Date: July 07, 2000

To: T10 Committee (SCSI)

From: George Penokie (IBM)

Subject: Defining Tragets/Initiators as Ports

Overview

In looking over the issue of how SCSI should deal with multiple port devices it appears a reasonable solution would be to change the notion of what a target and initiator is defined as. If you confine a target or initiator to only contain a service delivery port and allow a SCSI device to contain any number of targets or initiators then defining multi-port devices becomes an easier task. In many ways the current definitions and descriptions already support this idea.

What follows are suggested changes to SAM that would define targets and initiators as ports and allow multiples on a single SCSI device.

Multiple port SCSI Device model

The following show the changes to figures in section 4 of SAM-2 that would be required to change to support a port = target/initiator model.

SAM-2 Section 4.4 The SCSI structural model

Figure 1 shows the main functional components of the basic SCSI hierarchy. The following clauses define these components in greater detail.
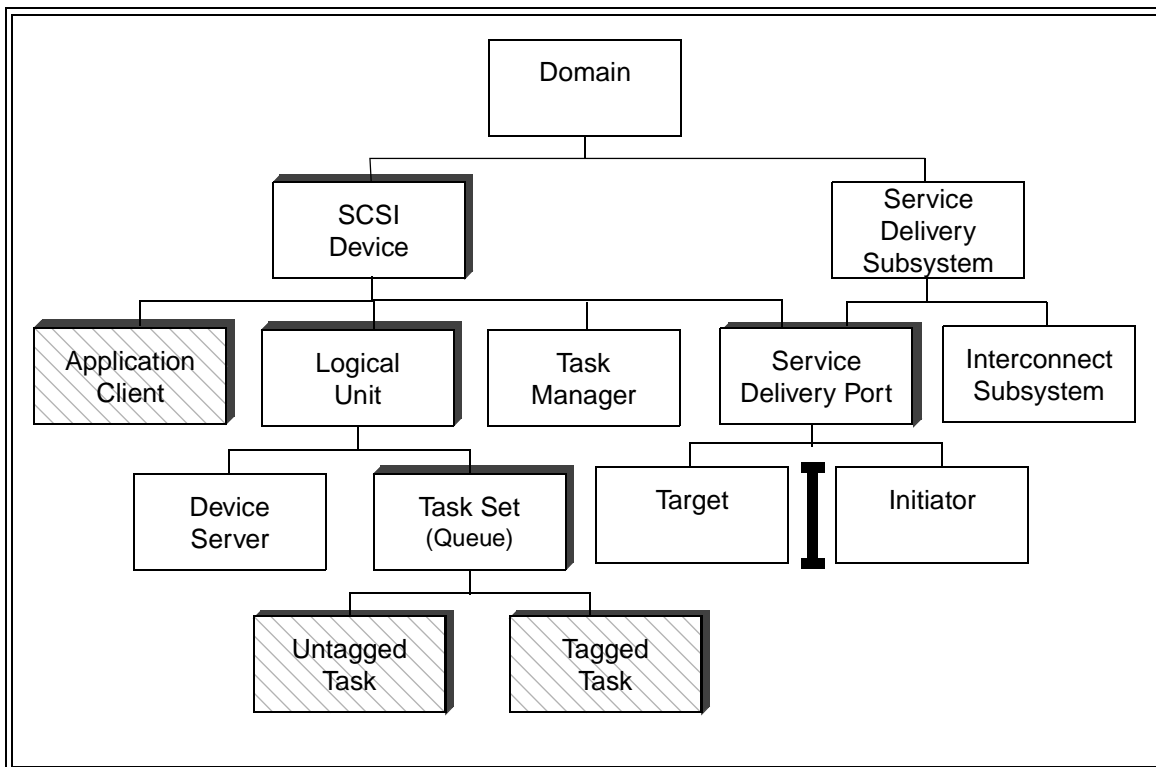

Figure 1 — Basic SCSI hierarchy

SAM-2 Section 4.5 SCSI domain

An example of some of the wording changes that would be required follow. In section 4.5 of SAM-2 the following paragraph would be modified as shown:

A SCSI device is an object that originates or services SCSI commands. As described in , when a SCSI device originat~~esing~~ a command ~~is called~~ that command goes through an initiator; a SCSI device containing logical units that service commands ~~is called~~ receives commands through a target. The service delivery subsystem connects all the SCSI devices in the SCSI domain, providing a subsystem through which application clients and device servers communicate (see 4.6). The boundaries of a SCSI domain are established by the system implementor, within the constraints of a specific SCSI protocol and interconnect standards.

