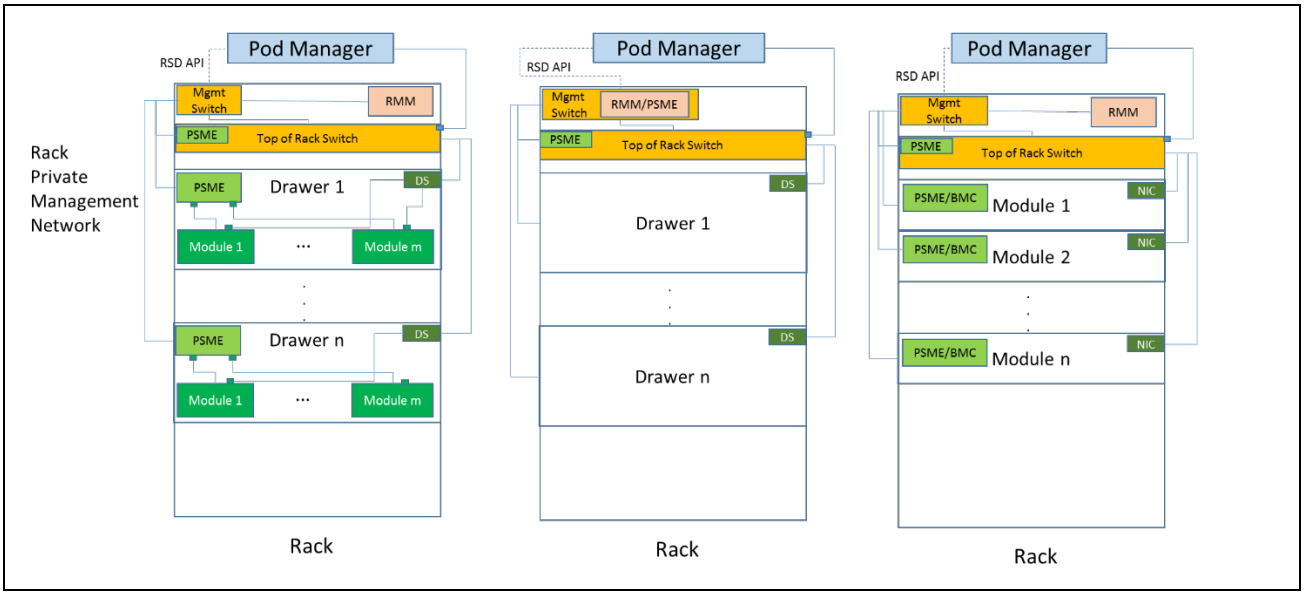
The RSD Platform meets the following generic requirements as stipulated (optional, recommended, or required) in each section.

3.2.1 Rack must have one or more logical Pooled System Management Engine software (PSME)

Required:

Figure 11 shows examples of RSD architecture implementations. In one implementation, the rack consists of drawers with PSME; the PSME interfaces with the modules and it has a separate RMM module. In another implementation the RMM and PSME reside on the same hardware, and another implementation consists of modules with BMC that exposes the PSME API. This example is meant to show all possible implementations, but the key point is that a logical RSD rack must have one of more PSMEs.

Figure 11. PSME in a Rack



3.2.2 Shared or highly efficient power supply

Required:

For optimal TCO, Compute/Storage Modules must support cost-effective, efficient, and manageable shared power. The solution is achieved by either, 1) sharing power across two or more Modules, or 2) having a > 90% efficient (delivered power to Module/input AC power) power configuration.

3.2.3 Shared or highly efficient cooling

Recommended:

For optimal TCO, Compute/Storage Modules are recommended to support cost-effective and manageable shared cooling. The Intel® RSD Platform is recommended to support shared cooling across two or more Modules. One option for shared cooling is to use a fan larger than 2U (3.5 inches) in diameter. If the system does not use fans, another (more efficient) cooling option is to implement shared liquid cooling.

If shared power is used, then the shared cooling solution is recommended.

3.2.4 JBOD support

Optional:

JBOD (Just a Bunch of Disks) is a collection of storage devices in a chassis for easy serviceability. The JBOD is generally connected through a cable (such as SATA, PCIe, etc.) between a Compute Module and a Storage Module.

3.2.5 Compute module with local boot drive

Optional:

An RSD Compute Module has iSCSI or iPXE support; this makes the boot or storage services available through the network. Compute Modules could use local storage such as M.2 drive, SSD or HDD for boot or for delta-file storage for better performance.

3.2.6 At least one Intel® RSD compute module in Pod

Required:

To compose systems, compute, and workloads, the RSD Pod must have at least one Compute Drawer with at least one Compute/Storage Module. It is possible that some racks support only storage and some support only compute, but a Pod must contain at least one compute module.

3.2.7 Compute module serviceability independence

Required:

To keep data centers always up and running and achieve hyperscale-like agility, Compute/Storage Modules must support modular CPU and memory resources that can be serviced or upgraded independent of other modules. The RSD Modules must be hot-pluggable into the RSD Drawer without requiring the Drawer to be powered down to provide high-RAS for RSD platforms.

3.2.8 Ethernet-based fabric for management and external network connectivity

Required:

The RSD Platform must support Ethernet-based fabric for management and external network connectivity. Notice that an RSD platform will support access to (at least) two isolated networks, one for normal in-band traffic and one for OOB management.

3.2.9 At least one Ethernet switch in the Pod

Required:

The RSD Pod must have at least one Ethernet switch component that connects the Pod to the external network.

3.2.10 Network switch support for network software agent

Required:

Network switch components (such as TORS) must support running a network software agent to configure network switches. The required switch management capabilities described in the PODM and PSME API specifications must be implemented.

3.2.11 PODM support PNC capabilites

Recommended:

The PNC provides a pooled storage capability that is connected to Compute/Storage Modules through the PCIe* fabric. The PNC module contains a PSME that enables PODM to assign drives to the compute modules. If the PNC is supported in the rack, then the PODM must support PNC PSME API compatibility.

3.2.12 Platform support for FPGA

Recommended:

It is strongly recommended the platform is capable of supporting an FPGA in the platform. The FPGA is a key performance differentiator in machine learning and big data workload processing. If the FPGA is present, PODM and PSME must discover these capabilities as described in the RSD API specifications.

3.2.13 Hot-pluggable modules in Intel® RSD drawers

Required:

In a hyperscale datacenter where uptime is required at all times, services can't be disrupted by adding or removing modules. The RSD Modules must be hot-pluggable into the RSD Drawer without requiring the Drawer to be powered down to provide high-RAS for RSD platforms.

3.2.14 Backward-compatibility for Intel® RSD v2.1 PODM

Required:

To achieve optimal TCO and improve overall data center operation, as well as ease the adoption of newer RSD systems into existing data centers, the RSD v2.1 PODM must support Intel® RSD v2.1 Racks and RSD v1.2 Racks.

3.2.15 Backward-compatibility for Intel® RSD v2.1 drawer

Required:

In order to achieve optimal TCO and improve overall data center operation, as well as ease the adoption of newer Intel® RSD systems into existing data centers and maintain compatibility while updates are happening, a PODM needs to be able to manage both a PSME v1.2 and v2.1 existing in the same rack.

If Intel® RSD v1.2 and v2.1 drawers are mechanically compatible, then the user should be allowed to interchange drawers. API backwards compatibility for N-1 major versions is required (for example v1.2 and v2.1)

3.2.16 Intel® RSD v2.2 and Intel® RSD v2.1 coexistence support within a Rack

Required:

An Intel® RSD Rack must support Platforms with Intel® RSD v2.2 and Intel® RSD v2.1 or Intel® RSD v1.2. The PODM must be Intel® RSD v2.2 compatible, but the PSME and RMM could be either Intel® RSD v1.2 or Intel® RSD v2.1 conformant, as indicated in the example listed in Table 5.

Table 5. Intel® RSD v2.1 component versions and platform support matrix

PODM version	PSME version	Intel® Xeon® Processor E5	Intel® Xeon® Processor Scalable family
1.2	1.2	Supported	Not Supported
2.1	1.2	Supported	Not Supported

PODM version	PSME version	Intel® Xeon® Processor E5	Intel® Xeon® Processor Scalable family
2.1	2.1	Supported	Supported[3]
2.2	1.2	Supported	Not Supported
	2.1	Supported	Supported[4]
	2.2	Supported	Supported

3.2.17 PODM-to-PSME communication channel protection

Required:

To withstand network attacks, PODM and PSME communications must be secured. To provide this security, the communication channel between PODM and PSME must be protected by the use of TLS.

3.2.18 PODM-to-RMM communication channel protection

Required:

To withstand network attacks, PODM and RMM communication must be secured. To provide this security, communication between PODM and RMM must be protected by using a private network or an OOB management network.

3.2.19 PSME-to-RMM communication channel protection

Required:

In order to withstand network attacks, PSME and RMM communication must be secured. To provide this security, PSME and RMM communication is protected by using a rack private network that is not reachable from outside the rack.

3.2.20 In-band re-configuration of the private network

Required:

To withstand network attacks and have a more secure design, the in-band software (application or workloads) on the Compute Module must not have access to the OOB management interconnect (i.e., rack private network or OOB management network).

[3] *IPMI Firmware Extensions commands may differ between Intel® Xeon® Processor E5 and Intel® Xeon® Processor Scalable family*
[4] *IPMI Firmware Extensions commands may differ between Intel® Xeon® Processor E5 and Intel® Xeon® Processor Scalable family*

3.2.21 User-maintained backup copy of data

Recommended:

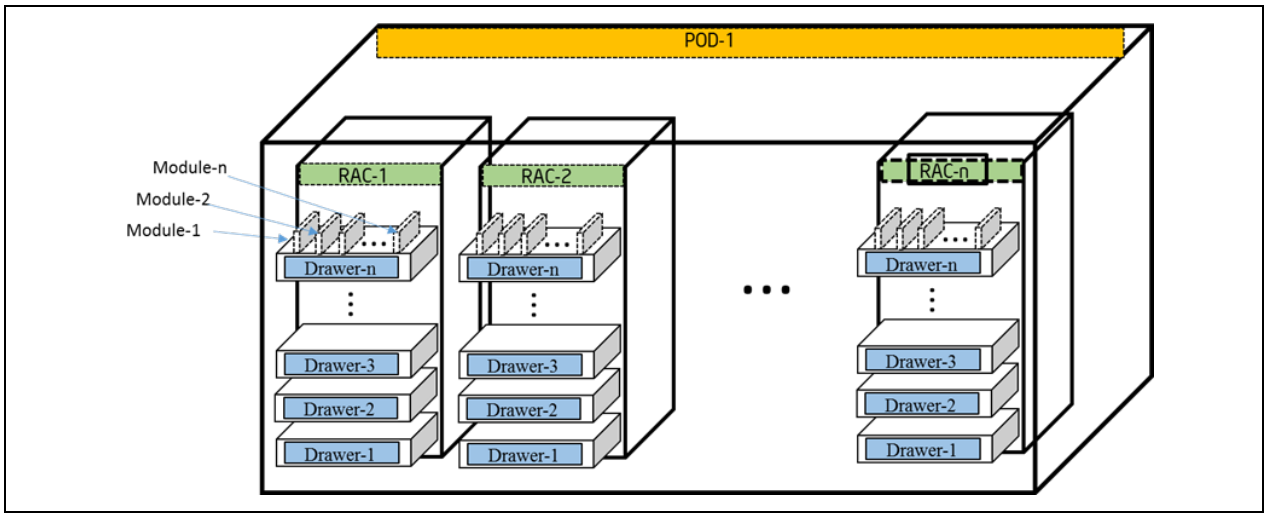
Before a user returns an unneeded Compute/Storage Module resource to PODM for re-composition, the user is responsible for backing up any data that is stored in the local memory on the Module. Once PODM re-composes the resources, the data stored on a Compute/Storage Module by the previous user will be cleared. By default, the data must be secure erased. It is strongly recommended that NVMe drives with secure erase capability are used for storage.

