IBM XIV Storage System User Manual

Version 10.1

GC27-2213-02

IBM XIV Storage System User Manual

Version 10.1

GC27-2213-02

Contents

Chapter 1. Introduction

Reliable high-end storage systems are critical for the successful operation of businesses. The XIV Grid Storage Platform is designed to guarantee secure, dependable, enterprise-grade data storage and access, straightforward and non-intrusive installation and upgrade and full scalability.

At the heart of the system are proprietary and innovative algorithms to offset any imaginable hardware malfunction and to minimize maintenance requirements. The flexibility and robustness of the system is further enhanced by virtue of the off-the-shelf hardware components (such as the SATA disk drives) that are easily integrated and supported.

Purpose and Scope

This document presents the XIV Command Line Interface (XCLI) functions. Relevant tables, charts, sample outputs and appropriate examples are also provided, as applicable.

This document contains the following chapters:

Introduction	Introduces the document, intended audience, related documentation and document conventions.
CLI Commands	Provides detailed information about each command in the XIV Command Line Interface (XCLI).
Event Descriptions	Provides detailed information about the events generated by the system.
Return Codes	Lists all UNIX return codes returned by the XCLI command.
Glossary	Provides an alphabetically ordered list of the definitions of the key terms and abbreviations used in this document.
Index	Provides an index of this document.

Intended Audience

This document serves as a reference for System Administrators and all IT staff that interface with the system via the CLI.

Related Documentation

- IBM XIV Theory of Operations
- IBM XIV XCLI Utility
- IBM XIV Storage System Release Notes - this document lists command addition, deletion and change for version 10.1.

Documentation Conventions

- *Notes* are embedded in the text, as shown in the example below.

  Note:

  This is an example of a Note.
- Code samples or output samples are documented in monospaced font. The text box for examples and output is framed around it. For example:
  –

    Example:

    ```
    xcli vol_rename vol=DBVolume new_name=DBVolume1
    ```

  –

    Output:

    ```
    Command completed successfully
    ```

Parameter Definitions

Definition	Description	Syntax
IP Address	An address of the form N.N.N.N, where each N is a number between 0 and 255	n.n.n.n for n between 0 and 255.
Existing <object>	The name of an object. The object must already exist	Letters, digits, ~, \, ., _, -, with a maximum of 63, no spaces at the beginning and the end, no ALL or NONE (regardless of case).
Format string with runtime tokens.	A format string, where pre-defined tokens are replaced with run time information.	Letters, digits, ., - with a maximum of 64, with {} to define tokens.
iSCSI initiator name.	A legal name of iSCSI initiator.	Up to 253 characters with no spaces.

Chapter 2. Host and Cluster Management

The following sections describe the XIV Command Line Interface (XCLI) for host and cluster management.

The sections are listed as follows:

- host_define(Defines a new host to connect to the XIV system.)
- host_add_port(Adds a port address to a host.)
- host_remove_port(Removes a port from a host.)
- host_list(Lists a specific host or all hosts.)
- host_list_ports(Lists all the ports of a host)
- host_rename(Renames a host.)
- host_delete(Deletes a host.)
- cluster_create(Creates a new cluster.)
- cluster_add_host(Adds a host to a cluster.)
- cluster_remove_host(Removes a host from a cluster.)
- cluster_list(Lists a specific cluster or all of them.)
- cluster_rename(Renames a cluster.)
- cluster_delete(Deletes a cluster.)
- map_vol(Maps a volume to a host or a cluster.)
- unmap_vol(Unmaps a volume from a host or a cluster.)
- mapping_list(Lists the mapping of volumes to a specified host or cluster.)
- vol_mapping_list(Lists all hosts and clusters to which a volume is mapped.)
- special_type_set(Sets the special type of a host or a cluster.)

The following commands are no longer in use:

- host_define_fc (This command has been replaced by the host_define command)
- host_define_iscsi (This command has been replaced by the host_define command)
- host_luns_list (This command has been replaced by map_list_luns)

Defining a New Host

Defines a new host to connect to the XIV system.

```
host_define host=HostName [ cluster=ClusterName ]
```

Parameters:

Name	Type	Description	Mandatory	Default
host	Object name	Name of host to be created.	Y	
cluster	Object name	Name of cluster to contain host.	N	No cluster.

This command is used to define a host that will attach to the XIV system. The name of the host must be unique in the system.

Note:

Use the Adding a Port to a Host command to add port addresses to this host. Specifying the cluster is optional.

Example:

```
xcli -u  -c XIV1 host_define host=server1
```

Output:

```
Command executed successfully.
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- HOST_NAME_EXISTS
  Host name already exists
- MAX_HOSTS_REACHED
  Maximum number of hosts already defined
- CLUSTER_BAD_NAME
  Cluster name does not exist

Adding a Port to a Host

Adds a port address to a host.