SAM-2 Section 4.7 SCSI Device models

In the basic structural model, an SCSI device (see figure 2) is composed of a service delivery port (see 4.6) combined with an initiator, or a target, or both an initiator and a target. or multiple initiators and targets.
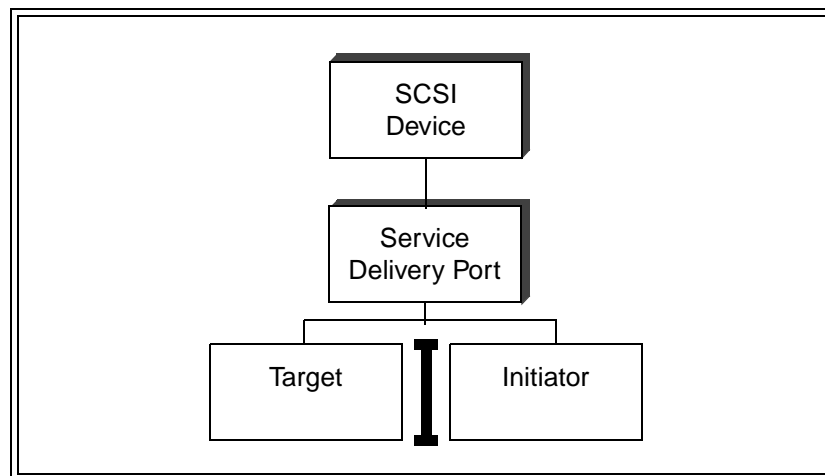

Figure 2 — SCSI Device hierarchy diagram

A SCSI device that contains an initiator ~~is an SCSI Device which~~ is capable of originating SCSI commands and task management requests (see). A SCSI device that contains a target ~~is an SCSI Device which~~ is capable of executing SCSI commands and task management requests (see). To be functional, an SCSI domain needs to contain a SCSI device that contains a target port and an SCSI device that contains an initiator port.

There are ~~three~~ four models for implementing initiator and target ~~capabilities~~ ports in an SCSI device, as shown in figure 3. An SCSI device may ~~perform~~ contain only target ports ~~functions~~, or only initiator ports ~~functions~~, or it may be capable of performing both functions.
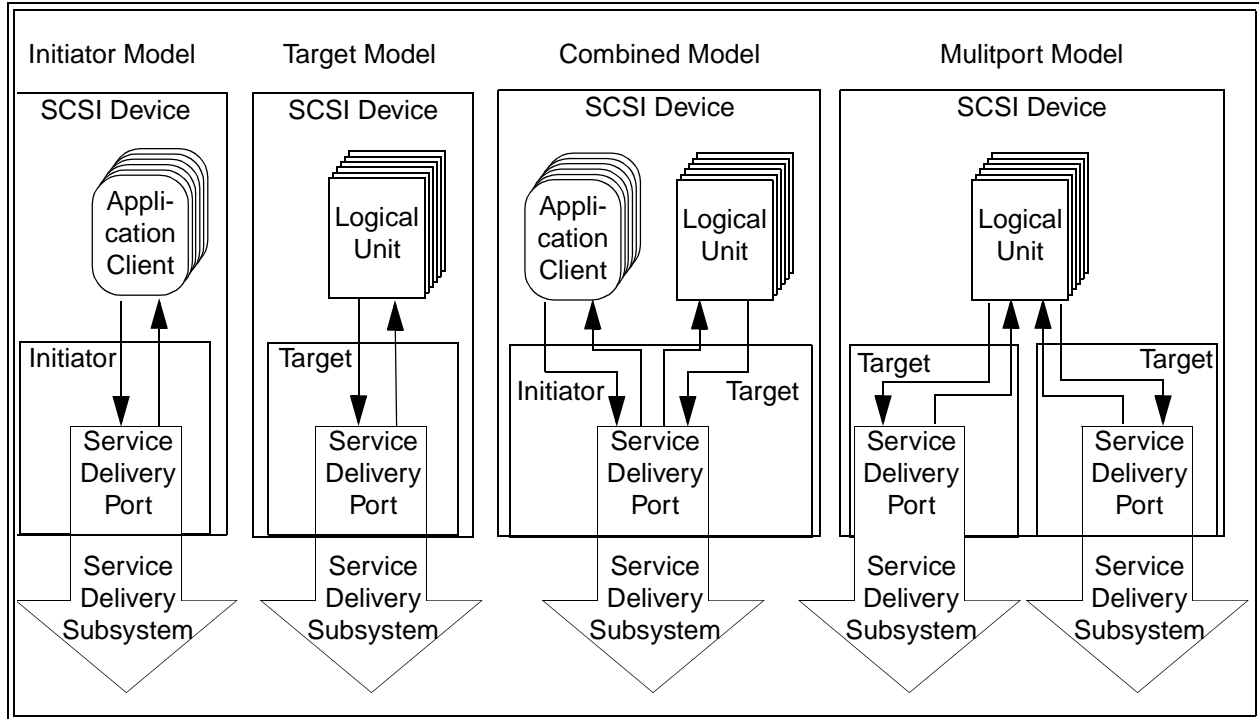
Figure 3 — SCSI device functional models

A SCSI device is referred to by the ~~its~~ role it's service delivery port takes when it participates in an I/O operation. That is, a SCSI device ~~is called~~ contains a target port when it receives SCSI commands or task management functions, or it ~~is called~~ contains an initiator port when it issues SCSI commands or task management requests.

