SCSI-3 Fault Tolerant Controller Configurations

utilizing SCC & New Event Codes

Editor:

Steve Sicola

Sicola@Peaks.enet.dec.com

High Availability Study Group

Document No.: X3T10 95-312r3

Rev 3.0

February 28, 1996

Overview

This document contains the background information about fault tolerant controller configurations, and the basic features and assumptions about fault tolerant controller configurations. This will be used as a baseline for issues pertaining to volume set /LUN relationships and how the LUNs are accessed by host across multiple paths under normal or post-failure conditions. Issues concerning interoperability in open systems environments are described that lead to slight changes to SCC and SCSI-3 in general. The specific modifications to the SCC specification as well as the addition of two new ASC/ASCq's to solve the problem of multiple path volume set/LUN access follows, after which a functional description is presented for the use of SCC and SCSI-3 with fault tolerant (multi-access path) controller configurations.

The basic concept is for controllers configured together to be considered 'attached' components in the SCC model. These attached controllers can then bind volume sets to LUNs with specific LUN access profiles specified for attached host to access LUNs with. The status of attachments and bound/unbound volume sets (or 'mounted' volume sets for another description) is available to hosts upon request. The LUN access profile specifies the controller (s) that the hosts may actively access LUNs through as well as which controller(s) (attached to each other) may serve as alternate paths to the LUN in the event of a current access path failure. The concept of 'mounted' refers to a volume set bound to a LUN. The concept of 'offline' refers to a volume set for which there is no LUN bound to it.

Fault Tolerant Controller Configurations

Architectural Concepts

Fault Tolerant Controller Configurations are defined as any two or more controllers sharing access paths to a set of devices. This broad definition has many possible host-controller configurations with respect to LUN ownership & access. Although each controller in a fault tolerant configuration shares access paths to the same devices as its partner(s), it does NOT mean that each controller has 'ownership' of every LUN configured on the access paths. Furthermore, controller s in a fault tolerant configuration may or may not both be actively servicing host requests at the same time (to different or same LUN(s)).

The concept of LUN ownership & access in a Fault Tolerant Controller configuration depends on the relationship between the controllers and their subsequent relationship with connected host(s). The

relationship between controllers and LUN ownership depends on the physical configuration of controllers in a fault tolerant configuration.

Typically, controllers are paired with another controller in a dual configuration. This configuration requires that both controllers 'know' about LUNs configured for service through the pair. This could mean 'all' attached LUNs on shared access paths OR it could mean some logical subset for reasons of access path sharing with other controllers or direct attached hosts.

Controllers in a configuration can be providing LUN service to host(s) simultaneously, called active-active, or in an active-standby with respect to LUN service. An active-active configuration allows any host to use either controller for a LUN access (load balancing) and fault tolerance. An active-standby configuration allow for only fault tolerance.

Concurrent access by more than one controller to the same LUN is allowed, but typically not supported by industry operating systems because of software interlock issues. Most high availability software in hosts today use the SCSI exclusive access commands Reserve and Release to interlock access from different hosts, but do so typically through the same controller for the same LUN, although it is possible to change paths. Typical operating systems access a LUN behind a single controller until and if that controller fails, except for potential load balance optimizations where the operating system can move service from one controller another for performance reasons. These operating systems do use the SCSI Reserve and Release commands to affect these access path changes.

Some controllers may artificially enforce this ownership model by only responding with a ready LUN on the controller that is 'preferred' to handle the device unless or if a controller failure occurs. This is a further refinement to the definition of LUN ownership. This artificial method of enforcing access path through one controller in a fault tolerant configuration overrides any attempts to use normal SCSI reserve/release mechanisms, but is used widely by controller manufacturers , because it simplifies the complex operation of managing a fault tolerant controller configuration within partner controllers.

Fault tolerant configurations that contain more than two controllers can be supported only with added intelligence in a host computer or with multiple pairs of controllers basically in separate configurations as viewed to the host, but with shared access paths to devices/LUNs. Hosts with more intelligence in large cluster environments can take advantage of a large fault tolerant controller configuration by performing controlled load balancing and knowledge of a the entire controller configuration(s) of attached controllers. The load balancing and wealth of knowledge lead to more controlled failover scenarios with potential for availability /performance during degraded situations, because of the shared access paths from all controllers.