```
host_add_port host=HostName < fcaddress=FCAdress | iscsi_name=iSCSIName >
```

Parameters:

Name	Type	Description	Mandatory	Default
host	Object name	The host name.	Y	
fcaddress		FC address of the added port.	N	
iscsi_name	iSCSI initiator name	iSCSI initiator name of the newly added port.	N	

The FC port address or iSCSI initiator (port) name assigned to the host must be unique per XIV system. The FC port name must be exactly 16 characters long, in hexadecimal form.

Only the following alphanumeric characters are valid: 0-9, A-F, a-f. In addition to the 16 characters, colons (:) may be used as separators in the 16 character port name. The iSCSI initiator name may not exceed 253 characters and may not contain any blank spaces.

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- HOST_BAD_NAME

  Host name does not exist
- HOST_PORT_EXISTS

  Host with this port ID already defined
- ISCSI_HOST_ILLEGAL_PORT_NAME

  Port name for iSCSI Host is illegal

  Troubleshooting: Port names for iSCSI Hosts must contain only printable characters.
- MAX_PORTS_REACHED

  Maximum number of ports already defined in the system

Removing a Port from a Host

Removes a port from a host.

```
host_remove_port host=HostName < fcaddress=FCAdress | iscsi_name=iSCSIName >
```

Parameters:

Name	Type	Description	Mandatory	Default
host	Object name	The host name.	Y	
fcaddress		FC address of the port to be removed.	N	
iscsi_name	iSCSI initiator name	iSCSI initiator name of the port to be removed.	N	

This command removes a port from a host.

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- HOST_BAD_NAME
  Host name does not exist
- PORT_DOES_NOT_BELONG_TO_HOST
  Port ID belongs to another host
- HOST_PORT_DOES_NOT_EXIST
  Port ID is not defined

Listing Hosts

Lists a specific host or all hosts.

```
host_list [ host=HostName ]
```

Parameters:

Name	Type	Description	Mandatory	Default
host	Object name	The host name.	N	All hosts.

This command lists all the hosts that have been defined in the XIV system.

A host name can be specified to list only a specific host or all the hosts.

The list contains the following comma separated information:

- Port addresses
- Containing cluster, if one exists
- Associated users and user groups

Example:

```
xcli -u  -c Nextra1 host_list host=mailserver
```

Output:

```
Name     Type      FC Ports   iSCSI Ports   User Group   Cluster
-------- --------- ---------- ------------- ------------ ---------
host_4   default              iscsi_4
host_5   default              iscsi_5
host_6   default              iscsi_6
host_7   default              iscsi_7
host_8   default              iscsi_8
host_9   default              iscsi_9
```

Id	Name	Description	Default Position
name	Name		1
type	Type		2
fc_ports	FC Ports		3
iscsi_ports	iSCSI Ports		4
creator	Creator		
user_group	User Group		5
cluster	Cluster		6

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Allowed	
Read-only users	Allowed	
Technicians	Disallowed	

Listing Ports

Lists all the ports of a host

```
host_list_ports host=HostName
```

Parameters:

Name	Type	Description	Mandatory	Default
host	Object name	The host name.	Y	

This command lists all the ports on a specified host.

Id	Name	Description	Default Position
host	Host		1
type	Type		2
port_name	Port Name		3

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Allowed	
Read-only users	Allowed	
Technicians	Disallowed	

Renaming a Host

Renames a host.

```
host_rename host=HostName new_name=NewName
```

Parameters:

Name	Type	Description	Mandatory	Default
host	Object name	The original host name.	Y	
new_name	Object name	The new host name. The new host name must be unique in the system.	Y	

This command renames a host. The new name of the host must be unique in the system.

The command still succeeds even if the new name is identical to the current name.

Example:

```
xcli -u  -c Nextra1 host_rename host=server2 new_name=mailserver
```

Output:

```
Command completed successfully
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- HOST_BAD_NAME
  Host name does not exist
- HOST_NAME_EXISTS
  Host name already exists

Deleting a Host

Deletes a host.

```
host_delete host=HostName
```

Parameters:

Name	Type	Description	Mandatory	Default
host	Object name	The host name.	Y	

This command deletes a host. After this command is executed, the deleted host can no longer connect to the system, and I/O requests from this host are not handled.

Example:

```
xcli -u  -c Nextra1 host_delete host=mailserver
```

Output:

```
Command completed successfully
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Warnings:

- ARE_YOU_SURE_YOU_WANT_TO_DELETE_HOST
  Are you sure you want to delete host *Host*?

Completion Codes:

- HOST_BAD_NAME
  Host name does not exist

Creating a Cluster

Creates a new cluster.

```
cluster_create cluster=ClusterName
```

Parameters:

Name	Type	Description	Mandatory	Default
cluster	Object name	Name of the cluster to be created.	Y	

This command creates a new cluster. The newly created cluster does not contain hosts, has the default type and has no mapping.

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- CLUSTER_NAME_EXISTS
  Cluster name already exists
- MAX_CLUSTERS_REACHED
  Maximum number of clusters already defined

Adding a Host to a Cluster

Adds a host to a cluster.