SAM-2 section 4.7.1 SCSI initiator model

A SCSI device with an initiator port (see figure 4) is composed of an Initiator Identifier for each initiator port and zero or more Application Clients.
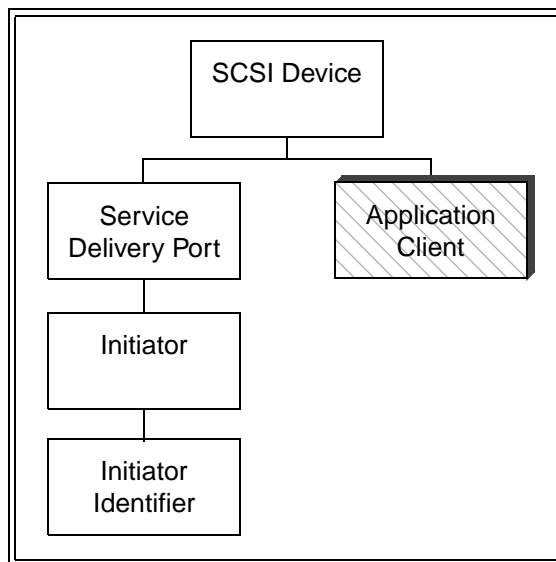

Figure 4 — Initiator hierarchy diagram

An Initiator Identifier is a field containing up to 64 bits that is a SCSI device identifier for the initiator ~~port~~ ~~device~~. An Application Client is the source of commands and task management functions. In this model, there is one application client for each pending command or task management function.

SAM-2 section 4.7.2 SCSI target model

A SCSI device with a target <u>port</u> (see figure 5) is composed of a target identifier <u>for each target port</u>, a task manager, and one or more logical units.
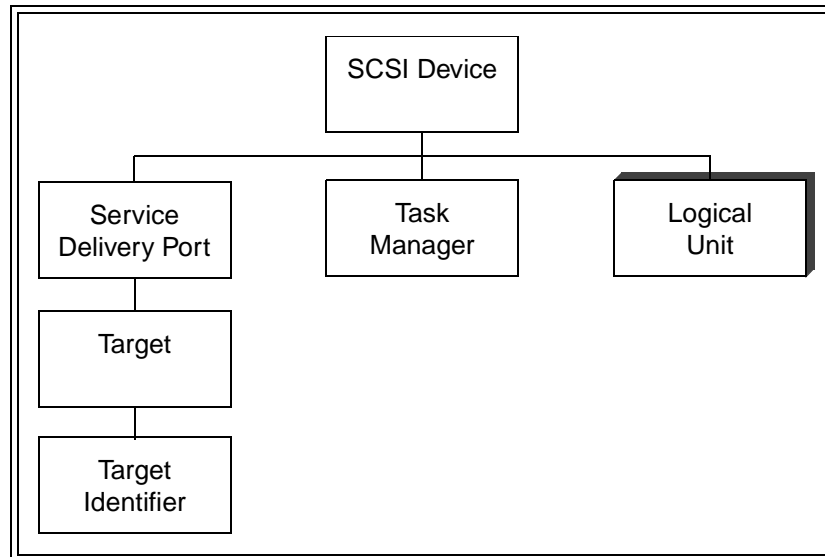

Figure 5 — Target hierarchy diagram

A Target Identifier is a field containing up to 64 bits that is a SCSI device identifier for the target device. A Task Manager is a server that controls one or more tasks in response to task management requests (see 4.7.4). A logical unit (see 4.8) is the object to which SCSI commands are addressed. One of the logical units composing a ~~target~~ <u>SCSI device with target ports</u> shall be addressed using the logical unit number zero.

SAM-2 section 4.9.3 ~~Target~~ identification of <u>target</u> tasks

A ~~target~~ <u>SCSI device</u> identifies a <u>target</u> task with a Task Identifier. The Task Identifier object represents either a Tagged Task Identifier or an Untagged Task Identifier. A Tagged Task Identifier is composed of an Initiator Identifier (see), a Logical Unit Identifier (see 4.8) and a Tag (see 4.9.1). An Untagged Task Identifier is composed of an Initiator Identifier and a Logical Unit Identifier.