Fault tolerant controller configurations provide a redundant path to LUNs and devices in the event of failure. Failover is defined as the event in which a surviving controller takes over service responsibilities for a failed partner in the fault tolerant controller configuration. Failback is the event in which a controller returns to the fault tolerant configuration after re-initialization or replacement after which that controller can accept service requests for LUN accesses.

Failure detection is achieved on of two ways. The first method of detection is host-based, where the hosts detect the failure. Hosts can detect the failure in one of number of methods, from command time-outs to periodic polling of each controller in the configuration. The time frame for host based controller failure detection is based upon the setting of specific command time-outs or periodic polling intervals. These times are typically vendor unique and can range from seconds to minutes.

The second failure detection mechanism lies with the controller configuration itself. The controllers have the opportunity for detection based upon the controller design or by a similar, lower level periodic polling or 'heartbeat' between controllers in the configuration.

The Problem

The problem with today's implementations of fault tolerant controller configurations is that the configuration of and reporting on the configurations is done differently by every controller vendor.

Open System Clustered environments, containing large number of host and storage subsystems have potentially larger problems managing the flow of information between several host -storage interconnects, both for load balancing and fault tolerant purposes. It is vital that LUNs be distinguishable from each other uniquely, for purposes of multiple paths to LUNs through different control units. Different methods for different vendors will cause problems in managing large open systems. Furthermore, the failure detection from hosts and controllers is handled differently by host operating systems and controllers as well. These problems pose serious issues for interoperability in the open systems environment.

In order to achieve interoperability in the open systems environment, standardization of the creation of and reporting on LUNs and access paths for fault tolerant controller configurations is required. The issue of configuration simplicity as well configuration check simplicity is desired for open system operating system driver development. The ability to manage LUN access methods for differing requirements of open system environments is also required. Furthermore, the failure detection mechanism should be standardized to be used optionally for quicker failover/failback in highly available system configurations. The generalization to any number of controllers in a configuration leads to more possibilities for load balancing in large host/controller environments.

The standard for creation of and reporting on fault tolerant controller configurations is within the scope of SCC. SCC is the set of commands to manage configuration and control operations for storage controllers. The standard for failure detection mechanism is simply two new ASC/ASCq's in SCSI-3 that can be used to identify the failover/failback event.

SCC & SCSI-3

The SCC specification defines a model comprised of a single controller (SACL) with one or more controller components (among others). Fault Tolerant Controller configurations that are in some cases in industry are non-compliant with the SCC model. These configurations can become compliant based upon assumptions about the configuration. In fact, only one assumption actually changes how most controllers in the industry would achieve compliance. The others are basic assumptions about controller configurations that are not specified by SCC, but are present nonetheless in every fault tolerant controller configuration.

The assumptions about fault tolerant controller configurations under the SCC model are:

1. Two or more controllers sharing access paths to storage devices. The SACL within each controller in a fault tolerant configuration is logically presented as a single SACL with multiple ports. Specifically, the two controllers must present the exact same configuration (for LUNs and configured containers/devices) when SCC 'Report' commands are presented to either controller. **LUNs are uniform across a configuration, i.e. LUN0 behind one controller is always LUN0 to any other controller in the configuration, regardless if it is configured for service on all attached controllers.** This is key assumption for compliance. The controllers must report in a standard way from any controller in the configuration. In SCC, that method is through the Controller Base Address (LUN0) on each controller. If a fault tolerant configuration comprising more than two controllers exists, then the configuration may contain one or more SACL's depending on the model of LUN ownership & access desired. Two controllers share at least fault tolerant 'ownership' of a set of LUN(s), but may or may not share access 'ownership' for host service.

   Two specific examples will show the range of possibilities with respect to fault tolerant controller configurations. These two examples are called Type 1 and Type 2 configurations respectively:

   a. Type 1 Configuration: This configuration is where 'n' controllers are in the fault tolerant configuration AND that ALL attached LUNs are shared between the 'n' controllers. The sharing is for fault tolerant purposes as well as access purposes. One or more controllers can

take over for failed partners. Access to LUN(s) behind controllers in this configuration may be host controlled (via reserve/release) or controller controlled (prefer path from a specific controller per LUN).