3.3 Intel® RSD components location identification support

To improve overall datacenter operations, a key attribute of Intel® RSD management is location-aware discovery. A data center manager or maintenance technician should be able to identify the physical location of Intel® RSD hardware components so they can be serviced. Figure 12 illustrates an example Pod and Intel® RSD hardware components across multiple Racks.

Refer to the Intel® RSD API specification for data format details.

Figure 12. Intel® RSD v2.2 Component Location Identification



3.3.1 Field replaceable units identification and location information

Required:

To help a service representative to locate and identify the Field Replaceable Units (FRU), all Intel® RSD components that are reported as FRUs must provide a unique identification number and location of the FRU. Refer to the Intel® RSD API specification Table 3 for the detailed format of component ID and location ID for supported components. The SMBIOS records generally provide the location of the component within the module. The Intel® RSD API must provide the location of the module within the chassis and the location of the chassis within the rack. In some cases the system admin could assign location id information chassis such as Rack Chassis, if supported it must follow the Intel® RSD API specifications.

3.3.2 Connectivity identification

Required:

The PODM must be able to understand the hardware connectivity topology of its components to provide better manageability and serviceability features. For example, compute node to storage node port/drive connectivity information helps to highlight the error path in case of compute node storage access failure. This must be implemented as described in the Intel® RSD API specifications, Table 3.

3.4 Intel® RSD fabric and network configuration

3.4.1 OOB management network and In-band data network separation

Required

For a more secure design, the Intel® RSD Platform must support access to two separate networks. One for OOB management access and one for in-band data access for host.

3.4.2 Secure NTP access availability

Required:

To improve overall data center operations, Intel® RSD event logs must be time stamped to help determination of event generation sequence. Intel® RSD system components (such as PODM, RMM, and PSME) must log events and errors. Intel® RSD components must synchronize time periodically from secure Network Time Protocol (NTP). This helps put together an overall Pod-wide event list and helps reduce the amount of time required to identify root cause issues. If only PODM has direct access, then the RMM and PSME must be able to get the time from PODM.

3.4.3 Secure DHCP server availability if DHCP discovery is used

Recommended:

If dynamic addressing is used with the PODM, RMM, and PSME using DHCP, then it is recommended that the data center implement a secure DHCP server to facilitate IP address assignment for the PODM, RMM, and PSME. If static IP addresses are assigned to the PODM, RMM, and PSME, then a DHCP server is not needed for Pod management. Another option is to use SSDP to discover and assign IP address.

3.4.4 Secure DNS support

Recommended:

If host names are assigned to the PODM, RMM, and PSME, then it is recommended that the data center implement a secure DNS service to facilitate collecting the IP addresses for the PODM, RMM, and PSME.

3.5 Intel® RSD platform configuration and provisioning

3.5.1 Serial over LAN (SOL) or KVM support for compute modules

Required:

To troubleshoot and debug issues remotely for improved overall datacenter operations, Compute Modules must provide serial console redirection support over LAN or KVM. Generally, this service is supported through a dedicated Baseboard Management Controller (BMC) or through a shared BMC from compute node.

3.5.2 FW and SW updates signed and checked

Required:

For a more secure design and to prevent tampering of the firmware, all FW/SW updates must be signed, and the integrity of the updates must be checked before they are used by individual components.

3.5.3 PODM or admin control of updates

Recommended:

For a more secure design and to prevent unauthorized access and tampering, it is recommended that a request for FW/SW updates come only from PODM or from an admin logged onto the hosting Platform who has the appropriate access rights.

3.6 Intel® RSD platform security

Section 3.6.1.x contains the **vision** of the RSD platform security architecture. Not all components specified in this section are available in the current RSD reference. Refer to Intel® RSD API specs for security features implemented in each RSD revision reference implementation.

3.6.1 Intel® RSD platform security architecture overview

Security work for the Intel® RSD Platform is driven by the following guiding principles:

- An RSD Composed Node provides at least the same level of security as an equivalent Intel® Architecture (IA) standalone server.
- An RSD cluster (made up of a set of Composed Nodes) is at least as secure as an equivalent cluster built with standalone IA servers.

Composed Nodes within the Intel® RSD are different from standalone IA server Modules in that Composed Nodes are “assembled” (composed) at the data center through the use of disaggregated, and in some cases, shared components. Specifically, Intel® RSD systems use management software at the Pod, Rack, and Drawer level to manage the inventory of components that are used to compose server Modules, to enable server Modules once they are composed, and to provide telemetry at different levels.

To follow the guiding principles above, Intel® RSD Platforms must support the following security objectives:

- Maintain integrity and availability of the Intel® RSD Platform
- Maintain isolation between (workloads running on) Composed Nodes in the presence of shared components

3.6.1.1 Maintain integrity and avalability of the Intel® RSD platform

In order to achieve this objective, Intel® RSD must provide the means for ensuring the following:

- Installation and update time integrity protection of all RSD management SW and FW
- Run time protection of the Intel® RSD management infrastructure
- Support for datacenter administrator separation of duty
- Protection against permanent denial of service (PDoS) as a result of a cyber-attack

3.6.1.1.1 Installation and update integrity protection

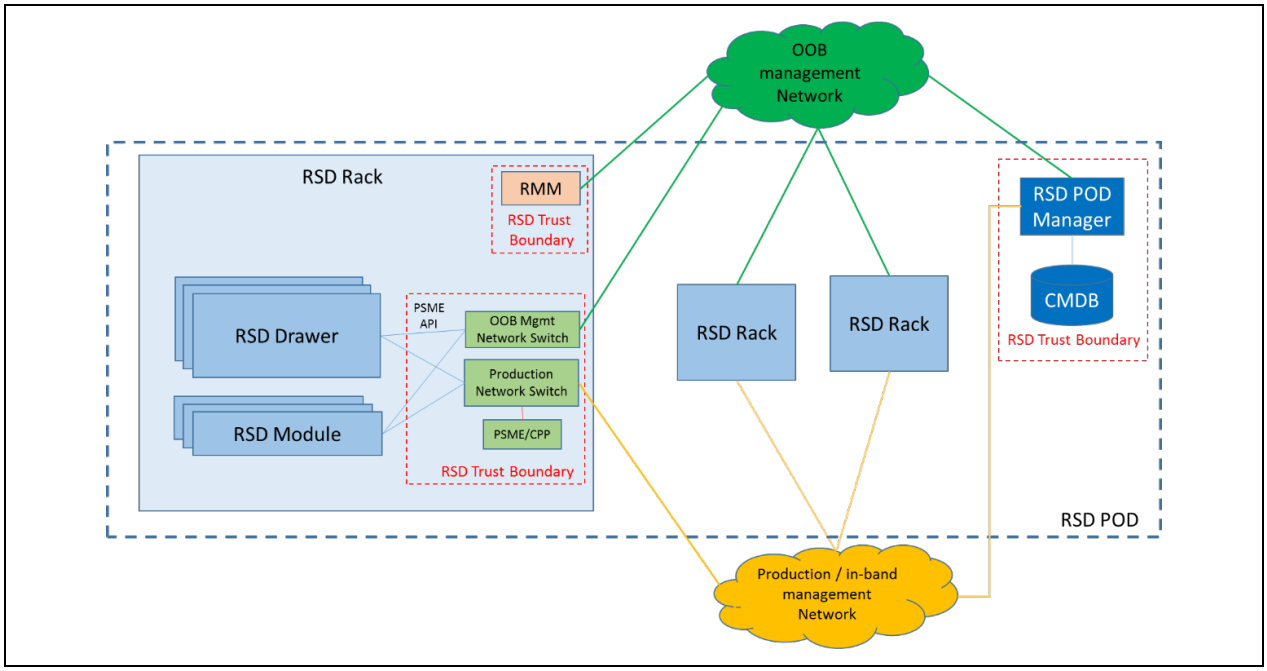
This requirement means that every firmware and software element inside the Intel® RSD trust boundary must be authenticated at installation and update time. Authentication in this context means that there must be a way to verify that the SW/FW being installed has not been changed in any way from what was originally delivered by the author.

Typical mechanisms for authenticating SW/FW involve delivering it with a cryptographic signature that can be verified at installation time. This signature includes, among other things, a cryptographic hash of the code being delivered such that a verifier can use the same hash algorithm to ensure that the code has not been altered in any way.

3.6.1.1.2 Run time protection of Intel® RSD management infrasructure

This requirement refers to ensuring that operation of the Intel® RSD management infrastructure is protected from entities that are outside the Intel® RSD trust boundary as outlined in Figure 13.

Figure 13. Intel® RSD Trust Boundary



The Intel® RSD management infrastructure must be protected from entities outside the Intel® RSD trust boundary including any management entities connected to the OOB management network, including other Intel® RSD Pods, other datacenter management servers, networking gear, etc., any software running on composed nodes or storage bricks that are made available through the Intel® RSD management infrastructure, and any other external entities that may be connected to other datacenter networks.

The main tools for achieving run time protection of the Intel® RSD management infrastructure are isolation and access control.

In terms of isolation, we must ensure that management logical functions either run on dedicated hardware, or that the hardware and system software hosting Intel® RSD management logical functions ensure this isolation from whatever other functions they may be hosting. For example, if a single server is hosting both the Pod management function and an unrelated datacenter switch management function, the system software in that server must ensure that the switch management function cannot be used to compromise the Pod management function. In this particular case, the Pod management function and the system software on that host would be inside the Intel® RSD trust boundary, but the switch management function would be outside. This must be enforced by the system software in that host.

Intel® RSD shared resources (i.e., a network switch) must also enforce isolation between their different users. In the case of the switch, it must enforce isolation between the OOB management network and the other networks in the datacenter. Enforcing this isolation means that SW running on composed nodes will have access to the orange networks, but not to the green network. This is critical for providing an environment in which OOB management functions can be isolated from the rest of the datacenter and addresses the requirement for protecting the Intel® RSD management infrastructure from elements outside the Intel® RSD trust boundary that have access to production in-band data networks, including SW running on composed nodes.

To support early initialization and security credential provisioning as well protected communication between elements inside the rack that do not support adequate access control (and thus should not be visible outside the rack), Intel® RSD racks must support a private and self-contained network (i.e., not reliant on external services like DHCP, DNS, etc.) This private network is not accessible outside the rack.

Because the different elements inside the Intel® RSD trust boundary don't reside on the same rack but must communicate with each other using an OOB management network that is outside the RSD trust boundary, isolation is not enough to guarantee run time protection. In order to allow protected communication between the different elements inside the Intel® RSD trust boundary, it is necessary to also use access control.

For Intel® RSD, access control means protecting all APIs provided by Intel® RSD logical management functions, as well as protecting login access to all hardware hosting those functions.