If a ~~target~~ <u>SCSI device contains</u> ~~implements the enhanced model for~~ multiple <u>target</u> ports ~~devices~~ (see), then the Task Identifier objects contain one additional object (beyond those mentioned above). For a multiple port <u>SCSI</u> device, a Tagged Task Identifier is composed of a <u>target</u> Port Identifier (see), an Initiator Identifier (see), a Logical Unit Identifier (see 4.8) and a Tag (see 4.9.1). An Untagged Task Identifier is composed of a <u>target</u> Port Identifier, an Initiator Identifier, and a Logical Unit Identifier.

SAM-2 section 4.9.3 ~~Initiator~~ identification of <u>initiator</u> tasks

An ~~initiator~~ <u>SCSI device</u> identifies an <u>initiator</u> task to a target <u>task</u> using a Task Address. The Task Address object represents either a Tagged Task Address or an Untagged Task Address. A Tagged Task Address is composed of a Logical Unit Identifier (see 4.8) and a Tag (see 4.9.1). An Untagged Task Address is composed of a Logical Unit Identifier.

If an ~~initiator~~ <u>SCSI device contains</u> ~~implements the enhanced model for~~ multiple <u>initiator</u> ports ~~devices~~ (see), then it may ~~chose to~~ enhance the definition of a Task Address with <u>initiator</u> port identification information.

SAM-2 section 4.10 Devices With Multiple Service Delivery Ports

~~Optionally, a single device may have multiple Service Delivery Ports (see 4.6), however,~~ The model for ~~such~~ a device with multiple service delivery ports is ~~not~~ a single SCSI Device (see) <u>with</u> ~~but~~ multiple ~~SCSI Devices, one for each~~ Service Delivery Ports. Similarly, a single device may respond to multiple SCSI device identifiers. The model for such a device also is one of multiple SCSI Devices, one for each SCSI device identifier. ~~Devices that have multiple Service Delivery Ports or respond to multiple SCSI device identifiers are called SCSI Multi-port Units.~~

SCSI <u>devices with</u> Multi-port ~~are objects in SCSI architecture model (see figure 6) that~~ do not fit in the two dimensional hierarchy shown in 4.4. Instead, ~~SCSI Multi-port Units~~ <u>they</u> combine basic objects from the SCSI device model in a way that produces a different structural plane. Figure 6 also shows that <u>SCSI devices from different domains may be connected to a single SCSI device with multiple ports</u>. ~~multiple SCSI devices instantiated by an SMU may be constituents of different SCSI Domains~~.
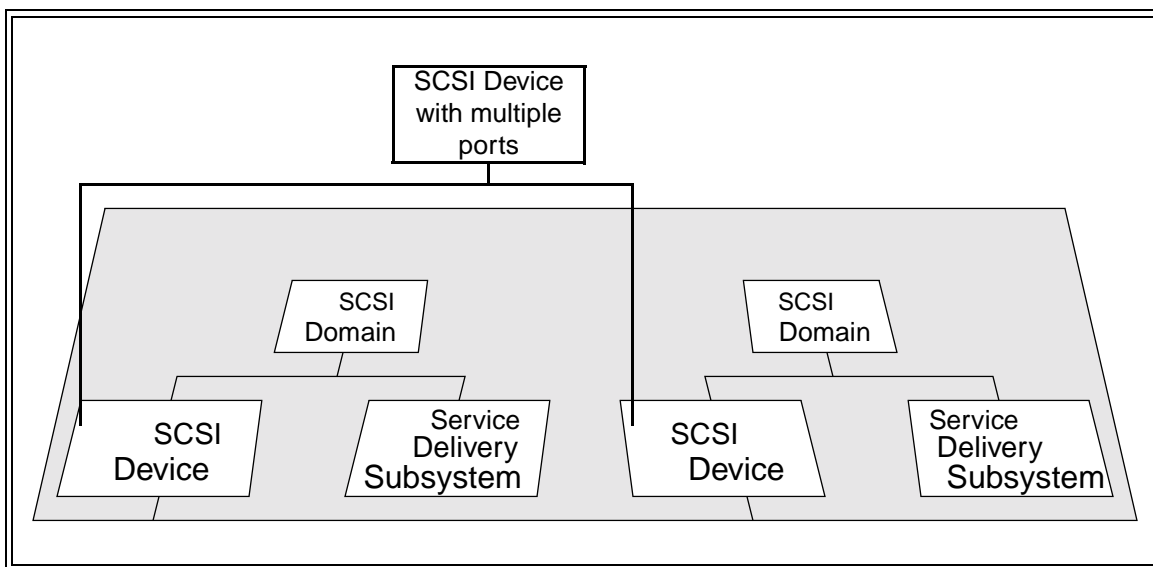

Figure 6 — A SCSI multi-port unit and multiple SCSI domains

The multiple SCSI device identifiers representing ~~an SMU~~ the ports shall meet the requirements for Initiator Identifiers (see) or Target Identifiers (see) or both. ~~The structure of an SMU depends on whether the device is a target or an initiator.~~ <u>SCSI devices</u> with multiple ports ~~units~~ that implement both target and initiator models combine the target and initiator multiple port structures in vendor-specific ways that meet product requirements while maintaining the multi-port model for the target and initiator functions performed by the product. How ~~an SMU~~ <u>multiple port SCSI device</u> is viewed by counterpart <u>SCSI devices</u> ~~initiators or targets~~ in the SCSI domain also depends on whether an initiator <u>port</u> is examining an ~~SMU~~ target <u>port</u> or a target <u>port</u> is servicing an ~~SMU~~ initiator <u>port</u>. The structures and views of SCSI <u>devices</u> ~~multi-port units~~ are asymmetric for target <u>ports</u> and initiator <u>ports</u>. The subclauses that follow discuss the four principle cases described above.