For example, three controllers with LUNs 1,2,3,4 configured. Controller A, B, & C can assume access 'ownership' if any partner fails while serving a specific LUN. Initial LUN 'access' ownership is setup at configuration time for the entire configuration.

b. Type 2 Configuration: This configuration represents a 'sub-component' attachment method, where controllers in a fault tolerant configuration are attached to specific LUNs between selected set(>=2) of controllers in the configuration. Any set of controllers in the configuration can be attached to any individual LUN or LUNs. When a LUN is assigned to the set of controllers, the 'ownership' of the LUN is set as well, for fault tolerance and access control. Any controller can assume 'ownership' after a failure. Access ownership may be host controlled (via Reserve & Release) or controller controlled (via preferred access path from a specific controller per LUN).

This configuration type allows for example a set of 3 controllers where controller A and controller B are attached to LUN1 & LUN3, while controller B and controller C are attached to LUN2 & LUN4 for fault tolerant purposes. The access ownership can be set to host or controller control for each LUN.

2. Controllers in a fault tolerant configuration communicate with each other to relate changes in state or configuration. The mechanism by which controllers communicate with each other while in a fault tolerant configuration is outside the scope of this document. Most controllers today communicate directly or indirectly about changes in state or configuration. The communication may take place directly via a communication path between each controller in which the controllers actively communication changes. The communication may take place indirectly through stored changes on attached devices. This assumption allow assumption (1) validity.
4. Controllers will include those with single or multiple host interfaces, and single or multiple shared device interfaces.
5. Controller may be pre-configured or configured from the attached host as a fault tolerant configuration. Configurations are verified during controller initialization as well as after initial configuration.
6. Any/all surviving controller within the configuration can assume the service of storage from the failed controller. This can be a controlled event, with specific notions of which controller will failover for another OR it may be random.

In order for controllers to be in a fault tolerant configuration, these assumptions are logical conclusions so that hosts can easily configure and identify the fault tolerant controller configurations. However, there are several commands in the current SCC specification that do not specify how the creation of and reporting on fault tolerant controller configurations is achieved.

SCC & SCSI-3 Changes

The proposed SCC changes include new commands and modifications to the existing SCC commands. They include:

Attach to Component Device - Optional

Report Component Device Attachments - Mandatory

Mount Volume Set - Optional

Dismount Volume Set - Optional

Report Volume Set Mount Status

The Attach to Component Device command (when addressing the controller device) is proposed to have the following format, which is exactly the same format today, with the exception of denoting the LUN_C field specifically:

Table 1 - ATTACH COMPONENT DEVICE service actions

Bit / Byte	7	6	5	4	3	2	1	0
0	OPERATION CODE (A4h)							
1	RESERVED			SERVICE ACTION (01h)				
2	RESERVED							
3	RESERVED							
4	(MSB) LUN_C=0							
5	LUN_C=0							(LSB)
6	(MSB) LIST_LENGTH =							
7	Number of Controllers to							
8	Create an Attachment With *8							
9								(LSB)
10	RESERVED							
11	CONTROL							

When an ATTACH TO COMPONENT DEVICE service action is received with LUN_C=0, this designates an action to attach controller components together in a fault tolerant controller configuration, sharing access paths to configured devices within this fault tolerant configuration. All configuration state is mutual between attached controllers.

The LUN_C field of 0 specifies that the controllers named in the parameter list shall be attached together into the controller configuration.

When a new controller is to be added to a configuration, or when a controller is to be deleted from a configuration, the ATTACH COMPONENT DEVICE service action shall be employed with a new list of controllers. Any controller that has failed, is still part of the configuration until replaced, thereafter requiring another ATTACH COMPONENT DEVICE service action, some type of automatic replacement by the controller, or a manual intervention to one controller in the configuration to update the configuration.