Access Control for Intel® RSD APIs

Pod Manager:

Pod Manager APIs are REST-based and must be protected using HTTPS. Authorization and authentication of users for these APIs is datacenter specific.

RMM:

The Rack Management Module handles rack level infrastructure functions like power and cooling, as well as provisioning of information (including security credentials) required for early rack initialization.

RMM provides a set of REST-based APIs that allow a Pod Manager to interact with it. These APIs are protected by HTTPS. Access control is such that only the controlling Pod manager can access these APIs. In other words, Pod Manager must authenticate itself to RMM. If this authentication is successful, a secure communication channel (e.g. TLS based) is established between Pod Manager and RMM.

Pod manager authentication relies on a credential that is provisioned to RMM at rack deployment time. This credential will be distributed by RMM to PSMEs in the rack during rack early initialization, or when a new PSME is powered on in an existing rack, using the rack private network.

PSME:

Like RMM, the PSME provides a set of REST-based APIs that allow the Pod manager to interact with it. These APIs are protected by HTTPS. Access control is such that only the controlling Pod manager can access these APIs. In other words, the Pod Manager must authenticate itself to the PSME. If this authentication is successful, a secure communication channel (e.g. TLS based) is established between the Pod Manager and PSME.

The credential used for the Pod manager authentication is provisioned by the RMM during early rack initialization, or when a new drawer is first powered on in a rack, using the rack private network.

Role-based Authorization

Optional for Intel® RSD v2.2:

Intel® RSD Security architecture allows for role-based authorization. The supported roles are Intel® RSD Viewer, Intel® RSD Admin, Intel® RSD Network Admin, and Intel® RSD Global Admin.

Pod manager will enforce access control to Intel® RSD functionality by external (to RSD) datacenter management infrastructure based on the roles described in Table 6.

Table 6. Intel® RSD admin roles

Role	Access mechanism	Authorization mechanism	Restrictions
RSD Viewer	HTTPS	DC specific	HTTP GET only (read only access), no ability to change state of any RSD element
RSD Admin	HTTPS	DC Specific (e.g. OAuth)	Can access most Pod Manager APIs and effect changes. Cannot access switch configuration and management APIs?
RSD Network Admin	HTTPS	DC specific (e.g. OAuth)	Can only access switch configuration and management APIs.
RSD Global Admin	HTTPS	DC Specific (e.g. OAuth)	Can access all PodM APIs.

3.6.1.1.3 Access control for hosts inside the Intel® RSD trust boundary

As previously mentioned, The Intel® RSD management logical functions are hosted by different physical hosts. RSD does not dictate a one-to-one relationship between an Intel® RSD management logical function and its physical host. Multiple functions could be hosted by the same physical host.

It may be possible to interact with the Intel® RSD by logging into physical hosts (through a local console or SSH) of the Intel® RSD logical management functions. Login access to these hosts should be strictly controlled as much as possible, it should be restricted to a system admin of the physical host. Although it is not recommended from a security perspective, a physical host may host more than just the Intel® RSD logical management functionality[5]. In this case responsibility for ensuring that the non- Intel® RSD functionality cannot interfere or compromise operation of RSD functionality falls entirely on system software and configuration of that host. Intel® RSD logical management functions has no way of protecting themselves from privileged level software or privileged users (e.g., root) in the physical host.

For the purpose of this discussion, we will assume that Pod manager, RMM, and MMC/BMC are hosted by their own physical hosts and the PSME and switch PSME (in a given drawer) are hosted by the same physical host.

Table 7. Login Access to Intel® RSD Management Hosts

Logical management function	Physical host	Users (login)	Default credentials
Pod manager	DC admin server (local console and SSH)	Host admin(root) Other non-privileged non RSD users (optional)	User name and password
RMM	Rack controller (local console and SSH)	RMM Admin (root)	User name and password
PSME/Switch PSME	Drawer controller (local console and SSH)	PSME Admin (root) Network admin (no access to PSME functionality)	User name and password
MMC/BMC	Baseboard management controller (local console. local console and SSH)	BMC admin (root)	User name and password

[5] *This may be the case for the Pod manager physical host, but we don't expect it to be the case for other RSD logical management function hosts.*

Table 7 summarizes the different login accesses for Intel® RSD logical management function hosts. Each of these hosts supports an admin login, which is used for managing the host and for installing software on it. This user is entirely in the Intel® RSD trust boundary as it has full control of the host and its software. For the host that supports both the PSME and switch PSME, it will be necessary to support a network admin login such as SSH. The host must be configured so that this login (or SW executed by switch PSME) cannot interfere with (or compromise) the operation of PSME.

3.6.1.2 Support for administrator separation of duty

Intel® RSD supports admin separation of duty by restricting access to Intel® RSD management functionality only to admins with the appropriate Intel® RSD admin roles.

Intel® RSD logical management functions must provide protected logging functionality that will help in forensics work.

3.6.1.3 Permanent denial of service (PDoS)

Optional for RSD v2.2:

Intel® RSD platforms must protect themselves from permanent denial of service that could result from cyber-attacks. Specifically, this means that Intel® RSD platforms must have the ability to recover without requiring administrator physical presence or factory involvement.

Security requirements for supporting automated recovery from cyber-attack include:

- The platform must define a minimum set of FW and/or SW that is critical to enabling platform-remote recovery. The platform must ensure this firmware is always available[6].
- The platform must be able to recover critical FW or SW compromised during an attack without requiring special equipment or physical presence by an administrator.

3.6.1.4 Maintain isolation between composed nodes

Intel® RSD platforms are primarily targeted at cloud datacenters and multi-tenant environments. As such, they must provide strong isolation between workloads running on different composed nodes.

Security requirements for supporting strong composed node isolation include:

- RSD platforms must ensure node isolation in the presence of shared physical resources (e.g., storage, memory, network, etc.)
  - Software (even at the highest privilege) running on a composed node must not be able to access shared resources allocated to another node.
  - Configuration of shared resources must only be performed by RSD management SW. RSD management SW must be protected from composed nodes.
  - Shared resources must support access control to ensure that they can only be accessed by authorized composed nodes.
- The RSD platform must prevent nodes from interfering with other nodes sharing a given physical resource (i.e., support basic QoS for shared physical resources).

[6] *This does not necessarily mean that this firmware cannot be compromised (although this is a way to make it always available!), but that if compromised, the platform has the ability to replace it with a good copy without having to rely on admin intervention.*

3.6.1.5 Support composed node attestation

Optional for RSD v2.2:

The Intel® RSD platform must support attestation of composed nodes, which is a way for a datacenter to prove to a workload owner (remote verifier) that the platform in which that workload is running supports the security needs of that workload.

Specific security requirements for supporting attestation include:

- The Intel® RSD platform must provide a hardware root of trust for measurement (RTM) for composed nodes and for Intel® RSD logical management function hosts.
- Each composed node in the platform must support providing attestation evidence about the environment in which that node was booted.

3.6.1.6 Intel® RSD private rack management network

Intel® RSD private rack management network is a key element of RSD security. This scheme requires RSD management elements such as PSME, RMM and Storage Node BMCs to contain at least two logical networks, where one logical network is attached to the rack-wide private management network and the other logical network is attached to the datacenter's out-of-band management network.

3.6.1.6.1 DHCP server

The DHCP server assigns external IP address to the RSD management elements (PSME, RMM, Storage Node BMC...). DHCP server also communicates the identity of Pod manager, the identity of Authentication Server to the RSD management elements. Further, the DHCP server response also contains the IP address of RMM if the rack contains an RMM.

As such, the DHCP server is in the trust domain and serves as one of the root of trusts on which RSD Security is built. Since DHCP is broadcast based, it is the responsibility of the environment where RSD is deployed to protect against a rogue (either malicious or accidental) DHCP server(s).

3.6.1.6.2 Intel® RSD drawer manageability security

RSD Drawer may have other manageability engines such as Compute Node BMC. RSD Drawer implementation is required to ensure that PSME is able to:

- Securely communicate with these manageability engines (no threat of spoofing or replay attacks)
  - Ensure that PSME is in a master role over these manageability engines (e.g. if the BMC implemented a LAN-based access mechanism, then the access security is configured so that PSME is the only master allowed to effect changes)

3.6.2 Composed node volatile memory clearing

Required:

The composed node reallocation needs to prevent a new user from accessing previous user contents. If the Composed Node resources are de-composed and reallocated, then the contents of volatile memory must be cleared before the resources are reallocated (one way to do this is to perform a cold reset on the Compute/Storage Module).

3.6.3 User to archive data before decomposing a node

Recommended:

If a user wants to preserve the persistent data in memory before decomposing a Composed Node, it is recommended the user store their data in a remote storage area before relinquishing control of the resources in the Composed Node.

3.7 Intel® RSD power and cooling

3.7.1 Power monitoring support

Required:

Intel® RSD must support power monitoring at the rack, drawer and module level. This feature helps the user to determine if a drawer or module exceeds a certain power budget level, and to take action to stay within the rack power limit. If the BMC exposes/supports IPMI, it is strongly recommended to follow specification *Intel® Intelligent Power Node Manager 4.0 External Interface Specification Using IPMI (332200)* or later version, or functional equivalent.

3.7.2 Power budgeting support

Recommended:

Intel® RSD recommends using a power control logic for the BMC/PSME to limit the power to the nodes. If the BMC exposes/supports IPMI, it is strongly recommended to follow the *Intel® Intelligent Power Node Manager 4.0 External Interface Specification Using IPMI (332200)* or later version, or functional equivalent.

3.7.3 Cooling failure reporting

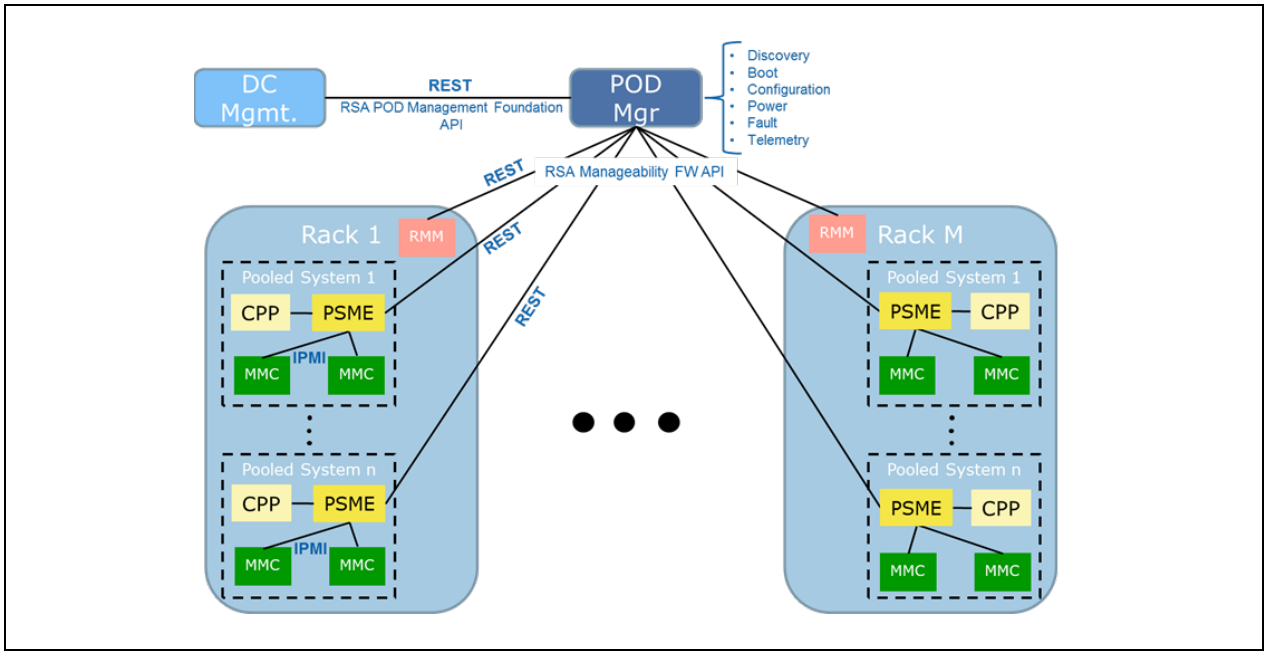
Required:

For improved data center operations, the location and status (on/off/failures) of a fan must be reported so that maintenance actions can be performed. An Intel® RSD system allows the fans to be placed at Rack level, at Drawer level, or at Module level, or in combinations, or have liquid cooling. This must be implemented as described in Intel® RSD API specifications, Table 3.