```
cluster_add_host cluster=ClusterName host=HostName map=<cluster|host>
```

Parameters:

Name	Type	Description	Mandatory	Default
cluster	Object name	Name of the cluster to contain the host.	Y	
host	Object name	Host to be added to the cluster.	Y	
map	Enumeration	Override the mapping of the host or cluster.	Y	

This command adds a host to a cluster.

This command fails if the host already belongs to another cluster.

This operation succeeds if the host already belongs to the specified cluster and has no effect.

Using the map parameter:

- If the map parameter is cluster, the mapping of the host and host type is changed to be the cluster's mapping and type.
- If the map parameter is host, the mapping of the cluster and its host type is changed to be the host's mapping and type.

The host or cluster is getting a single SCSI unit attention message, even if the change affects multiple volumes.

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- HOST_BAD_NAME
  Host name does not exist
- CLUSTER_BAD_NAME
  Cluster name does not exist
- HOST_BELONGS_TO_ANOTHER_CLUSTER
  Host already belongs to another cluster

Removing a Host from a Cluster

Removes a host from a cluster.

```
cluster_remove_host cluster=ClusterName host=HostName
```

Parameters:

Name	Type	Description	Mandatory	Default
cluster	Object name	Cluster name.	Y	
host	Object name	Host to be removed from cluster.	Y	

This command removes the specified host from a cluster. The host then no longer belongs to any cluster. The host's special type and mapping remain identical to the cluster's special type and mapping, and therefore, I/O is not interrupted. The association of the host with user or user groups remains the same as the cluster's association.

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	

User Category	Permission	Condition
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- HOST_BAD_NAME
  Host name does not exist
- CLUSTER_BAD_NAME
  Cluster name does not exist
- HOST_NOT_IN_CLUSTER
  Host is not part of specified cluster

Listing Clusters

Lists a specific cluster or all of them.

```
cluster_list [ cluster=ClusterName ]
```

Parameters:

Name	Type	Description	Mandatory	Default
cluster	Object name	Cluster to be listed.	N	All clusters.

This command lists a specific cluster or all of them. For each cluster, a special type and a comma separated list of hosts is listed.

Id	Name	Description	Default Position
name	Name		1
hosts	Hosts		2
type	Type		3
creator	Creator		4
user_group	User Group		5

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Allowed	
Read-only users	Allowed	
Technicians	Disallowed	

Renaming Clusters

Renames a cluster.

```
cluster_rename cluster=ClusterName new_name=NewName
```

Parameters:

Name	Type	Description	Mandatory	Default
cluster	Object name	Cluster to be renamed.	Y	
new_name	Object name	New name of cluster.	Y	

This command renames the specified cluster.

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- CLUSTER_BAD_NAME
  Cluster name does not exist
- CLUSTER_NAME_EXISTS
  Cluster name already exists

Deleting Clusters

Deletes a cluster.

```
cluster_delete cluster=ClusterName
```

Parameters:

Name	Type	Description	Mandatory	Default
cluster	Object name	Cluster to be deleted.	Y	

This command deletes a cluster. All hosts contained in the cluster remain active and are not deleted. The special type of each host is set to the cluster's special type. The mapping of each host is set to the cluster's mapping. No I/O interruption is caused by this command.

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- CLUSTER_BAD_NAME
  Cluster name does not exist
- **Target event not found. href = ARE_YOU_SURE_YOU_WANT_TO_DELETE_ASSIGNED_CLUSTER**

Mapping a Volume to a Host or Cluster

Maps a volume to a host or a cluster.

```
map_vol <host=HostName | cluster=ClusterName> vol=VolName lun=LUN [ override=<no|yes> ]
```

Parameters:

Name	Type	Description	Mandatory	Default
host	Object name	Host name.	N	
cluster	Object name	Cluster name.	N	
vol	Object name	Volume name.	Y	
lun	Integer	LUN identifier.	Y	
override	Boolean	Override existing mapping.	N	no

This command maps a volume to a host or to a cluster. It maps the volume to all the hosts that are contained in the cluster.

if the command specifies a volume or a LUN already specified for the cluster the command will fail; in other cases, a warning will be presented stating that a host-specific mapping will be established, but the command will not fail.

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Conditionally Allowed	This volume is a snapshot. The master volume of this snapshot is mapped to a host or cluster that is associated with the user executing this command. This snapshot was created by an application administrator.

User Category	Permission	Condition
Read-only users	Disallowed	
Technicians	Disallowed	

Warnings:

- ARE_YOU_SURE_YOU_WANT_TO_PERFORM_HOST_SPECIFIC_MAPPING

  '*Host*' is part of a cluster. Are you sure you want to map this volume only for that specific host?
- ARE_YOU_SURE_YOU_WANT_TO_MAP_VOLUME

  Are you sure you want to map volume *Volume*, which is already mapped to another host/cluster?

Completion Codes:

- HOST_BAD_NAME

  Host name does not exist
- HOST_BELONGS_TO_CLUSTER

  Host is part of a cluster
- CLUSTER_BAD_NAME

  Cluster name does not exist
- VOLUME_BAD_NAME

  Volume name does not exist
- SNAPSHOT_IS_INTERNAL

  Target snapshot cannot be internal

  Internal snapshots cannot be mapped, modified or deleted
- VOLUME_ALREADY_ASSIGNED

  Mapping conflict: volume is already assigned
- LUN_ALREADY_IN_USE

  Mapping conflict: LUN is already in use
- EXT_LUN_ILLEGAL

  LUN is out of range or does not exist
- VOLUME_HAS_HOST_SPECIFIC_MAPPING

  Specified Volume is currently mapped to another LUN in a host-specific mapping
- LUN_HAS_HOST_SPECIFIC_MAPPING

  Specified LUN currently has another volume mapped in a host-specific mapping

Unmapping a Volume from a Host or Cluster

Unmaps a volume from a host or a cluster.