SAM-2 section 4.10.1 ~~SMU~~ target <u>port</u> structure

Figure 7 shows the structure of ~~an SMU~~ <u>SCSI device with multiple</u> target <u>ports</u>. A single task manager and collection of logical units share all the <u>targets</u> ~~SCSI Devices~~ and Service Delivery ports in the <u>SCSI device</u> ~~SMU~~.
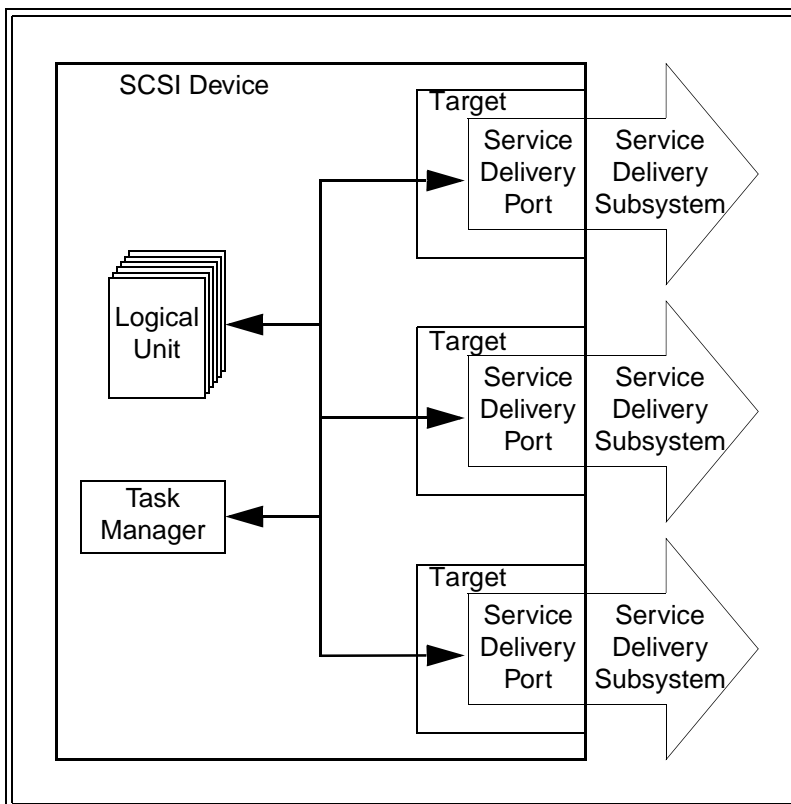

Figure 7 — ~~SMU~~ <u>Multiple</u> target <u>port</u> structure model

Two-way communications shall be possible between all logical units and all service delivery ports, however, communications between any logical unit and any service delivery port may be inactive sometimes. Two-way communications shall be available between the task manager and all service delivery ports. Each service delivery port shall accept commands addressed to LUN 0 and shall forward them to a device server for processing. The REPORT LUNS commands (see SPC-2) shall be accepted by logical unit 0 from any service delivery port and shall return the logical unit inventory available via that service delivery port. The availability of a the same logical unit through multiple delivery ports is discovered by matching Device Identifier values in the INQUIRY command Vital Product Data page (see SPC-2).

SAM-2 section 4.10.2 ~~SMU~~ <u>Port</u> Task Identifiers

In addition to the Task Identifier constituent objects described in , ~~SMU~~ Target <u>ports</u> shall provide a Port Identifier object to contribute to the construction of Task Identifiers. The exact nature of the Port Identifier is vendor specific. The complete contents of the Task Identifier object, including the Port Identifier, is defined in .