4 Intel® RSD API

Intel® RSD management software interfaces with PODM, RMM, and PSME using the RSD API (PODM REST API spec, RMM REST API spec and PSME REST API spec), as shown in the block diagram in Figure 14. This section outlines top level API requirements. Refer to the Intel® *RSD API* Table 3 for API details and for individual parameter requirements for each API.

Figure 14. Intel® RSD v2.2 API Block Diagram



4.1 Intel® RSD API interface

Intel® RSD APIs are supported by PODM, RMM, and PSME. The RSD API uses the RESTful protocol. The RESTful protocol is built using HTTP and HTTPS.

4.1.1 Intel® RSD API compliance

Required:

The PSME NB API and RMM NB API must support the Intel® RSD API Specification. Refer to the Intel® RSD API specification Table 3 for the Intel® RSD Schema for required, recommended, and optional parameters for various Intel® RSD APIs. All Intel® RSD components such as PODM, RMM and PSME are required to comply with Intel® RSD API definitions.

4.1.2 Intel® RSD API support for access control and secure communication channel

Required:

All Intel® RSD APIs must support access control and secure communication channels except the root entry point.

§

5 Module Design Guidelines

This section describes the Intel® RSD Platform Compute/Storage Module design guidelines. A Compute/Storage Module is generally a combination of compute and storage resources with network and/or storage connectivity. In some cases, the terms Module and Blade may be a used interchangeably for a single hardware element.

5.1 Module reset, power, and performance

5.1.1 Module power on/off support

Required:

For the ability to conserve power when a module is not in use, the PSME must provide power on/off support for each Compute/Storage Module.

5.1.2 Module reset support

Required:

For dealing with configuration and software changes, the PSME must provide reset support for each Compute/Storage Module.

5.1.3 Power monitoring support

Required:

For improved TCO and high availability of services, it is required that Compute/Storage Modules implement power monitoring support and support rack level power monitoring. Refer to Section 3.7.1 for related details.

5.1.4 Power budgeting support

Recommended:

For improved TCO, datacenter operations and high availability of services, Compute Modules are recommended to support power budgeting. Refer to Section 3.7.2 for related details.

5.2 Module features

5.2.1 Expose TPM capabilities if TPM present

Required:

If TPM is present as a physical component on the motherboard, or embedded firmware in ME, the PSME must expose the TPM version number and capabilities. Refer to the RSD API specification for details.

5.2.2 Expose SMBIOS information to PSME

Required:

The RSD Compute Module must expose SMBIOS information to the PSME for it to expose the module capabilities to the PODM as defined in the PSME API specification.

5.2.3 Expose FPGA capabilities if FPGA is present

Required:

If the FPGA is present in the compute module (either integrated in the processor socket or discrete module in the board), its AFU capabilities must be exposed through OOB for the PSME. Refer to the RSD API specs for details.

5.2.4 BIOS/firmware support for PNC if PNC supported

Required

If the PNC is supported by the rack, compute and PNC modules must provide a mechanism to verify cable connectivity. If the cable does not have a clock, then SRIS should be enabled to allow the compute module to handle link failures.

5.2.5 Minimum 10GbE NIC per module for data plane

Recommended

To keep up with workload networking demands in a hyperscale-like data center, it is recommended that a Compute/Storage Module support a logical connection to one port with a minimum 10 Gb Ethernet NIC speed.

5.3 Module firmware update

Compute/Storage Module firmware updates are generally performed as part of a normal maintenance cycle for bug fixes or upgrading to a new feature set. Critical updates are typically performed for fixing security related issues, important bugs and even new Firmware versions containing new capabilities. An in-band firmware update operation is time consuming and less flexible for management. An RSD aligned Compute/Storage Module must provide capability to do a remote firmware update for all commonly updatable firmware components.

5.3.1 Module in-band firmware update blocking from composed system user

Required

For a more secure design, the Compute module must provide a mechanism to block in-band firmware (such as BIOS, ME, BMC, NIC, DIMM, SSD, Firmware and etc.) updates from a composed system user, but may allow updates from an administrator.

5.3.2 Firmware update authentication

Required

The BMC, Intel ME, BIOS, onboard NIC and other component firmware updates must have an authentication mechanism to verify the update image before the Compute/Storage Module runs the updated firmware.

5.3.3 Module configuration default support

Recommended

If recovered from an unknown state (power loss during configuration change or bad combination of configuration settings or flash corruption, etc.), the module is recommended to provide a mechanism to restore module default settings (similar to entering the BIOS setup screen and invoking restore default menu).

5.4 Module configuration information

Note: In order to make better workload placement decisions and have glass box insights into your datacenter, module configuration information is needed to be exposed. The requirements for each specific implementation determine whether the Module configuration information is stored on power-up or provided in response to a request from the PODM or an orchestrator.

5.4.1 Module processor information (out-of-band)

Required:

The processor information details must be available through an out-of-band interface. Refer to the Intel® RSD API specification Table 3 for details.

5.4.2 Module memory information (out-of-band)

Required

Memory information details must be available through an out-of-band interface. Refer to the RSD API specification for details.

5.4.3 Module storage information (out-of-band)

Recommended

Local storage information details recommended to be available through an out-of-band interface. Refer to the Intel® RSD API specification for details.

5.4.4 Compute module remote OS boot support

Required

The Compute Module in the RSD must provide a mechanism to select the OS boot path remotely. Compute Modules must support either an iPXE interface or an iSCSI interface for a remote OS boot. If the Compute Module supports doing a remote boot, the user must configure the boot method (either local boot or remote boot).

5.4.5 Compute module iPXE support

Recommended

It is recommended that Compute Modules support iPXE for remote booting. If it supports iPXE, the Compute Module should provide a mechanism to configure the iPXE parameters remotely through PSME/BMC. iPXE is more secure than PXE.

5.4.6 Compute module iSCSI support

Recommended

If the Compute Module supports an iSCSI interface, the Compute Module should provide a mechanism to configure iSCSI parameters remotely through PSME/BMC.

5.4.7 Compute module OS boot from local storage

Recommended

If local storage is present on the Compute Module, the Module should have the ability to support an OS boot from local storage. The local storage on the Compute Module should consist of resources such as SSD/HDD, M.2, or NVDIMM.

5.4.8 Module security support information

Recommended:

The Compute/Storage Modules are recommended to support Root of Trust Measurement (RTM). Security support information (such as TPM presence, Intel TXT, and ACM support) for a Compute/Storage Module is recommended to be given to the BMC/PSME to communicate the Module's security-related support information to the PODM.

5.5 Reliability availability and serviceability (RAS) support

For improved overall data center operations and to understand the health of your datacenter at all times, health information needs to be exposed properly to the orchestrator. This section covers those requirements.

5.5.1 Out-of-band health event support

Required:

The BMC on a Compute/Storage Module must monitor hardware errors and health information for the Module (out-of-band, not dependent on the OS that is running on the Module), and provide the information to the PSME. One of the mechanisms for the PSME to get the health information from the compute module to have the Intel® RSD v2.2 IPMI FW extension API, refer to Table 3. Implementation must align to APIs defined in the Intel® RSD API spec.

5.5.2 Error persistence over reboot

Recommended:

Hardware error information is recommended to persist across a reboot unless cleared by an administrator or the hardware is replaced or is otherwise corrected. Refer to the Intel® RSD API specification Table 3for error and health information reporting format details.

5.5.3 Reporting health and performance information

Required:

The PSME must provide a mechanism to provide health and performance information to the PODM. Refer to the Intel® RSD API specification Table 3 for error and health information reporting format details.

§

6 PCIe* Direct Attach Pooled I/O Design Guidelines

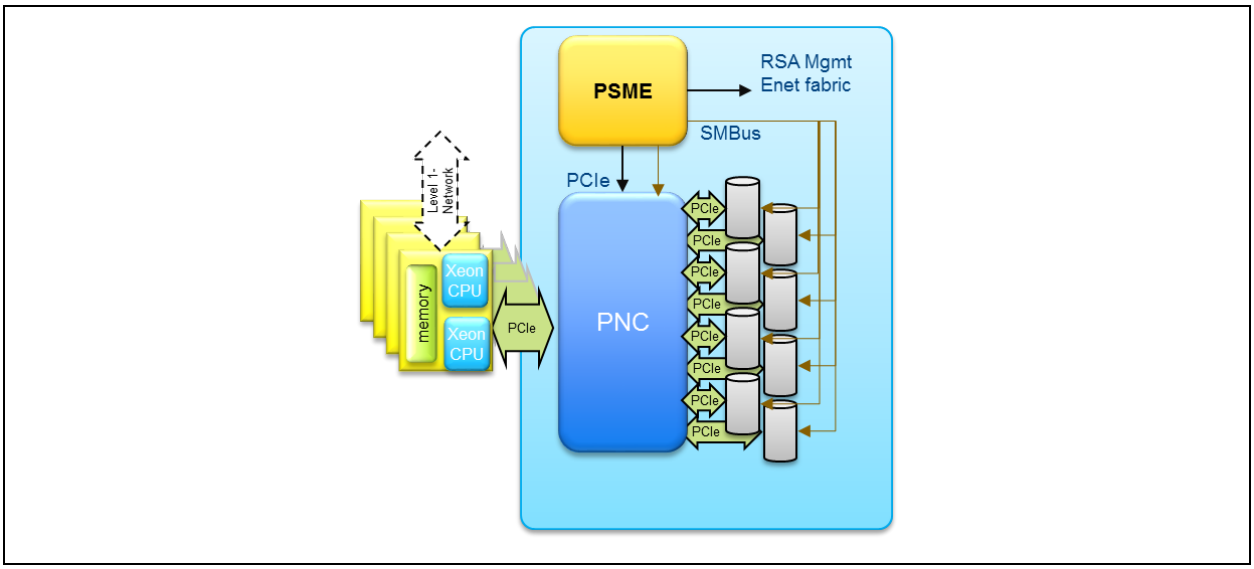
6.1 Overview

Overall the PCIe* direct attached Pooled I/O System is optional, however if it is present, this section covers the requirements for support for such a system.