```
unmap_vol <host=HostName | cluster=ClusterName> vol=VolName
```

Parameters:

Name	Type	Description	Mandatory	Default
host	Object name	Host name.	N	

Name	Type	Description	Mandatory	Default
cluster	Object name	Cluster name.	N	
vol	Object name	Volume name.	Y	

This command unmaps a volume from a host or a cluster.

The command to unmap from a cluster will unmap the volume from all the hosts that are contained in that cluster.

The command fails if the specified host is contained in a cluster. In this case, the unmapping of the host must be performed through the cluster.

The command does not fail when the volume is not mapped to the host/cluster.

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Conditionally Allowed	The volume is a snapshot, where its master volume is mapped to a host or cluster associated with the user and the snapshot was created by an application administrator.
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- HOST_BAD_NAME
  Host name does not exist
- HOST_BELONGS_TO_CLUSTER
  Host is part of a cluster
- CLUSTER_BAD_NAME
  Cluster name does not exist
- VOLUME_BAD_NAME
  Volume name does not exist

Listing the Mapping of Volumes to Hosts or Clusters

Lists the mapping of volumes to a specified host or cluster.

```
mapping_list <host=HostName | cluster=ClusterName>
```

Parameters:

Name	Type	Description	Mandatory	Default
host	Object name	Host name.	N	
cluster	Object name	Cluster name.	N	

This command indicates per each host mapping whether it is cluster-based.

Id	Name	Description	Default Position
lun	LUN		1
volume	Volume		2
size	Size		3
master	Master		4
serial	Serial Number		5
locked	Locked		6

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Allowed	
Read-only users	Allowed	
Technicians	Disallowed	

Completion Codes:

- HOST_BAD_NAME
  Host name does not exist
- CLUSTER_BAD_NAME
  Cluster name does not exist

Listing Hosts/Cluster to which a Volume is Mapped

Lists all hosts and clusters to which a volume is mapped.

```
vol_mapping_list vol=VolName
```

Parameters:

Name	Type	Description	Mandatory	Default
vol	Object name	Volume name.	Y	

This command lists all the hosts and clusters to which a volume is mapped, as well as hosts that are part of a cluster and have host-specific mapping to the volume. The output list contains two columns: name of host/cluster and type (host or cluster).

Id	Name	Description	Default Position
host	Host/Cluster		1
type	Type		2
lun	LUN		3

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- VOLUME_BAD_NAME
  Volume name does not exist

Setting the Special Type of Hosts or Clusters

Sets the special type of a host or a cluster.

```
special_type_set <host=HostName | cluster=ClusterName>  type=<default|hpux>
```

Parameters:

Name	Type	Description	Mandatory	Default
host	Object name	Host name.	N	
cluster	Object name	Cluster name.	N	
type	Enumeration	Special map type.	Y	

This command sets a special type for a host or a cluster. The only supported special type is *hpux*. It should be specified for hosts or clusters that run the HP/UX operating system. All other operating systems do not require a special type.

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- HOST_BAD_NAME
  Host name does not exist
- HOST_BELONGS_TO_CLUSTER
  Host is part of a cluster
- CLUSTER_BAD_NAME
  Cluster name does not exist

Chapter 3. Volume Management

The following sections describe the XIV Command Line Interface (XCLI) for volume management.

See also:

- Volume Snapshot Management
- Consistency Group Management
- Storage Pool Management

The sections are listed as follows:

- vol_create(Creates a new volume.)
- vol_resize(Resizes a volume.)
- vol_copy(Copies a source volume onto a target volume.)
- vol_format(Formats a volume.)
- vol_lock(Locks a volume so that it is read-only.)
- vol_unlock(Unlocks a volume, so that it is no longer read-only and can be written to.)
- vol_by_id(Prints the volume name according to its specified SCSI serial number.)
- vol_list(Lists all volumes or a specific one.)
- vol_rename(Renames a volume.)
- vol_delete(Deletes a volume.)
- reservation_list(Lists volume reservations.)
- reservation_key_list(Lists reservation keys.)
- reservation_clear(Clear reservations of a volume.)

The following commands are no longer in use:

- vol_clear_keys (Command is no longer in use in this version. Supplanted by new command reservation_clear.)

Creating a Volume

Creates a new volume.