Table 2 - ATTACH COMPONENT DEVICE (LUN_C=0) parameter List

Bit / Byte	7	6	5	4	3	2	1	0
0	Controller Component World Wide Name							
	Controller Component World Wide Name 1							
7								
...								
n-7	Controller Component Name x = n/8							
n								

The parameter list contains a set of Controller Component World Wide names, each 8 bytes in length

The result of this command, if successful, will be the controllers specified being attached. A name of the resulting configuration will be determined by the controller receiving the ATTACH TO COMPONENT DEVICE COMMAND. This name shall be reported when the REPORT COMPONENT DEVICE ATTACHMENTS service action is invoked.

The reporting of controller attachments and therefore the fault tolerant configuration is achieved with the REPORT COMPONENT DEVICE ATTACHMENTS service action. The command received by a controller with the LUN_C field of 0h will report specific information about the attachments that this controller has in effect with other controllers, the name of this attachment, and the potential controllers that could be attached. This implies the controllers can communicate, otherwise the attachment would not be possible.

The REPORT COMPONENT DEVICE ATTACHMENTS service action requires the following changes:

Table 2 - REPORT COMPONENT DEVICE ATTACHMENTS (LUN_C=0) Service Action

Bit Byte	7	6	5	4	3	2	1	0
0	OPERATION CODE (A3h)							
1	RESERVED			SERVICE ACTION (02h)				
2	RESERVED							
3	RESERVED							
4	(MSB)	LUN_C=0						
5								(LSB)
6	(MSB)							
7	Allocation Length							
8								
9								(LSB)
10	RESERVED							RPTSEL
11	CONTROL							

When the LUN_C field is zero, then the rest of the data in the Report Component Attachment command are ignored. The controller being queried will report:

Bit Byte	7	6	5	4	3	2	1	0
	Controller Component Name							
0	(MSB)	Controller Attachment Name (CAN)						
7								(LSB)
8	Number of Controllers in Attachment (x)							
9	(MSB)	Controller 1 World Wide Name						
17								(LSB)
...								
z=17+ x*8	Controller x World Wide Name							(LSB)

The parameter list contains the name of the attachment as well as the world wide names of the controllers currently part of the attachment.

The unique identification of a Controller Configuration is required. It does not need to be a World-wide type name, rather a ‘system-wide’ unique name that can cover the configuration across the hosts to which it is attached. This name will survive any controller failures and replacements, so as to not rely upon the serial number of the controller or any packaging specific addresses. Hosts will ‘know’ about the Controller configuration by name, simplifying any mapping of access paths to devices and fault tolerance in general. The model used for the Controller Attachment is that of the FibreChannel model, utilizing an 8 bit Vendor Unique field as the most significant byte and the last 7 bytes assigned by the controller during configuration.

Mount Volume Set

The Mount Volume Set command binds a volume set to a LUN. It also specifies the LUN access profile to be used when accessing this LUN and the controller(s) involved in the LUN access. The controllers involved in LUN access are the primary path(s) and secondary path(s).

Bit Byte	7	6	5	4	3	2	1	0
0	OPERATION CODE (TBD)							
1	RESERVED			SERVICE ACTION (TBD)				
2	RESERVED							
3	RESERVED							
4	(MSB)			LUN_C=0				
5								(LSB)
6	(MSB)							
7	Allocation Length							
8								
9								(LSB)
10	RESERVED							RPTSEL
11	CONTROL							

The parameter list includes:

Bit Byte	7	6	5	4	3	2	1	0
0	Volume ID byte0							
7	Volume ID byte7							
8	LUN byte 0							
15	LUN byte 7							
16	RES | LUN Access Profile Bit Mask							
17	Controller Count(CC)							
18	Controller 0 WWN byte 0							
25	Controller 0 WWN byte 7							
25+ CC*8	.... Controller CC-1 byte 7							

The 8 byte Volume ID is bound to the 8 byte LUN with the Mount command

Byte 16 specifies the LUN Access Profile Bit Mask

Bit

0 - Controller 0 in list of controller is the ONLY controller that will allow access from hosts to this LUN

1 - Controller 0 in list of controllers is the PRIMARY access path for this LUN from hosts. Controller 1 through CC-1 are Secondary access paths to be used singly after a primary failure.