SAM-2 section 4.10.3 ~~SMU~~ initiator <u>port</u> structure

Figure 8 shows the structure of ~~an SMU~~ <u>SCSI device with multiple</u> initiator <u>ports</u>. A collection of Application Clients share all the ~~SCSI Devices~~ <u>initiator ports</u> and Service Delivery ports in the SMU.
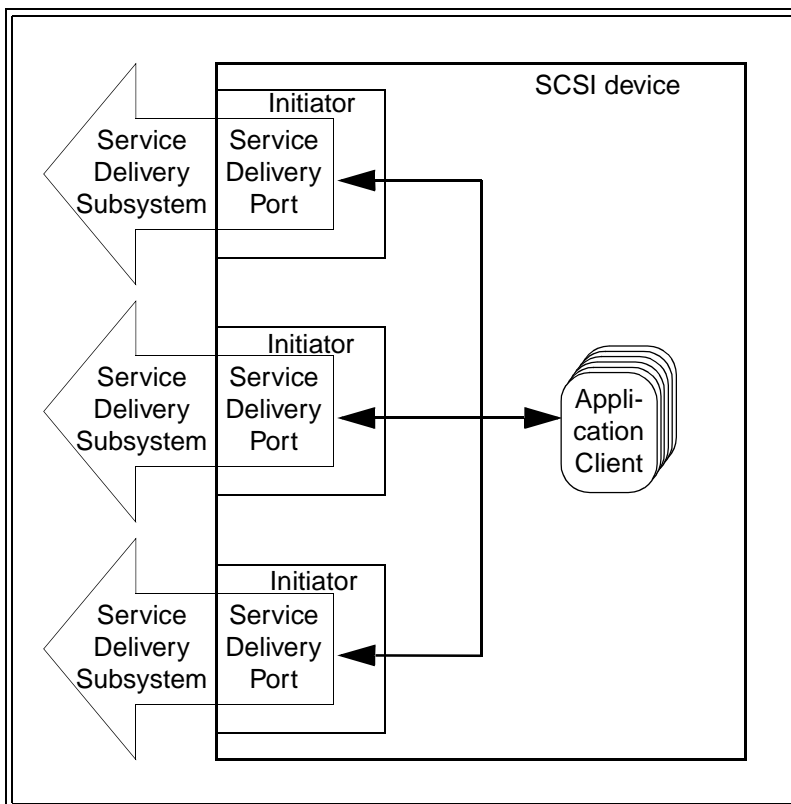

Figure 8 — SMU initiator structure model

Two-way communications shall be possible between all application clients and all service delivery ports, however, communications between any logical and any service delivery port may be inactive sometimes. Mechanisms by which a <u>SCSI device</u> <u>with</u> target <u>ports</u> would have the ability to discover that it is communicating with an ~~SMU~~ initiator <u>port</u> are beyond the scope of any standards in the SCSI family of standards.

SAM-2 section 4.10.4 Initiator view of a ~~SMU~~ <u>SCSI device with multiple target ports</u>

A ~~SMU target~~ <u>SCSI device</u> may be connected to multiple independent service delivery subsystems <u>through target ports</u> in such a way that no single initiator <u>port</u> can communicate with a logical unit or task manager using two or more of the service delivery ports in the <u>SCSI device</u> ~~SMU~~. In this case, the application clients in the initiator <u>ports</u> <u>cannot determine if a SCSI device has multiple ports</u>. ~~distinguish the SMU from a basic SCSI Device~~.

However, SCSI <u>devices with multiple ports</u> ~~Multi-port Devices~~ may be configured in many different ways where application clients have the ability to discover that one or more logical units are accessible via multiple service delivery ports. Figure 9 and figure 10 show two examples of such configurations.

Figure 9 shows ~~an SMU~~ <u>SCSI device with multiple</u> target <u>ports</u> participating in a single SCSI Domain with two SCSI initiator devices. There are <u>three SCSI devices, one of which has</u> ~~four service delivery ports in this configuration,~~ two <u>target ports</u> ~~for the SMU target,~~ <u>one with one initiator port,</u> <u>and one with one initiator port.</u> ~~one initiator port each for the two initiators.~~ There are two Target Identifiers and two Initiator Identifiers in this SCSI Domain. Using the INQUIRY command Vital Product Data page as described in , the application clients in each of the <u>SCSI devices that contain</u> initiator <u>ports</u> have the ability to discover that logical