```
vol_create vol=VolName < size=GB | size_blocks=BLOCKS > pool=PoolName
```

Parameters:

Name	Type	Description	Mandatory	Default
vol	Object name	Volume name.	Y	
size	Positive integer	Volume size in GB.	N	
size_blocks	Positive integer	Size in number of blocks.	N	

Name	Type	Description	Mandatory	Default
pool	Object name	The name of the Storage Pool to which the volume belongs.	Y	

This command creates a new volume.

The name of the volume must be unique in the system.

Space for the volume is allocated from the specified Storage Pool and the volume belongs to that Storage Pool. Specifying the Storage Pool is mandatory.

When creating a volume, the storage space that is needed to support the volume's capacity is reserved from the soft capacity of the storage pool for the volume.

The command fails if this reservation cannot be committed.

The volume is logically formatted at creation time, which means that any read operation results in returning all zeros as a response.

The size is the actual "net" storage space, as seen by the user's applications, not including any mirroring or other data protection overhead.

The free space consumed by the volume will be the smallest multiple of 17GB which is bigger than the specified size.

The size can be specified either in gigabytes or in blocks (where each block is 512 bytes). If the size is specified in blocks, volumes are created in the exact size specified. If the size is specified in gigabytes, the actual volume size is rounded up to the nearest 17GB multiple (making the actual size identical to the free space consumed by the volume, as described above). This rounding up prevents a situation where storage space is not fully utilized because of a gap between the free space used and the space available to the application. The size specified in blocks is exact.

The term GB (gigabytes) is defined in this context as 10^9 (and not as 2^{30} as in many other contexts).

Upon successful completion of the command, its lock state is *unlocked*, meaning that write, format and resize operations are allowed.

The creation time of the volume is set to the current time and is never changed.

Example:

```
xcli -u  -c Nextra1 vol_create vol=DBVolume size=2000
```

Output:

```
Command executed successfully.
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- VOLUME_CANNOT_HAVE_ZERO_SIZE
  Volume size cannot be zero
- POOL_DOES_NOT_EXIST
  Storage Pool does not exist
- VOLUME_EXISTS
  Volume name already exists
- VOLUME_BAD_NAME
  Volume name does not exist
- VOLUME_BAD_PREFIX
  Volume name has a reserved prefix
- NOT_ENOUGH_SPACE
  No space to allocate volume
- MAX_VOLUMES_REACHED
  Maximum number of volumes already defined
  Maximum number of volumes already defined

Resizing a Volume

Resizes a volume.

```
vol_resize vol=VolName  < size=GB | size_blocks=BLOCKS >  [ shrink_volume=<yes|no> ]
```

Parameters:

Name	Type	Description	Mandatory	Default
vol	Object name	The name of the volume to be resized.	Y	
size		The new volume size.	Y	
size_blocks		New size of volumes in number of blocks.	N	
shrink_volume	Boolean	Must be specified as `yes` if the new size is smaller than the current size.	N	No

This command resizes a volume.

The volume can be resized in either direction. However, whenever the volume is downsized, you have to specifiy this with shrink_volume="yes".

The new size of the volume is specified as an integer multiple of 10^9 bytes, but the actual new size of the volume is rounded up to the nearest valid size, which is an integer multiple of 16 x 2^{30} bytes.

The command can only be used to increase the size of a volume, and fails if the new size is smaller than the current size. If the new size equals the current size, the command will succeed without changes to the volume.

The volume's address space is extended at its end to reflect the increased size, and the additional capacity is logically formatted (that is, zeros are returned for all read commands).

When resizing a regular volume (not a writable snapshot), all storage space that is needed to support the additional volume's capacity is reserved (static allocation). This guarantees the functionality and integrity of the volume, regardless of the resource levels of the volume's Storage Pool. The command fails if this reservation cannot be committed.

The volume's lock state must be unlocked when the command is issued, or otherwise the command fails.

- Resizing a master volume does not change the size of its associated snapshots.
- These snapshots can still be used to restore their individual master volumes.
- The same goes for resizing a snapshot: it does not change the size of its master volume.

Example:

```
xcli -c Nextra1 vol_resize vol=DBVolume size=2500
```

Output:

```
Command completed successfully
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Warnings:

- ARE_YOU_SURE_YOU_WANT_TO_ENLARGE_VOLUME

  Are you sure you want to increase volume size?

- ARE_YOU_SURE_YOU_WANT_TO_REDUCE_VOLUME
  Decreasing volume size may cause data loss. Are you sure you want to proceed?
- VOLUME_WILL_CROSS_1TB_SIZE
  Many operating systems do not support a resize operation across the 1TB boundary, are you sure?
  Troubleshooting: Snapshot backup before resize is advised

Completion Codes:

- VOLUME_BAD_NAME
  Volume name does not exist
- ILLEGAL_VOLUME_SIZE
  Illegal volume size
- NOT_ENOUGH_SPACE
  No space to allocate volume
- NOT_ENOUGH_SPACE_ON_REMOTE_MACHINE
  Not enough free space to set requested size of slave volume
- VOLUME_LOCKED
  Volume is locked
- VOLUME_IS_SLAVE
  Volume is defined as a slave volume
- VOLUME_MIRROR_NOT_OPERATIONAL
  Volume has a mirror that is not operational
- VOLUME_HAS_DATA_MIGRATION
  Data Migration is defined for this volume
- CAN_NOT_SHRINK_MAPPED_VOLUME
  Mapped volume's size can not be decreased
- CAN_NOT_SHRINK_VOLUME_WITH_SNAPSHOTS
  Size of volume with snapshots can not be decreased
- CAN_NOT_SHRINK_REMOTE_VOLUME_WITH_SNAPSHOTS
  Remote volume has snapshots
- CAN_NOT_SHRINK_MAPPED_REMOTE_VOLUME
  Remote volume is mapped
- REMOTE_VOLUME_HAS_DATA_MIGRATION
  Data Migration is defined for slave volume
- VOLUME_CANNOT_HAVE_ZERO_SIZE
  Volume size cannot be zero
- CAN_NOT_SHRINK_SNAPSHOTS
  Size of snapshots can not be decreased

Copying Volumes

Copies a source volume onto a target volume.