The Direct Attach Pooled I/O System provides disaggregation of I/O devices from the Compute Module. The I/O devices include storage and FPGA devices that are directly attached via a compliant PCIe electrical bus interface. It shall be noted that optical transports can be used in connecting the I/O pool system but they must be transparent to the PCIe interface at both the initiator and target ends. This section outlines the Direct Attached Pooled I/O related features and requirements containing PCIe I/O devices.

The Direct Attached Pooled I/O System enables multiple nodes to access a pool of PCIe I/O devices which provide a flexible resource assignment and to maximize the efficiency and utilization of I/O resources. Within an Intel RSD system, the Pooled System Management Engine (PSME) configures the I/O devices, PNC, and node connections to compose and release I/O resources within the I/O pool as instructed by the Pod Manager (PODM). The assignment of these I/O devices to a particular node is accomplished through the Pooled Node Controller (PNC), by logically binding PCIe End-Point devices to upstream ports that are typically attached to compute modules in the system. It is possible that a single module could be attached to two or more PNC upstream ports for multi-pathing capabilities. Figure 15 shows an example of a pooled I/O system within the RSD architecture.

Figure 15. Example of IO pooled system with PNC and PSME



6.2 System topology and mapping

To facilitate the composition of systems, the PODM must have a comprehensive view of the system topology in order to make effective decisions on how to logically assemble the system. At a high level the PODM needs to know the following key attributes:

- The number of upstream ports that are availability in the PNC(s), the widths of the ports and the speed capabilities.
- Compute node attachment to the pooled I/O system; the PNC port number to which a compute node is attached.

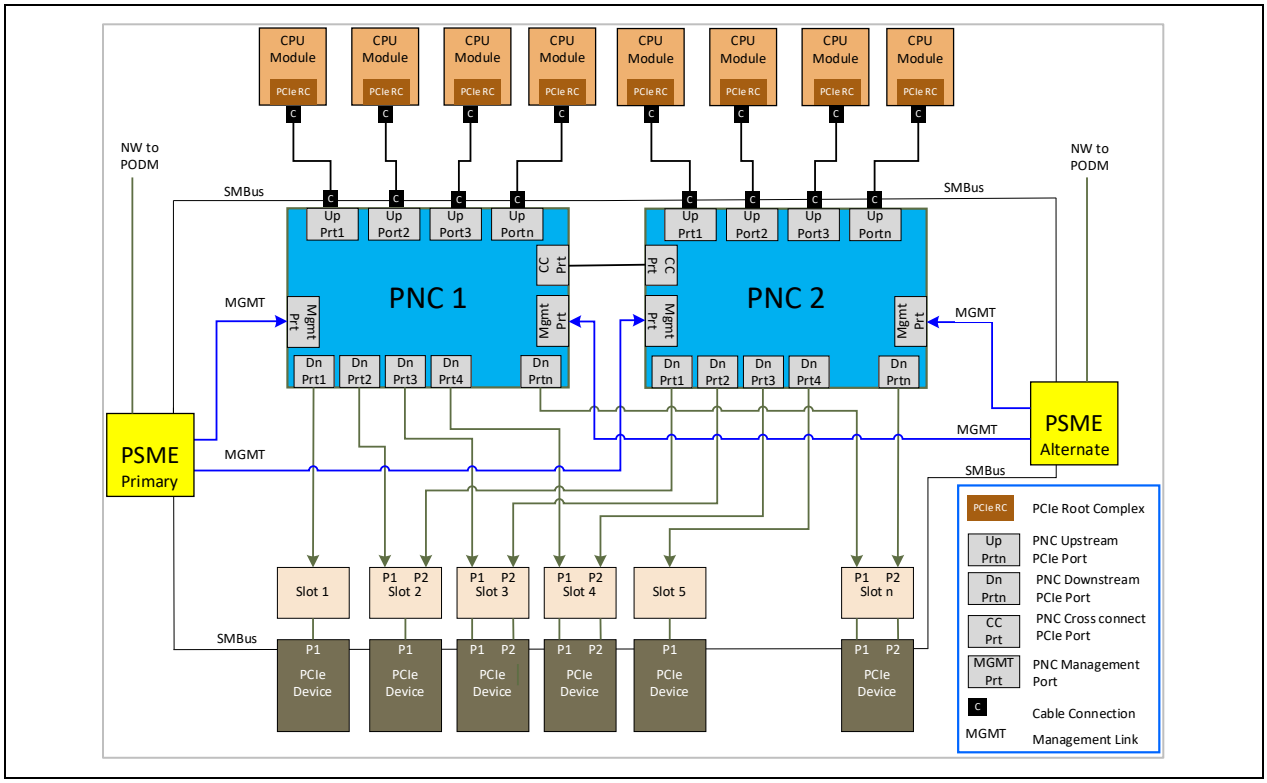
- The number of downstream ports that are available, the widths of these ports and the speed capabilities.
- How the downstream ports are physically mapped to the I/O devices or slots. It is possible that a slot may be wired to support more than one PCIe port, i.e., U.2 connectors may be configured for dual port devices.
- What devices are populated and the mapping to PNC downstream ports.

The PSME is responsible for exposing this information to the PODM through the Redfish-based APIs, Table 3, at the time of power up or system reset. Once the topology is established, the PSME will discover the I/O devices which is covered in Section 6.3. Discovery of the I/O devices is accomplished through the out-of-band interface to the I/O devices described in Section 6.2.12.

Multiple PSMEs may exist in a system and they may be capable of managing one or more of the resident PNCs as shown in Figure 16. A Primary PSME will be designated in the case where multiple PSMEs have management access to a PNC. The Primary PSME will be responsible for the configuration and management of the PNC. Other PSMEs having management access will be designated as alternate and will have a passive or inactive role with the attached PNC.

The assignment of the Primary PSME may be predetermined by the I/O system, but must be acknowledged by the PODM prior to configuring the I/O system. The PODM will be able to assign the Primary PSME if more than one PSME are present for an I/O pooled system. Refer to Section 6.2.5 for more details.

Figure 16. Example of System Topology



6.2.1 Enumeration of components in the system must be deterministic and persistent across power or initialization cycles

Required

The enumeration of components, within the system must be deterministic and consistent across power cycles or system initialization unless the system has undergone reconfiguration. This is required to provide the PODM a consistent view of the pooled system upon each power up or system reset.

Such enumerable components include the PSME(s), PNC(s), external port connections, and device slots, downstream and upstream PNC ports.

6.2.2 PSME exclusive management link to PNC

Required:

A private management link from the PSME processor to the PNC is required to configure and manage the PNC.

- The PSME shall support a separate management link to a Mgmt. Port for each PNC that the PSME is managing within its management domain.
- Capable of mapping the device to the PSME Mgmt. link for firmware upgrade.
- The PNC shall be managed only via the Mgmt. port. Manageability includes configuration, telemetry, and firmware update.
- Capable of exposing telemetry of the PNC device.
- Only one PSME may govern the PNC at any given time.

The specifics of the type, speed and width of the management link are defined by the PCIe switch used in the system and is left to the designer for implementation. The management link is used to configure the PNC ports, establish any necessary PCIe domains within the PNC, collect telemetry information, and monitor any error conditions. It can also be directly mapped to an I/O device PCIe port for firmware upgrade and control.

6.2.3 Expose and enumerate PNC devices in a pooled system

Required:

The PSME shall expose the total number of PNCs that are resident in the pooled IO system. The PNCs will be logically enumerated so that the PODM is able to reference a specific PNC component. This will allow the PODM to logically map data connections from a compute node connected to an upstream port of an enumerated PNC to an I/O device.

6.2.4 Expose PSME mapping of management connections to PNCs

Required if multiple PSMEs management connections are available to a PNC:

The PSME shall expose the mapping of its management interfaces to the enumerated PNC(s) resident in the pooled system. As shown in Figure 16, a system may have multiple PSMEs, each having a management link to one or more PNCs. The PSME shall expose which enumerated PNCs it has a management connection with.

6.2.5 Assignment of primary PSME for PNC

Required:

A Primary PSME will be assigned to a PNC:

- Only one Primary PSME may be assigned to a PNC at any given time and
- Only the Primary PSME may govern the PNC.
- The I/O pooled system may predetermine the assignment; it may be persistent across power or system reset cycles.

- All assignments must be acknowledged by the PODM prior to the PSME configuring the PNC.
- The PODM may assign the Primary PSME, overriding any I/O pool system assignment.
- Other PSMEs with management connections to the PNC will be assigned as alternate.
- Alternate PSME(s) will take either a passive or inactive role with the PNC. Any state information of the Primary PSME should be migrated to Alternate PSMEs.
- PODM will notify all PSMEs of their state of Primary or Alternate upon change of state.

6.2.6 Expose and enumerate PNC upstream ports

Required:

The PSME shall be able to expose and enumerate of upstream ports controlled by each PNC.

This information is used in establishing connection paths of I/O resources on downstream ports to compute modules on the enumerated upstream ports.

Attributes to be reported:

- PNC number
- PNC upstream Port Number
- Port max lane width (optional)
- Port current lane width
- Port max speed (optional)
- Port current speed

6.2.7 Expose and enumerate PNC downstream ports

Required:

The PSME shall be able to expose and enumerate the PNC downstream ports configured within the PNC.

This information is used in establishing connection paths of I/O resources on downstream ports.

Attributes to be reported:

- PNC number
- PNC upstream Port Number
- Port max lane width (optional)
- Port current lane width
- Port max speed (optional)
- Port current speed

6.2.8 Expose data path cross-connections between multiple PNCs

Recommended:

The PSME shall be able to exposing the cross connection topology between enumerated PNCs that allow devices to be logically allocated or assigned to compute modules attached to an adjacent PNC upstream port.

Generally the cross connection between PNCs provides a connection path from any of the I/O devices to a compute node upstream port. This provides a means for the PODM to determine locality of the device to the compute node (such as the number of PNCs to reach the end point device) and to provision I/O BW appropriately across the cross-connection link. Note the PODM should first assign I/O resources to the host that does not utilize a cross-connection link.

6.2.9 Expose and enumerate device slots of the IO pooled system

Required if the pooled system has connector slots for IO devices:

The PSME shall be able to enumerate and expose the I/O slots and slot PCIe port configuration within the I/O pooled system. This will identify a PCIe slot and PCIe port configuration of that slot within the I/O pooled system. This is independent of the slot being populated or unpopulated.

6.2.10 Expose mapping of device slot connectivity to PNC downstream ports

Required if the pooled system has connector slots for I/O devices:

The PSME shall be able to expose the mapping of slot ports to PNC downstream ports. As shown in Figure 16, device slots may be wired to have multiple PCIe port connections to one or more PNCs.

This will allow the PODM to logically map data connections from a PNC enumerated downstream port to an enumerated I/O device slot port. Consider the case where dual port SSDs are connected to two independent PNCs providing multipath capability. It should be noted that it is possible to connect a single port device into a dual port slot.

6.2.11 Compute module to PNC upstream port connection ID mapping

Required:

There shall be a means for the PODM to map connections from a compute module to a PNC upstream port. Such a means is accomplished by matching a pair of unique connection IDs from the compute module end and from the I/O pooled system end. In the case of removable cables, this may be accomplished by reading the cable ID field or serial number and matching it with the compute module end. Where connectivity is permanently set, such as in hardwired backplanes, a hard-coded connection ID could be presented to the PODM for mapping.