2 - Any Controller (0 through CC-1) may act as a primary path, with the rest acting as secondary access paths if the primary fails.

3 - Any Controller (0 through CC-1) may act as primary concurrently, allowing access to this LUN through multiple access paths at once.

4:7 - Reserved for future use.

Dismount Volume Set

The Dismount Volume Set unbinds a volume set from a LUN, thereby removing the capability for that volume set to be accessed from a host.

Bit Byte	7	6	5	4	3	2	1	0
0	OPERATION CODE (TBD)							
1	RESERVED			SERVICE ACTION (TBD)				
2	RESERVED							
3	RESERVED							
4	(MSB)			LUN_C=0				
5								(LSB)
6	(MSB)							
7	Allocation Length							
8								
9								(LSB)
10	RESERVED							RPTSEL
11	CONTROL							

The parameter list includes the volume set ID and the LUN field to unbind the volume set.

Bit Byte	7	6	5	4	3	2	1	0
	Volume Set ID							
0	Volume Set ID byte 0							
7	Volume Set ID byte 7							
8	LUN Number byte 0							
15	LUN Number byte 7							

Report Volume Set Mount Status

The Report Volume Set Mount Status command returns all the volume sets known behind a controller attachment. Any controller in the attachment will return the same information. The information is grouped per volume set and includes the LUN for the volume set (0 means dismounted/offline), the LUN profile, and the controllers involved with this LUN from the set of attached controllers.

Bit Byte	7	6	5	4	3	2	1	0
0	OPERATION CODE (TBD)							
1	RESERVED			SERVICE ACTION (TBD)				
2	RESERVED							
3	RESERVED							
4	(MSB)	LUN_C=0						
5								(LSB)
6	(MSB)							
7	Allocation Length							
8								
9								(LSB)
10	RESERVED							RPTSEL
11	CONTROL							

The parameter list includes the volume ID, the LUN, the profile, the controller count, and the Controller WWN's

Bit Byte	7	6	5	4	3	2	1	0
	Volume Mount Status							
0	Volume ID byte 0							
7	Volume ID byte 7							
8	LUN ID byte 0							
15	LUN ID byte 7							
16	RES | Selected LUN Access Profile Control Bit Mask							
17	RES | Supported LUN Access Profile Control Bit Mask							
18	Controller Count (CC)							
19	Controller WWN 0 byte 0							
26	Controller WWN 0 byte 7							
...								
N +8* CC-1	Controller WWN CC-1 byte 7							
	Volume ID byte 0							
...								
	Controller WWN CC-1 byte 7							

For the Report Command, the only difference in fields is byte 17. Bit 7 of byte 17 denotes the ability of a host to change the LUN Access Profile Control Field. If the C/NC field is a '1', then the Selected Access Profile Control Bit Mask may be changed by hosts. Byte 17 is the mask of supported Access Profile Control options. Byte 16 will reflect the currently selected LUN Access Profile Control option. If the C/NC field is '0' then Byte 16 and 17 will contain the same LUN Access Profile Control field.

SCSI-3 Changes

The event codes would be used with exceptions and be returned following the conventions of the Exceptions Mode page. These events would be reported by one ore more of the controllers in the fault tolerant controller configuration. The choice of which controller in a multi-controller fault tolerant configuration is outside the scope of this profile because the mechanisms to allow choice are here with the use of the new event codes, coupled with some of the persistent reserve concepts and other SCSI-3 facilities.

The specific ASC/ASCq event codes are:

FAILOVER - tbd code

FAILBACK - tbd code

The use of these codes is optional by the controller. Controllers that utilize these ASC/ASCq's and host operating system drivers that recognize these event codes can react to the failover or failback of a controller in a configuration in a proactive, performance oriented way, rather than in the command error recovery path after a command time-out has occurred on the failing controller.

Benefits of Change to Industry

The benefits of the changes to SCC and the new ASC/ASCq's are:

1. In a multi-host, multi-controller environment where the controllers may or may not share access to storage, these changes will provide for easy configuration checks by Operating System Drivers during initialization or during normal operations. The fact that SCC supports this function will

further ease in interoperability in the open system environment, with varying operating systems running on various host systems in open networks.