```
vol_copy vol_src=VolName vol_trg=VolName
```

Parameters:

Name	Type	Description	Mandatory	Default
vol_src	Object name	Name of the source volume from which the data is to be taken.	Y	
vol_trg	Object name	Name of the target volume to which the data is to be copied.	Y	

This command copies a source volume onto a target volume.

All data stored on the target volume is lost and cannot be restored.

This command performs the following as a single atomic action:

- Deletes the target volume.
- Creates a new volume with the same name as the target volume and the same size as the source volume.
- Instantly copies the source volume data onto the target volume.

All volume to host mappings of the target volume remain intact during this process. Except for its size, the target volume retains all of its properties, including its name, ID, lock state, creation time and all other attributes.

Immediately after the completion of the command, the volumes are independent of each other and are valid for any further operations (including deletion).

If the target volume is larger then the source volume, excess storage space is freed and returned to the target volume's Storage Pool. If the target volume is smaller than the source volume, all storage space that is needed to support the additional volume's capacity is reserved from the storage pool.

Both the source volume and the target volume must exist when the command is issued.

The command fails in the following cases:

- The source volume is larger than the target volume, and there is not enough free space in the Storage Pool that contains the target for target volume resizing.
- The target volume has a snapshot associated with it or if the target volume is a snapshot.
- The target volume is locked.
- The target volume is part of any mirroring definitions (either master or slave).
- The source volume is a slave of a synchronous mirroring, and it is currently inconsistent due to either a re-synchronization or an initialization process.

Example:

```
xcli -c Nextra1 vol_copy vol_src=DBVolume vol_trg=DBVolumeCopy
```

Output:

```
Command completed successfully
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Warnings:

- ARE_YOU_SURE_YOU_WANT_TO_COPY_VOLUME

  Are you sure you want to copy the contents of volume *source Volume* to volume *target Volume*?

Completion Codes:

- SOURCE_VOLUME_BAD_NAME

  Source volume name does not exist
- TARGET_VOLUME_BAD_NAME

  Target volume name does not exist
- NOT_ENOUGH_HARD_SPACE

  No space to allocate for volume's current usage
- NOT_ENOUGH_SPACE

  No space to allocate volume
- VOLUME_IDENTICAL

  Source and target are the same volume
- VOLUME_HAS_SNAPSHOTS

  Volume has snapshots
- TARGET_VOLUME_LOCKED

  Target volume is locked
- TARGET_VOLUME_HAS_MIRROR

  Mirror is defined for target volume
- TARGET_VOLUME_HAS_DATA_MIGRATION

  Data Migration is defined for target volume
- SOURCE_VOLUME_DATA_MIGRATION_UNSYNCHRONIZED

  Data Migration has not completed to source volume
- VOLUME_IS_SNAPSHOT

  Operation is not permitted on snapshots

  Operation is not permitted on snapshots
- VOLUME_IS_NOT_CONSISTENT_SLAVE

  Operation not allowed on slave volume that is not consistent.
- TARGET_VOLUME_NOT_FORMATTED

  Target volume is not formatted

Formatting a Volume

Formats a volume.

```
vol_format vol=VolName
```

Parameters:

Name	Type	Description	Mandatory	Default
vol	Object name	Name of the volume to be formatted.	Y	

This command formats a volume. A formatted volume returns zeros as a response to any read command.

All data stored on the volume is lost and cannot be restored.

The formatting of the volume is done logically and no data is actually written to the physical storage space allocated for the volume. This allows the command to complete instantly.

The volume's lock state must be unlocked when the command is issued.

This command fails if the volume has snapshots associated with it, or if the volume is a snapshot, or if the volume is part of any mirroring or Data Migration definition.

Example:

```
xcli -c Nextra1 vol_format vol=DBVolume
```

Output:

```
Command completed successfully
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Warnings:

- ARE_YOU_SURE_YOU_WANT_TO_FORMAT_VOLUME

  Volume *Volume* may contain data. Formatting will cause data loss. Are you sure you want to format volume *Volume*?

Completion Codes:

- VOLUME_BAD_NAME
  Volume name does not exist
- VOLUME_HAS_SNAPSHOTS
  Volume has snapshots
- VOLUME_IS_SNAPSHOT
  Operation is not permitted on snapshots
  Operation is not permitted on snapshots
- VOLUME_LOCKED
  Volume is locked
- VOLUME_HAS_MIRROR
  Mirror is defined for this volume
- VOLUME_HAS_DATA_MIGRATION
  Data Migration is defined for this volume

Locking a Volume

Locks a volume so that it is read-only.