units in the ~~SMU~~ <u>SCSI devices that contain</u> target <u>ports</u> are accessible via multiple Target Identifiers (service delivery ports) and map the configuration of the SCSI devices ~~target~~.
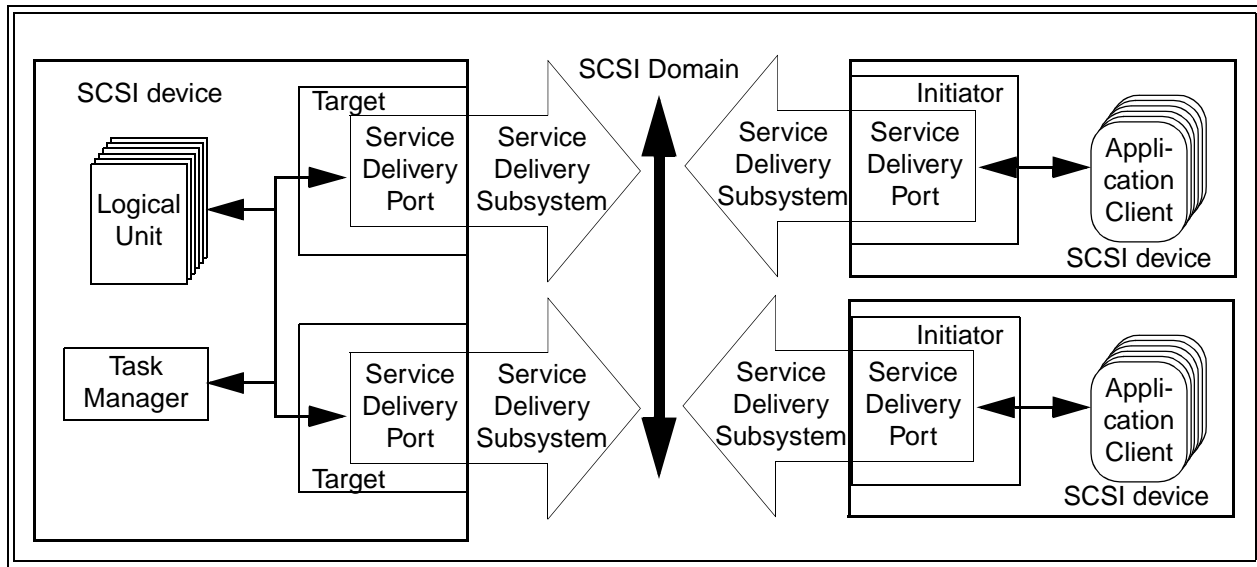

Figure 9 — ~~SMU target~~ <u>SCSI device with target ports</u> configured in a single SCSI Domain

Figure 10 shows <u>a SCSI device with multiple target ports</u> ~~an SMU target~~ participating in two SCSI Domains and <u>a SCSI device with multiple initiator ports</u> ~~an SMU initiator~~ participating in the same two SCSI Domains. <u>There are two SCSI devices, one with two target ports and the other with two initiator ports.</u> ~~four service delivery ports in this configuration, two for the SMU target and two for the SMU initiator~~. There is one Target Identifier and one Initiator Identifier in each of the two SCSI Domains. Using the INQUIRY command Vital Product Data page as described in , the application clients <u>in the SCSI device containing initiator ports</u> ~~SMU initiator~~ have the ability to discover that logical units in the <u>SCSI device containing target ports</u> ~~SMU target~~ are accessible via multiple service delivery ports ~~in the SMU initiator~~ and map the configuration. However, the methods available to application clients to distinguish between the configuration shown in figure 10 and the configuration shown in figure 22 may be beyond the scope of the SCSI family of standards.
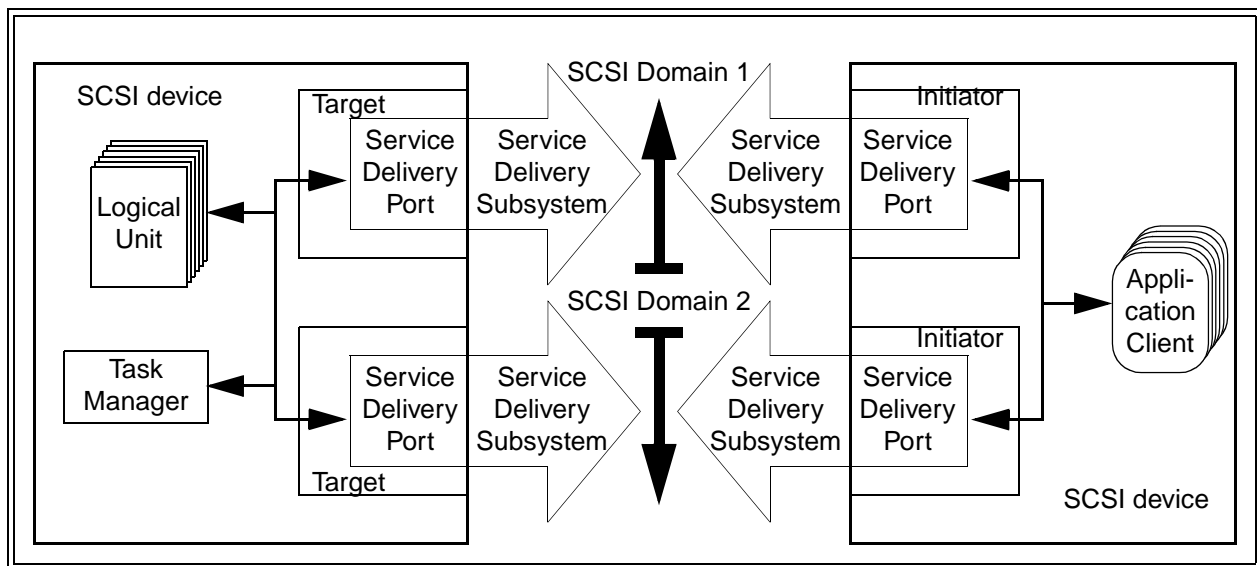