Refer to Section 6.8.5 regarding connection identification for the compute module.

6.2.12 Expose the connection presence of each upstream port

Optional:

The PSME shall expose the state of the connection presence for each PNC upstream port. In the case of removable cables to make the connection to the compute module, this would be a cable present detect indication.

6.3 I/O device discovery support

I/O Devices are discovered by the PSME and exposed to the PODM as I/O resources that can be composed and logically assembled with a compute module.

Device discovery happens upon:

- Power up and initialization of the pooled system or subsystem
- Hot add of an I/O device

6.3.1 Expose the presence of an I/O device

Required:

The PSME shall have the means to expose the presence of an I/O device in a PCIe slot. This is used to comprehend the population of I/O devices in PNC downstream slots.

6.3.2 Discovery of device type and capability

Required:

The PSME shall be able to discover the IO device, its type, i.e. SSD or FPGA, along with capabilities such as capacity, performances, firmware, serial numbers and any pertinent information for each device.

Mandatory

- Type of device, i.e. SSD, FPGA, etc.
- Manufacturer ID, model and serial number
- Number of PCIe ports, width and speed

Optional

- Storage capacity
- Performance
- Wearing information
- Firmware revision level
- Memory capacity
- Number of memory channels and speed
- Board service package rev level
- Function of programmable logic
- Other attributes or device capabilities

6.3.3 PSME configuration of I/O device support if sharing of IO device supported

Required:

The PSME shall have the ability to configure each I/O device individually if sharing of the I/O devices is supported. For example, if the NVMe SSD is present, it should be able to partition the SSD before sharing SSD between multiple compute nodes.

6.3.4 Expose SSD and NVME metrics

Recommended:

The PNC controller will expose:

- PCIe link metrics
- SSD metrics per SSD device

6.4 I/O device assignment to compute module

This section describes the requirements for logically assigning and releasing of device resources to compute nodes via upstream PNC ports.

6.4.1 Full assignment of a device PCIe function to a single compute node

Required:

The PSME shall be able to connect the complete or full PCIe function to an upstream port.

Note: If the system supports multiport or multifunction devices, each PCIe function can be logically assigned to independent PNC upstream ports. As an example, a dual port device will present a PCIe function per port that can be assigned, or logically bound, to two different hosts or even to a single compute module using multiple upstream ports.

6.4.2 Assignment of single PCIe function to multiple upstream ports

Optional:

In Intel® RSD v2.1 assigning PCIe functions across multiple upstream ports is not supported to partition device resources. Virtual functions of a device may be assigned to the same PNC upstream port and compute module.

6.4.3 Dynamic assignment of a device shall not affect other device connectivity

Required:

Dynamic assignment of an I/O device ownership shall not impact other device-host connections.

This is accomplished via a hot-plug event to the system without the system having to be reset. This allows for the composition of I/O resources in logically assembling a system without disruption to other systems.

6.4.4 Dynamic release of a device shall not affect other device connectivity

Required:

Dynamic release of a device ownership without impacting other devices-host connections. This is accomplished via a hot-plug event to the system without the system having to be reset. This allows for releasing I/O resources in disassembling a system.

6.4.5 Devices with data storage must secure data upon release

Recommended:

Data stored on a device should be secured upon releasing the assignment of a device from the compute module. The data may be encrypted or fully erased to ensure security of the data.

6.4.6 I/O resources must be in an unassigned state prior to assignment to a compute node

Required:

I/O device resources shall be in an unassigned or released state before they are permitted to be assigned to a compute node. Assigning an I/O device from one compute node to another compute node is not permitted without first releasing the I/O resource.

6.5 Adding or removing devices from the I/O pool

Hot-plug is used for resource modifications for the pool such as drive addition or removal. The hot-plug event may be from a physical event, or it may be necessary to emulate the event by the management software on the PSME.

6.5.1 Physical hot add support of devices to the IO pool

Required:

Physically adding a device to the I/O Pool shall be supported. The PSME will be responsible for the logical binding of the device to a compute node once present. Upon the addition of I/O devices the PSME will be responsible for discovering the device. Refer to Section 6.3.

6.5.2 Managed removal of device from the I/O pool support

Required:

PODM shall notify all compute modules that are logically attached to the device to place them in a quiescent state. The Compute Module will in turn notify the PODM when it has completed the task. After all the Compute modules have reported their quiescent state to the Pod Manager, the Pod Manager will inform the I/O Pool PSME that the device is ready for removal. The PSME will then activate an indicator to notify personnel which device is able to be removed from the I/O pool.

6.5.3 Surprise removal of a device from the I/O pool support

Required:

The compute module and the PNC shall support surprise disconnect of a device and continue in an operational state upon recovery of the disconnection. The effected PCIe link shall be functional upon adding a device to the slot. The PSME will alert the PODM of the event.

6.5.4 Surprise disconnect of the IO pool shall be supported

Required if removable connections to the pool are employed:

PCIe busses connected to the Pooled I/O System that are removable shall support surprise disconnect. The compute module shall be able to continue in an operational state upon recovery of the disconnection. The PSME will alert the PODM of the event.

6.5.5 Notification of devices added or removed from the I/O pool

Required:

Upon the physical addition or removal of a device, the PSME will alert the PODM of such an event. Upon addition, it will also report the device assets to the PODM.

6.6 Error handling and telemetry

6.6.1 Down port containment support for all PNC downstream ports

Required:

All Downstream ports of the PNC shall support Down Port Containment. This will contain link failures in the event of a link down or surprise disconnect of a device.

6.6.2 Fault and service indicators for I/O devices

Recommended:

The PSME is recommended to activate a fault or service indication for each device in the I/O pool. Typically this would be a LED indicator located near the I/O device.

6.6.3 PNC trap of PCIe error events detected on the PCIe link

Recommended:

PCIe errors are trapped within the PNC and sent to the PSME via PNC management link. The PSME may log the error events or alert the Pod manager depending on the severity of the error and system policies. If the link is down and not recoverable, the PSME shall alert the PODM as a surprise disconnect.

6.6.4 Expose PNC, device and I/O pooled system telemetry

Recommended:

The PSME should be capable of reading telemetry information from the PNC, I/O devices and enclosure, and be capable of exposing the information to the PODM.

Refer to the Section 7.2.4, PSME telemetry requirements for the details.

6.7 Pooled I/O system support

This section describes the requirements for the pooled I/O system and chassis.

6.7.1 Device serviceability while system powered on

Required:

The Compute/Storage Modules must be able to service I/O devices without affecting running Compute/Storage Modules. In order to achieve this, the I/O chassis must support serviceability while the I/O chassis is powered, and must allow device serviceability without affecting other devices that are currently in use.

6.7.2 Pooled system enclosure management support

Recommended:

Enclosure management of the Pooled I/O System and the I/O devices is provided via the PSME.

- Thermal monitoring within strategic location of the chassis
- Thermal monitoring of each I/O device
- Voltage monitoring of each voltage rail
- Reset function to each I/O device in the I/O pool
- Reset function for the entire I/O pooled system
- Power on/off control
- Activate indicators for fault and/or attention for each I/O device
- I/O device power on sequencing for power surge control if needed

6.7.3 AUX power to cable connector

Optional:

AUX power to the cable connection on both ends of the cable allows for the PSME/BMC to read the cable information prior to power on of the system and establish connection mapping. Such fields for cable ID can be read and passed on to the Pod Manager. The system mapping can be done prior to power on saving time in the system coming up. This should be done from both the Compute module and IO pool endpoint of the cable.

6.7.4 Exposing cable electrical parameters for cable signal drive support

Optional:

Reading the cable loss characteristics and configuring the appropriate I/O drive capability of the cable electrical interface. This would provide a means to optimize cable I/O drivers for various lengths and cable characteristics. This information may be stored and accessed via the I^2C interface of certain cables. It would be the responsibility of the local BMC at each end of the cable to perform this operation and would be done during power up or the detection of cable attachment. This should be done from both the Compute module and I/O pool endpoint of the cable.

6.8 Compute module requirements for I/O pooled systems

This section describes the requirements necessary for the compute module to support a direct attached PCIe pooled system.

6.8.1 Independent PCIe* domain per compute module connection

Required:

Establish independent PCIe domain per host, (zone), whereby actions or events in one domain do not affect the operation or state of other zones configured within the PCIe switch. Host issued reset only impacts its own zone (not the rest of the switch).

6.8.2 Down port containment support for all connected ports

Required:

All Downstream ports of the Compute Module that are connected to the I/O Pooled System shall support Down Port Containment. This will contain link failures in the event of a surprise link down or disconnect.

- Root must be configured to support completion time out; this will create an all 1's completions for any pending transactions.
- The OS must be able to recover from a surprise disconnect; the PCIe driver must properly handle all 1's completion of the PCIe Transaction.
- Be able to reconnect to the device and establish the link upon reconnection of the link.
- PCIe Port that supports DPC should not set the Hot-Plug Surprise bit in the Slot Capabilities register; refer to the implementation note regarding surprise Hot-Plug with DPC in the PCIe Base Specification.

6.8.3 BIOS shall allocate memory space for all potential I/O devices

Required:

BIOS must allocate memory space for all potential PNC slots for dynamic assignment of I/O devices.

1 MB is the minimum allowed memory space as defined by the *PCI Express Base Specification.*

6.8.4 Compute module visibility of IO device controlled by the PSME

Required:

Host visibility to devices behind a switch port is strictly under the control of the PSME. Only when a device is assigned to a host will the host be able to enumerate and see the device. Compute modules will be inhibited to seeing any devices prior to the PNC being configured and enabled by the PSME.

6.8.5 Compute module connection identification

Recommended:

Connection identification is supported on the compute module PSME.

The compute module presents a unique connection ID that corresponds to the I/O pooled system PNC upstream port connection. This would allow the Pod Manager to determine the mapping based on matching connection ID fields from the compute module and PNC port of the I/O pool.

For removable cables this could be the cable ID or serial number that matches the other end of the cable attaching to an I/O pool. For permanent connections, such as hard wired backplanes, this could be a hard coded ID value.

6.8.6 Compute module managing the assigned I/O device

Optional:

Managing the assigned I/O device from the compute module is a platform design decision.

6.8.7 Compute module managing the I/O pool system is not allowed

Required:

Out of band management connection to the I/O pool from the host node directly is not permitted.

All OOB management is performed by the local PSME of the I/O pool. For instance, the Cable Management Interface, CMI, specified in the PCIe External Cable Specification 3.0 shall not be used as a management interface into the I/O pooled system.

§

7 PMSE Design Guidelines

This section describes RSD Platform PSME design guidelines.

PSME overview

The PSME is responsible for Drawer identification management, as well as supporting the PSME REST API and communicating with the BMC to perform Module-level management. In some implementations, RMM and PSME may co-exist in the same hardware. In some implementations, and in some storage bricks the PSME functionally may be provided by the BMC. In general, it is required to implement all of the required APIs as defined in the Intel® RSD API specifications, refer to Table 3 for a list of applicable specifications.

7.1 PSME reset (power on)

Required:

To troubleshoot possible issues and for overall improved data center operations. Must be able to remotely restart PSME services.

7.2 PSME configuration management

7.2.1 PSME API compliance

Required:

If PSME is used, all required APIs must be implemented as defined in PSME API specification.