```
vol_lock vol=VolName
```

Parameters:

Name	Type	Description	Mandatory	Default
vol	Object name	Name of the volume to lock.	Y	

This command locks a volume so that hosts cannot write to it.

A volume that is locked is write-protected, so that hosts can read the data stored on it, but cannot change it. In addition, a locked volume cannot be formatted or resized. In general, locking a volume prevents any operation (other than deletion) that changes the volume's image.

This command succeeds when the volume's lock state is already set to the one the user is trying to apply, while leaving it in the same lock state.

The lock states of master volumes are set to *unlocked* when they are created.

The lock states of snapshots are set to *locked* when they are created.

In addition to the lock state, snapshots also have a modification state. The modification state is a read-only state (which cannot be changed by the user explicitly) and it is initially set to *unmodified* when the snapshot is created. The first time a snapshot lock state is set to *unlocked*, the modification state of the snapshot is changed to *modified*, and it is never changed thereafter.

Example:

```
xcli -c Nextra1 vol_lock vol=DBVolume
```

Output:

```
Command completed successfully
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Conditionally Allowed	The volume is a snapshot, where its master volume is mapped to a host or cluster associated with the user and the snapshot was created by an application administrator.
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- SNAPSHOT_IS_INTERNAL
  Target snapshot cannot be internal
  Internal snapshots cannot be mapped, modified or deleted
- VOLUME_BAD_NAME
  Volume name does not exist
- VOLUME_IS_SLAVE
  Volume is defined as a slave volume
- SNAPSHOT_IS_PART_OF_SNAPSHOT_GROUP
  Snapshot is part of a Snapshot Group

Unlocking a Volume

Unlocks a volume, so that it is no longer read-only and can be written to.

```
vol_unlock vol=VolName
```

Parameters:

Name	Type	Description	Mandatory	Default
vol	Object name	The name of the volume to unlock.	Y	

This command unlocks a volume so that it is no longer read-only and can be written to.

A volume that is unlocked is no longer write-protected.

The lock state of regular volumes is set to *unlocked* when they are created.

The lock state of snapshots is set to *locked* when they are created.

In addition to the lock state, snapshots also have a modification state. The modification state is a read-only state (which cannot be changed by the user explicitly) and it is initially set to *unmodified* when the snapshot is created. The first time a snapshot lock state is set to *unlocked*, the modification state of the snapshot is changed to *modified*, and it is never changed thereafter.

Note:

The modification time is the time when the unlock command was executed, regardless of the actual changes performed on the volume via write commands.

Example:

```
xcli -c Nextra1 vol_unlock vol=DBVolume
```

Output:

```
Command completed successfully
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Conditionally Allowed	The volume is a snapshot, where its master volume is mapped to a host or cluster associated with the user and the snapshot was created by an application administrator.
Read-only users	Disallowed	
Technicians	Disallowed	

Warnings:

- ARE_YOU_SURE_YOU_WANT_TO_UNLOCK_SNAPSHOT

  After unlocking, this snapshot cannot be used to restore its master volume. Are you sure you want to unlock snapshot *Snapshot*?

Completion Codes:

- VOLUME_BAD_NAME

  Volume name does not exist
- VOLUME_IS_SLAVE

  Volume is defined as a slave volume
- SNAPSHOT_IS_INTERNAL

  Target snapshot cannot be internal

  Internal snapshots cannot be mapped, modified or deleted
- SNAPSHOT_IS_PART_OF_SNAPSHOT_GROUP

  Snapshot is part of a Snapshot Group

Finding a Volume Based on a SCSI Serial Number

Prints the volume name according to its specified SCSI serial number.

```
vol_by_id id=n
```

Parameters:

Name	Type	Description	Mandatory	Default
id	Positive integer	SCSI volume ID.	Y	

This command prints the volume name according to its specified SCSI serial number.

Id	Name	Description	Default Position
name	Name		1
size	Size (GB)		2
size_MiB	Size (MiB)		
master_name	Master Name		3
cg_name	Consistency Group		4
pool_name	Pool		5
creator	Creator		6
capacity	Capacity (blocks)		
modified	Modified		
sg_name	Snapshot Group Name		
delete_priority	Deletion Priority		
locked	Locked		
serial	Serial Number		
snapshot_time	Snapshot Creation Time		
snapshot_of	Snapshot of		
sg_snapshot_of	Snapshot of Snap Group		
wwn	WWN		
locked_by_pool	Locked by Pool		
used_capacity	Used Capacity (GB)		7

Example:

```
xcli -c Nextra1 vol_by_id id=59
```

Output:

```
Name       Size (GB)   Master Name   Consistency Group   Pool   Creator   Used Capacity (GB)
volume_1   51                                                              0
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Allowed	
Read-only users	Allowed	
Technicians	Disallowed	

Completion Codes:

- VOLUME_BAD_SERIAL
  Volume with requested SCSI serial number does not exist

Listing Volumes

Lists all volumes or a specific one.

```
vol_list [ vol=VolName ] [ pool=PoolName ]
```

Parameters:

Name	Type	Description	Mandatory	Default
vol	Object name	Name of a specific volume to be listed.	N	All volumes.
pool	Object name	Name of a specific pool whose volumes are to be listed.	N	Volumes in all pools.

This command lists volumes. When a volume name is specified, it is the only one that is listed.