Figure 10 — ~~SMU target~~ <u>SCSI device with target ports</u> configured in multiple SCSI Domains

Figure 11 shows the same configuration as figure 10 except that the two SCSI Domains have been replaced by a single SCSI Domain.
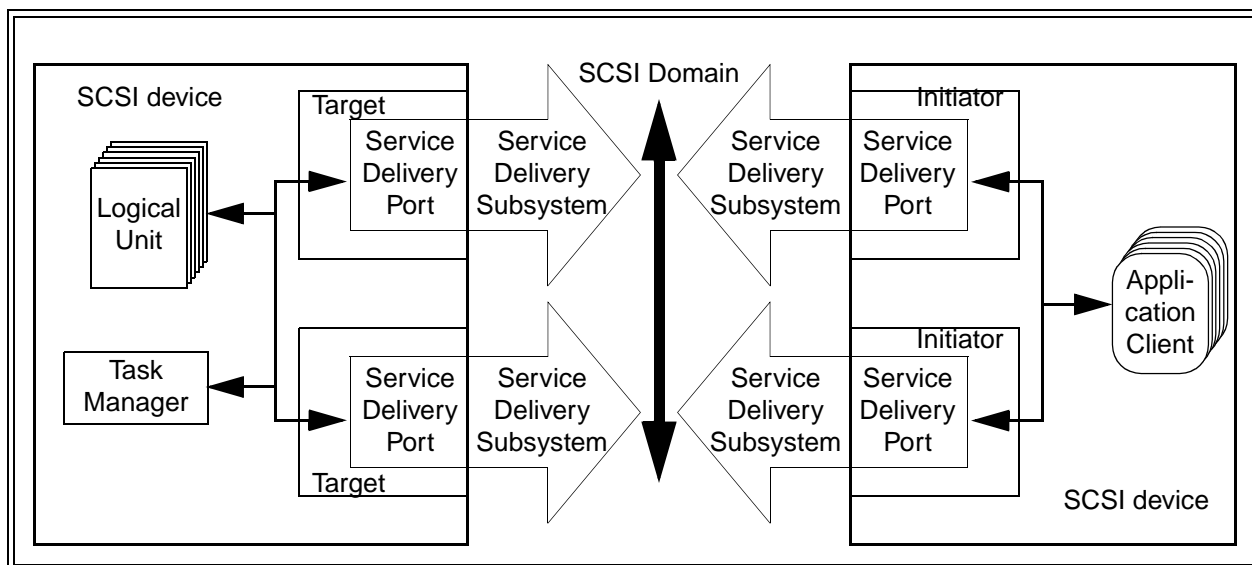

Figure 11 — ~~SMU target~~ <u>SCSI device with target ports</u> and ~~SMU initiator~~ <u>SCSI device with initiator ports</u> configured in a single SCSI Domain

This model for application client determination of SMU target configurations relies on information that is available only to the application clients via SCSI commands. The service delivery ports in the initiators (figure 9) or initiator (figure 10 and figure 11) are unable to distinguish the multiple service delivery ports in the SMU from individual service delivery ports in two separate SCSI Devices.

SAM-2 section 4.10.5 Target view of an SMU initiator

A SCSI ~~target~~ device <u>with target ports</u> ~~or SMU target~~ does not have the ability to detect the presence of a <u>SCSI device with initiator ports</u> ~~an SMU initiator~~. Therefore, <u>a SCSI device with</u> target <u>ports</u> handles a <u>SCSI device with multiple initiator ports</u> ~~an SMU initiator~~ exactly as it would handle multiple separate <u>SCSI devices each with a single initiator port.</u> ~~initiator devices.~~ For example, <u>a SCSI device with target ports</u> ~~SCSI target~~ handles the configurations shown in figure 10 and figure 11 in exactly the same way it handles the configuration show in figure 9.

> NOTE 1 The implications of this view of an ~~SMU~~ initiator are more far reaching than are immediately apparent. For example, if an ~~SMU~~ <u>SCSI device with</u> initiator <u>ports</u> makes an exclusive access reservation via one service delivery port, then access will be denied to the other service delivery port(s) on that same <u>SCSI device</u> ~~SMU initiator~~.

SAM-2 section 4.10.6 SMU considerations for task management functions

Although there is only one task manager for all logical units and all service delivery ports in a <u>SCSI device</u> ~~n SMU~~, the task manager in <u>a SCSI device</u> ~~an SMU~~ shall observe the requirements described in the following clauses in addition to the requirements placed on a task set manager by the SCSI architecture basic structural model.