2. In any highly available system environment, the use of (1) above and the new ASC/ASCq's will provide for much faster failover (recovery from controller failure by a partner controller sharing access to storage) than would normally be provided by time-outs. The ASC/ASCq combination provides for quick notification of failure from a surviving partner controller .
3. In a multi-host, multi-interconnect environment, the changes to SCC will afford host a much easier, standard method for identifying opportunities for load balancing storage service across controllers in a redundant controller configuration that provides perceived multi-port, single controller support with fault tolerance. These changes are consistent and complimentary to the proposed Persistent Reserve changes (Snively).
4. Large systems will benefit from the ease of mounting and dismounting volume sets across a large and sometimes repetitive LUN naming environment. Volume sets can also be placed in 'offline' status simply by dismounting the volume. Any volume set can be brought 'online' with a mount volume command.

Functional Description

Utilizing SCC, the Attach Component Device command may be used by host to create an attachment between controllers to create or add to a fault tolerant controller configuration.

Hosts can interrogate a controller that has been pre-configured in a fault tolerant configuration or verify an Attach Component Device service action by using the Report Component Device Attachment service action noted above.

The controllers must also share the use of LUN0 on every controller in the same fault tolerant controller configuration. LUN0 on each attached controller will report the same configuration. This keeps the subsystem consistent. Furthermore, any host or external user interface configurations entered on one controller MUST also be relayed immediately to all other controllers in the same fault tolerant controller configuration.

The controller attachment is named for use in systems where controllers may come and go from the fault tolerant configuration due to reconfigurations, failures, and upgrades. The naming must cover the needs of the overall system the configuration is attached to. The naming must handle a change in membership, either from an addition to the configuration, a deletion from the configuration, or from a replacement in configuration (after failure). The name must be unique with a system installation, but not necessarily world-wide unique. The name must essentially be a controller configuration 'handle' that can be used by any host operation system to key off of in order to handle multiple paths to the same devices or LUNs.

LUNs are either pre-configured or setup by hosts. The Mount Volume Set command allows setup of volume sets bound to a desired LUN. The Mount Volume Set command also contains the LUN access profile to be used for that LUN. The access profile includes the primary access path(s) and the secondary (failover) access path(s). Volume sets may be unbound from LUNs and made inaccessible to hosts with the Dismount Volume Set command. The Mount and Dismount Volume Set commands are optional.

The status of volume sets and LUNs across a controller attachment can be obtained with the Report Volume Set Mount Status command. This command will report on all volume sets behind a set of attached controllers. This command will note any LUN bindings, access profile, and controllers that may access this LUN using the profile. This command is mandatory to generalize reporting of volume set bindings to LUNs and access path information.

A normally functioning fault tolerant controller configuration consisting of two or more controller devices acting as one SACL, will react to one of the partner controllers failure in the following way:

1. One or more controllers will detect the failure of a partner controller.

2. If the ASC/ASCqs are used, then the detecting controller(s) will respond with them after the completion of the their current command from an attached host. The surviving controllers will also be alerted to this fact by means of their direct or indirect communication path between controllers.

3. The controllers will decide which controller will take over the failed controller's LUN service, or may wait for host that utilize reserve and release features to move the LUN service to one or more of the surviving controller devices.

4. The failed controller device will still show up in the report component device attachment, but of course will not respond to any host requests.

A functioning fault tolerant controller configuration that has a partner controller either restarted or replaced after failure, will react to this event in the following way:

1. The controller(s) in the configuration may automatically actively incorporate the restarted/replaced controller device in the fault tolerant configuration.

2. The controller(s) in the configuration may be directed to incorporate the restarted/replaced controller device in the fault tolerant configuration by a local interface or via SCC over the host interconnect.

3. The restarted/replaced controller will then be ready for LUN service after verification of the configuration to LUNs and configured containers/devices by direct or indirect communication with the other controller(s) in the fault tolerant controller configuration.

4. The hosts may be notified by one or more controller devices that the previously failed member of the configuration has returned using the optional ASC/ASCq mechanism. After this load balancing may occur from other controllers to this newly returned controller.