7.2.2 PSME authentication credential

Required:

In order for the PSME to identify and authenticate a PODM, the PSME must be provisioned with an authentication credential that the PSME can use to verify the identity of the PODM.

7.2.3 PSME time sync mechanism

Required:

For ease of troubleshooting overall improved datacenter operations, the PSME must log all management events with a timestamp. The PSME must synchronize time from PODM periodically and keep the time tolerance within one second as compared to PODM.

7.2.4 PSME telemetry requirements

Required:

For ease of troubleshooting overall improved datacenter operations, the PSME must provide a set of telemetry data provided by the Intel® PSME REST API list Table 3. It is recommended that all telemetry data be time stamped.

7.2.5 Serial-over-LAN and KVM credential change with user changes

Recommended:

PODM is recommended to change the credential for serial-over-LAN and KVM when the composed system user is changed. This is to ensure the previous composed system user does not have access to the same system. If the composed system is part of a user group, it may keep the credential the same.

7.2.6 PSME support for power and thermal capping

Recommended:

It is recommended that the PSME collect the power budget and capping capabilities, and provide them to the RMM (if discrete) or via RSD REST APIs. The PSME can provide the monitored power consumption information for each Compute/Storage Module (and total of PSME managed resource power consumption). In addition, the PSME is responsible for managing the fans in the Rack or Drawer, if present. Implementation must align to the Intel® RSD API specification Table 3.

7.3 PSME software update

This section describes requirements for PSME software updates if a PSME is implemented in the solution.

7.3.1 PSME remote software update

Required:

The PSME must be designed to accept online software updates or offline (staged) software updates.

For an online software update, the Drawer/Module functionality must not be lost during the software update process. But during the PSME software update/reset window, the PSME may not respond. The PSME must inform its clients (such as PODM or RMM) before it goes through an online update process and must request that the clients re-register after a pre-specified time period.

For an offline software update, the Drawers associated with the PSME are reset. During this time Drawer functionality is lost. This mechanism must normally be performed during scheduled maintenance time, and the workloads must be shut down or migrated to a different Drawer before performing the update.

7.4 PSME reliability, availability and serviceability support

For improved overall data center operations and to understand the health of your datacenter at all times in order to take appropriate action, the PSME is responsible for handling errors related to the PSME managed assets, such as Drawer and Module.

7.4.1 Drawer event reporting

Required:

For improved overall data operations and the ability to keep the datacenter running at all times, drawer serviceability is required. In order to do that, the RSD APIs must support event reporting for Drawer insertion and Drawer removal events.

Drawer insertion:

- Once a Drawer is inserted and powered on, the RMM must assign the PSME ID, and the RMM must advertise the new PSME to the PODM.
- The PSME must provide the Drawer details such as Compute/Storage Module presence information to the PODM.

Drawer removal:

- Once a Drawer is removed, there is no PSME that can communicate its removal. The RMM must periodically poll the PSME for its availability. If the PSME is not responding, it is either due to PSME failure or Drawer removal. The RMM must report the PSME failure or Drawer removal event to the PODM.
  - Module removal.
  - Module failure events, such as boot failure detected by BMC watchdog timer Implementation must align to the RSD API spec.

7.4.2 Drawer (PSME) hot add only when RMM is present and running

Recommended:

To avoid periodic discovery of the system components, it is recommended to insert a Drawer only into a Rack that has an active RMM, if RMM is present. If the RMM fails, the PSME IDs are kept intact to keep the Composed Nodes running.

This avoids the situation where a Drawer is moved into a Rack with an RMM that has failed, and the PSME continues to provide access to Composed Node resources provided by that Drawer.

Furthermore, it is recommended that the PODM detect RMM failure. If any new PSME shows up in a Rack with an inactive RMM, it is recommended that the PODM reject the new PSME until the RMM is active again.

§

8 RMM Design Guidelines

This section describes the RSD Platform RMM design guidelines. RMM API is a required component for RSD, but the hardware that runs the RMM firmware could be dedicated or shared with PSME or other components. RMM can either be run as a separate component, or APIs can be implemented in PSME or PODM (for example).

8.1 RMM overview

The Rack Manager (RMM) is responsible for handling infrastructure functions such as rack level shared power, rack level shared cooling, and assigning PSME IDs.

8.2 RMM reset (power on)

Required:

The RMM must be configured first with the PODM authentication credentials (Refer to Section 7.2.2 for details). Once the credentials are configured, the RMM can communicate with the PODM without any external configuration.

8.2.1 RMM boot and PSME ID assignment if descrete RMM present

Required:

If a separate RMM component is present, the RMM must assign a unique PSME ID for each PSME that is managed by an RMM. The conditions listed in Table 8 must be met during the RMM boot process and the PMSE ID assignment processes.

If redundant RMMs are configured for high availability support in a hyperscale datacenter, the existing Composed Nodes can continue to operate in the event of an RMM failure.

Table 8. RMM and PSME interaction during boot and ID assignment

RMM condition	PSME condition	Requirement specification
During run-time, Primary and secondary RMMs running and primary RMM fails	During run-time, previously configured PSME running	The secondary RMM must take control from primary RMM without changing the PSME ID.
During boot-up, RMM failed or no RMM found	During boot-up, previously configured PSME resource found and running	PODM has the list of PSME that was previously running, reports the RMM failure and continue to use the PSME
During boot-up, RMM failed or no RMM found	New PSME (Drawer) is added to the Rack. Since PSME is not finding RMM, the PSME ID will not be set	The PSME must wait for RMM to assign the PSME ID and complete the boot operation. PODM will not use the new PSME
During boot-up, RMM finds no PSME	No PSME ID will be allocated	PODM will not use PSME
During boot-up, RMM present	During boot-up, PSME present	The PMSE must ping the RMM by sending the active PSME ID and continue to boot until done.

8.2.2 RMM assigns PSME ID if PSME not configured

Required:

If a separate RMM component exists, and if the RMM finds a PSME that reports itself as not being configured, then the RMM must configure the PSME ID. This condition occurs under two conditions: when the RMM is reset, and when a PSME is hot-added.

8.2.3 PSME enters “PSME ID Not Configured” state

Required:

To properly discover, manage and recover from newly added RSD components after a failure scenario, if a new RMM is found during a PSME boot process (as a new insertion to rack or for a redundant configuration for example), the PSME must advertise itself as “not configured.” If the same RMM is found, then the PSME must retain the old PSME ID until the RMM assigns a new ID.

8.3 RMM general support

8.3.1 RMM event handling

Required:

Similar to PSME, the RMM, if present, must handle the following events and report the events to the PODM:

- Drawer Insertion
  - New PSME detected and the PSME is assigned with an ID
  - New drawer location identified
- Drawer Removal
  - PSME removed from the RMM list
  - Drawer removed location identified
- RMM internal errors reported
- (If HA RMMs are present), a new RMM becoming the active primary after a redundancy loss is signaled
- Power events such as a new power supply coming online or power supply failure
- Power threshold crossing events
- Cooling threshold crossing events

All events must align to Intel® Rack Scale Design Rack Management Module (RMM) API Specification definitions, refer to Table 3.

8.4 RMM power and cooling support

8.4.1 Rack power monitoring support by RMM if shared power is used

Required:

If rack-level shared power is used, the RMM must provide power monitoring support for power supplied to the rack. Refer to Intel® RSD API specifications Table 3, Intel® RSD Reference Documents for the data format.

8.4.2 Rack power budgeting support by RMM if shared power is used

Recommended:

If rack level shared power is used, the RMM is recommended to provide setting a power limit for racks. Refer to Intel® RSD API specifications for the data format, refer to Table 3, Intel® RSD Reference Documents.

§

9 Pod Manager (PODM) Design Guidelines

This section describes the PODM design guidelines used on the RSD Platform.

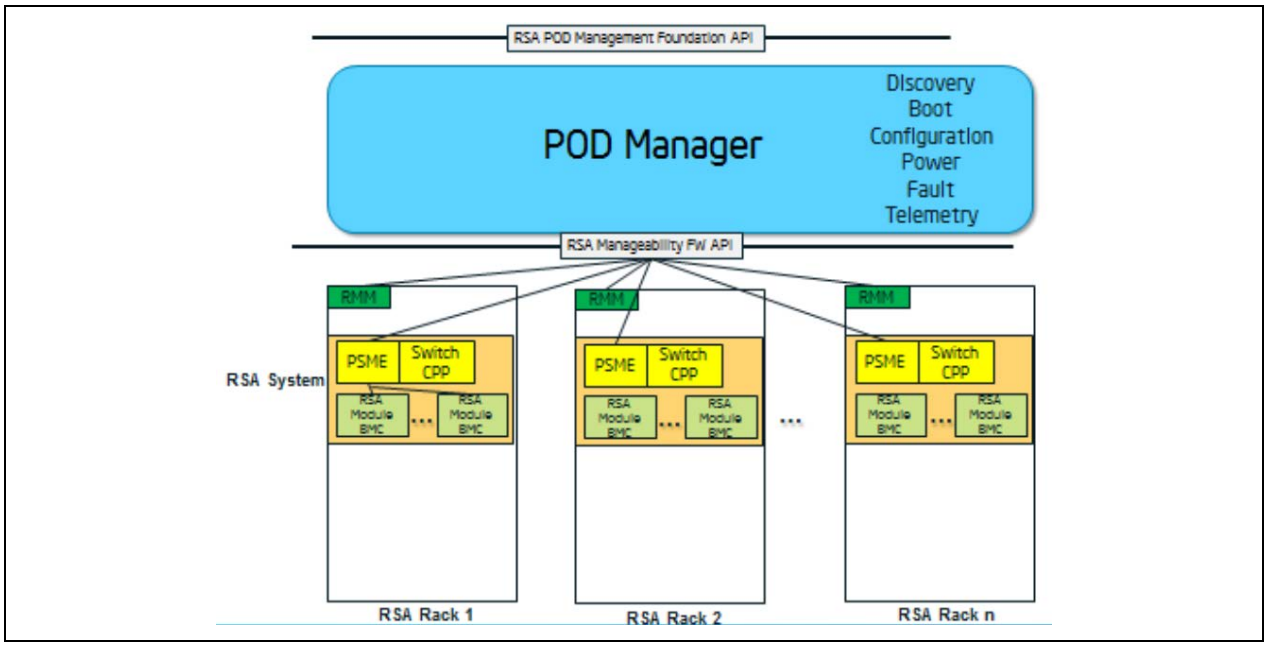
9.1 PODM overview

The PODM, shown in Figure 17, is responsible for the discovery of resources in the Pod, configuring and managing the resources, and composing a logical server.

PODM is an optional separate component and will often not be required in-rack. However to be 'RSD conformant' a Rack must be able to be managed by a certified PODM. This section outlines requirements for those who implement a Pod Management solution. All APIs, as defined in the *Intel® Rack Scale Design Pod Manager REST API Specification*, must be implemented as defined in that spec, refer to Table 3.

When working within the Pod, the PODM interacts with RMMs in the Rack and PSMEs to create a representation of the RSD Pod. The PODM assigns the physical resources to match the requirements specified by the RSD Solution Stack by creating a logical resource, called a composed node or logical server.

Figure 17. Logical view of the Intel® RSD v2.2 Pod manager



9.2 PODM configuration management

9.2.1 PODM powered independent of rack power

Required:

To maintain the PODM services even when the rack is reset, the PODM must be independently powered.

