Changes are proposed to make the SAM-3 glossary more accurate and useful.

Revision History

r0 Initial proposal

Specific Changes

All proposed changes reference SAM-3 r04.

Change 1 ['service']: Like an 'object', a 'service' is an architectural abstraction. Following the model of the former:

3.1.71 object: An architectural abstraction or container that encapsulates data types, services, or other objects that are related in some way.

Modify the latter as follows:

3.1.114 service: Any architectural abstraction for an operation or function performed by a SCSI object that is invoked by other SCSI objects.

Change 2 ['service' should be 'process']: In a few instances, the word 'service' is used in a manner that is not consistent with the glossary entry. A suitable replacement for 'service' is 'process' and such a replacement fits traditional SCSI usage.

In 4.5 (SCSI domain), modify the following as shown.

A SCSI device is an object that originates or services processes SCSI commands. … A SCSI device containing logical units that service process commands is called a SCSI target device and receives commands through a SCSI target port or a SCSI target/initiator port.

In 5.1 (The Execute Command remote procedure), modify the following as shown:

Data-Out Buffer: A buffer containing command specific information to be sent to the logical unit, such as data or parameter lists needed to service process the command. …

In 5.4.2 (Execute Command request/confirmation SCSI transport protocol services), modify the following as shown:

Data-Out Buffer: A buffer containing command specific information to be sent to the logical unit, such as data or parameter lists needed to service process the command (see 5.1). …
Change 3 ['service delivery port']: The SAM-2 changes to support SCSI devices with multiple ports eliminated the concept of a 'service delivery port', but one instance of usage of that term remains. In 4.6.2 (Synchronizing client and server states), modify the last paragraph as shown.

The model assumes that state synchronization, if required by a SCSI transport protocol standard, is enforced by the service delivery subsystem transparently to the server (i.e., whenever the server invokes a SCSI transport protocol service to return a response as described in 7.10 and 5.4. It it is assumed that the SCSI port service delivery port for such a SCSI transport protocol does not return control to the server until the response has been successfully delivered to the SCSI initiator device).

Change 4 [the 'call' architectural abstraction]: The terms 'call', 'procedure call', and 'remote procedure call' are used interchangeably and only 'call' is defined in the glossary. The following changes are needed to cleanup this hodge-podge.

Modify the glossary as follows:

3.1.9 3.1.x procedure call: An architectural abstraction having the appearance of a programming language function call that is used to model service interfaces. The act of invoking a procedure.

In 4.2 (The SCSI distributed service model), make the following modifications:

Client-server relationships are not symmetrical. A client may only originate requests for service. A server may only respond to such requests. The client calls the server-resident procedure and waits for completion. From the client's point of view, the behavior of a remote service invoked in this manner is indistinguishable from a conventional local procedure call. …

In 5.1 (The Execute Command remote procedure), make the following modifications:

5.1 The Execute Command remote procedure call

An application client requests the processing of a SCSI command by invoking the SCSI transport protocol services described in 5.4, the collective operation of which is conceptually modeled in the following remote procedure call:

Change 'remote procedure call' to 'procedure call' in the following subclauses:

- 4.2 (The SCSI distributed service model)
- 4.6.1 (The service delivery subsystem object)
- 5.1 (The Execute Command remote procedure)
- 5.4.1 (SCSI transport protocol services in support of Execute Command overview) [twice]
- 5.4.3.1 (Data transfer SCSI transport protocol services introduction)
- 5.8.1 (Unlinked command example)
- 5.8.2 (Linked command example) [twice]
- 6.3.2 (Hard reset)
- 6.3.4 (L_T nexus loss)
- 7.10 (Task management SCSI transport protocol services)

Change 'call' to 'procedure call' in the following subclauses:

- 4.14 (The SCSI model for distributed communications)
- 5.4.2 (Execute Command request/confirmation SCSI transport protocol service) [unless 03-002 is approved before this proposal]
To both make the usage of 'procedure call' consistent and have a common nomenclature between clauses 5 and 7, change 'Function call:' to 'Request' in the following subclauses:

- 7.2 (ABORT TASK)
- 7.3 (ABORT TASK SET)
- 7.4 (CLEAR ACA)
- 7.5 (CLEAR TASK SET)
- 7.6 (LOGICAL UNIT RESET)
- 7.7 (QUERY TASK)
- 7.8 (TARGET RESET)
- 7.9 (WAKEUP)

**Change 5 [unused glossary entries]:** Remove the following glossary entries for terms that are no longer used in SAM-3:

- **3.1.32 ended command:** A command that has completed or aborted.
- **3.1.49 implementation option:** An option whose actualization within an implementation is at the discretion of the implementor.
- **3.1.65 logical unit option:** An option pertaining to a logical unit, whose actualization is at the discretion of the logical unit implementor.
- **3.1.72 peer-to-peer protocol service:** A service used by an upper level protocol implementation to exchange information with its peer.
- **3.1.81 protocol option:** A function whose definition within a SCSI transport protocol standard is optional.