9.2.2 PODM REST API compliance

Required:

PODM must be in compliance with the *Intel® Rack Scale Design Pod Manager REST API Specification*, refer to Table 3.

9.2.3 Secure communication channel for management network

Required:

For a more secure design, the PODM must be connected to the RMM (if present) and PSME through a private network. Any management related activity (such as reconfiguration) must be performed only after establishing a secure communication channel between the PODM and the PSME, and between the PODM and the RMM.

9.2.4 PODM authentication certificate

Required:

For a more secure design, the PODM must securely identify or authenticate itself to the PSME and RMM (if present) using a PODM authentication certificate.

One way to do this (and there are others), is for the PODM to create a private/public key pair, then request a certificate (from a certificate authority) that includes a public key for this key pair. The certificate gets provisioned to RMMs and PSMEs that use it to authenticate this PODM (Refer to Section 7.2.2 for more details).

9.2.5 PODM timestamp support

Required:

For better troubleshooting and improved overall data center operations, the RSD components must log all events with a timestamp. One-way to achieve this is for the PODM to synchronize the time using syslog or NTP.

9.2.6 Only one active PODM per pod

Required:

If PODM is implemented to manage a Pod of RSD Racks, each Rack is assigned to an instance of PODM and PODM uses Rack IDs to distinguish the racks in the Pod. If HA PODM is implemented, each redundant PODM must send heartbeat requests to the other PODM to determine HA failures (active-active, or active-passive). In the HA PODM, the CMDB for the PODM instances must be in sync to provide a status for the HA PODM.

9.2.7 PODM to allow addition of new drawers only when RMM is alive

Required:

If a new Drawer (with a new PSME) is added to a rack, the PSME is not activated with PODM credentials. Hence PODM must not be able to use this new drawer for composing nodes.

§

10 Network Switch Design Guidelines

This section describes the design guidelines for the network switch configuration.

10.1 Intel® RSD networking overview

In a typical RSD Platform, Compute/Storage Modules are connected at the Module level to a NIC. The PSME for switch management could be on the switch itself or it could be external to the switch.

In Intel® RSD before v2.2 the PSME network agent is designed for managing the Red Rock Canyon (RRC). RRC switches do not provide a network-accessible configuration command set, RSD provided APIs and an implementation for configuring RRC switches; the same APIs have been implemented by OEMs for configuration of Top-of-Rack (ToR) switches.

Starting with RSD v2.2, RSD moves to Top-of-Rack switches which support comprehensive configuration command sets. RSD's goal is to enable network vendors to align with a common configuration model and an evolving industry standard. Until this goal is available, OEMs and users are encouraged to leverage industry recognized and widely adopted configuration management tools (for example, Ansible) which provide scalable configuration of a switches from numerous vendors. The switch APIs from previous versions are still defined in the PSME (and PODM) API specifications to provide backward compatibility for current release.

10.1.1 Module-to-port mapping configuration file support if dynamic discovery not supported

Required:

If dynamic discovery of the mapping between switch ports and Modules is not possible, then there must be a configuration file that is available in the PSME to describe the physical connections between the switch ports and the Modules for the Intel® RSD management software.

10.1.2 Switch PSME support for base network services

Required:

The switch PSME must expose the interface/API for the following base network services:

- HW management
- Interface Management - user can configure switch port interfaces
- Protocol Management
  - VLAN configuration
  - LAG configuration using a number of ports
  - ACL configuration
  - IDs for switch port neighbor ports (e.g., mapping switch ports to NIC port MAC address)
  - MAC Address - view and configure switch MAC address tables for the purpose of defining rules about which packets are forwarded or discarded.

10.1.3 Device discovery and switch configuration reporting

Required:

The switch PSME should provide initial device discovery and switch system configurations before compute systems can be composed. It is generally performed autonomously without initiating a request from Pod Manager.

10.1.4 Switch functionality change event generation

Required:

If any of the following conditions occur, the switch PSME should generate an event to Pod Manager and notify the switch state:

- Port is not functional
- Port state is up or down
- Link state is up or down

§

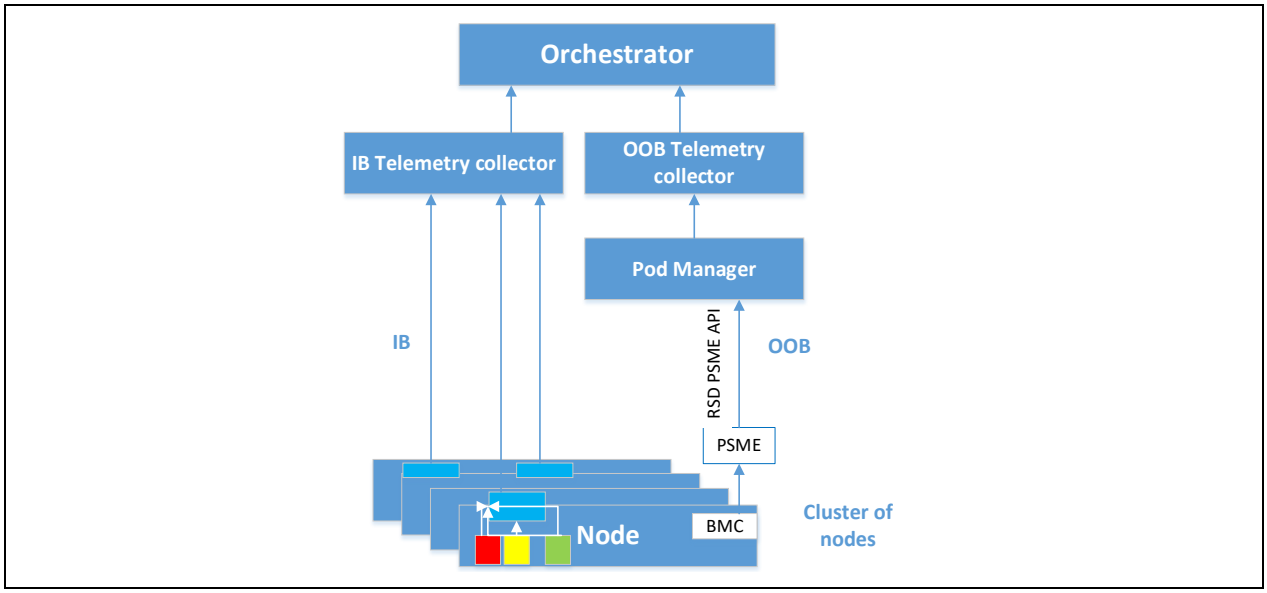
11 Telemetry

The Intel® RSD platform contains various compute, storage and communication elements with varying power and performance characteristics. By monitoring various parameters in these elements and taking appropriate actions, one could achieve optimal performance and lower the TCO.

11.1 Intel® RSD telemetry architecture overview

Intel® RSD Telemetry Architecture allows telemetry to be collected through an Out-of-Band (OOB) interface, through the Intel® RSD PSME API, and through an In-Band (IB) interface as shown in Figure 18.

Figure 18. RSD Telemetry Architecture



RSD exposure of the telemetry is classified under the following categories:

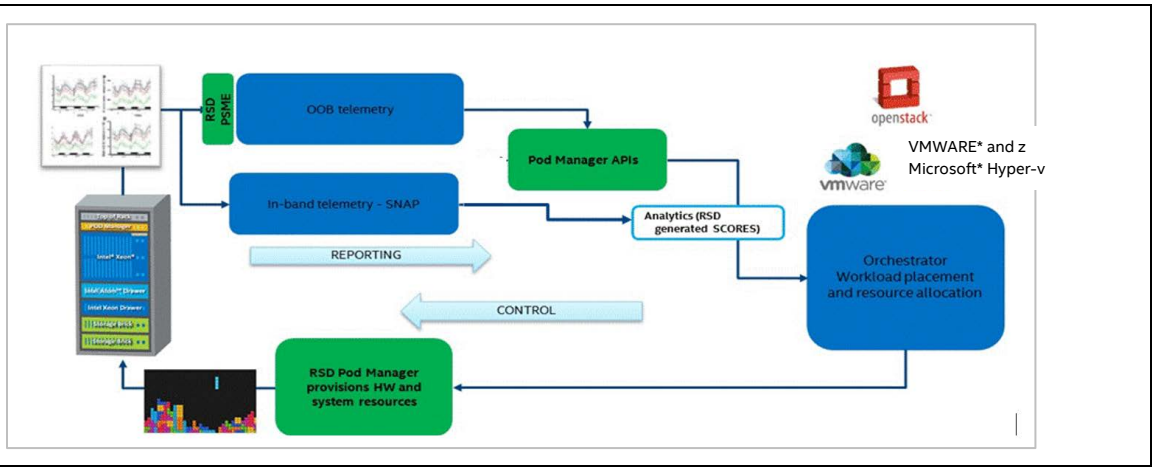
- State
- Event
- Configuration
- Available Capacity
- Performance

Refer to Section 11.3.1 for specific telemetry features supported for each RSD revision.

11.2 Monitoring architecture for Intel® RSD

Figure 19 shows a typical RSD telemetry flow, where the telemetry data is collected from compute, storage, and network devices and provided to the Pod Manager through the RSD PSME API interface for OOB telemetry items and through an in-band interface for the items that are not generally available through the OOB interface.

Figure 19. Typical Telemetry Flow



11.3 Telemetry support requirements for Intel® RSD

11.3.1 PSME API support for telemetry

Required:

The PSME API must support telemetry features listed in Table 9. The PSME communicates with BMC/ME/BIOS to provide the following functionality. Refer to the Intel® RSD v2.2 PSME API Specification for supported telemetry metrics.

Note: Refer to respective platform hardware and firmware feature set for retrieving telemetry data. For example, the Intel® Xeon® Processor Scalable family platform exposes the rich set of telemetry in Intel® Server Platform Services (SPS) FW NM configuration.

Table 9. Intel® RSD Telemetry Support Summary

Component	Sensor/Usage	Required/Recommended	Support starting RSD version
Processor	Health	Required	2.2
	Utilization	Required	2.2
	Average Frequency	Required	2.2
	Throttling	Required	2.2
	Temperature	Required	2.2
	Consumed Power	Required	2.2
Memory	Health	Required	2.2
	Bandwidth	Required	2.2
	Throttling	Required	2.2
	Temperature	Required	2.2
	Consumed Power	Required	2.2
Chassis (SLED) metrics	Inlet Temperature	Recommended	2.2
	Outlet Temperature	Recommended	2.2
Computer System	IO Bandwidth	Required	2.2
PNC	Port Health	Required	2.2
Pooled NVMe drives	Health	Required	2.2

11.3.2 Pod manager SB and NB API support for telemetry

Required:

For the orchestration software to take advantage of the telemetry data, the Pod Manager North Bound (NB) API must support the telemetry APIs. In turn, the Pod Manager must implement the South Bound (SB) telemetry API to get the data from the PSME.

11.3.3 Support for in-band telemetry collection

Recommended:

All telemetry data are not available through the OOB interface. It is recommended the Intel® RSD platforms support the in-band telemetry data collection service for the orchestration to take full advantage of the telemetry data.

11.3.4 Support for correlation of IB and OOB telemetry data

Recommended:

Incorporate IB and OOB metrics and event data collection at the “Telemetry Service” which can be hosted with the Pod Manager service and correlate the in-band data with the OOB data at the RSD controller layer.

§