Id	Name	Description	Default Position
name	Name		1
size	Size (GB)		2
size_MiB	Size (MiB)		
master_name	Master Name		3
cg_name	Consistency Group		4
pool_name	Pool		5
creator	Creator		6
capacity	Capacity (blocks)		
modified	Modified		
sg_name	Snapshot Group Name		
delete_priority	Deletion Priority		
locked	Locked		
serial	Serial Number		

Id	Name	Description	Default Position
snapshot_time	Snapshot Creation Time		
snapshot_of	Snapshot of		
sg_snapshot_of	Snapshot of Snap Group		
wwn	WWN		
locked_by_pool	Locked by Pool		
used_capacity	Used Capacity (GB)		7

Example:

```
xcli -c Nextra1 vol_list vol=DBVolume
```

Output:

```
Name                Size (GB) Master Name Pool      Creator  Used Capacity (GB)
DBLog               3006                  MainPool  admin    0
Dev                 2010                  MainPool  admin    0
Marketing           1013                  MainPool  admin    0
Dev.snapshot_00001  2010      Dev         MainPool  admin
Dev.snapshot_00002  2010      Dev         MainPool  admin
Dev.snapshot_00003  2010      Dev         MainPool  admin
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Allowed	
Read-only users	Allowed	
Technicians	Disallowed	

Renaming a Volume

Renames a volume.

```
vol_rename vol=VolName new_name=NewName
```

Parameters:

Name	Type	Description	Mandatory	Default
vol	Object name	Name of the volume to be renamed.	Y	
new_name	Object name	New volume name.	Y	

This command renames a volume.

The new name of the volume must be unique in the system.

This command succeeds even if the new name is identical to the current name.

This command succeeds regardless of the volume's lock state.

Renaming a snapshot does not change the name of its master volume. Renaming a master volume does not change the names of its associated snapshots.

Example:

```
xcli -c Nextra1 vol_rename vol=DBVolume new_name=DBVolume1
```

Output:

```
Command completed successfully
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Conditionally Allowed	The volume is a snapshot, where its master volume is mapped to a host or cluster associated with the user and the snapshot was created by an application administrator.
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- VOLUME_BAD_NAME
  Volume name does not exist
- VOLUME_EXISTS
  Volume name already exists
- SNAPSHOT_IS_INTERNAL
  Target snapshot cannot be internal
  Internal snapshots cannot be mapped, modified or deleted
- SNAPSHOT_IS_PART_OF_SNAPSHOT_GROUP
  Snapshot is part of a Snapshot Group
- VOLUME_BAD_PREFIX
  Volume name has a reserved prefix

Deleting a Volume

Deletes a volume.

```
vol_delete vol=VolName
```

Parameters:

Name	Type	Description	Mandatory	Default
vol	Object name	Name of the volume to delete.	Y	

This command deletes a volume. All data stored on the volume is lost and cannot be restored.

This command cannot be applied to a snapshot. To delete a snapshot, use Deleting a Snapshot.

All storage space allocated (or reserved) for the volume is freed and returned to the volume's Storage Pool.

The volume is removed from all LUN Maps that contain mapping of the volume.

This command deletes all snapshots associated with this volume. Even snapshots that are part of Snapshot Sets. This can happen when the volume was in a Consistency Group and was removed from it.

This command cannot be applied to a volume that is part of a Consistency Group or to a volume that is mapped to a host or cluster.

The command succeeds regardless of the volume's lock state.

Example:

```
xcli -c Nextra1 vol_delete vol=DBVolumeCopy
```

Output:

```
Command completed successfully
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Warnings:

- ARE_YOU_SURE_YOU_WANT_TO_DELETE_VOLUME

  Are you sure you want to delete volume *Volume*?
- ARE_YOU_SURE_YOU_WANT_TO_DELETE_VOLUME_WITH_SNAPSHOTS

  Volume *Volume* has snapshots! Are you sure you want to delete this volume AND all its snapshots?

Completion Codes:

- VOLUME_BAD_NAME
  Volume name does not exist
- VOLUME_HAS_MIRROR
  Mirror is defined for this volume
- SNAPSHOT_IS_PART_OF_SNAPSHOT_GROUP
  Snapshot is part of a Snapshot Group
- SNAPSHOT_IS_INTERNAL
  Target snapshot cannot be internal
  Internal snapshots cannot be mapped, modified or deleted
- VOLUME_BELONGS_TO_CG
  Volume belongs to a Consistency Group
- VOLUME_IS_MAPPED
  Volume that is mapped to a host cannot be deleted
- VOLUME_HAS_MAPPED_SNAPSHOT
  Volume which has a snapshot that is mapped to a host cannot be deleted

Listing Volume Reservations

Lists volume reservations.

```
reservation_list  [ vol=VolName ]
```

Parameters:

Name	Type	Description	Mandatory	Default
vol	Object name	Name of the volume to list reservations of.	N	All volumes.

Lists volume reservations.

Example:

```
xcli  -u -c XIV1 reservation_list vol=Vol1
```

Output:

```
Volume Name   Reserving Port   Reservation Type   Persistent
vol1                           none               none


Reservation Type   Persistent Access Type   Initiator UID   PR Generation
                   none                     -1              0
```

Id	Name	Description	Default Position
name	Volume Name		1
reserved_by_port	Reserving Port		2
reservation_type	Reservation Type		3

Id	Name	Description	Default Position
persistent_reservation_type	Persistent Reservation Type		4
access_type	Persistent Access Type		5
reserving_initiator_uid	Initiator UID	uid of reserving host	6
pr_generation	PR Generation		7

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Allowed	
Read-only users	Allowed	
Technicians	Disallowed	

Completion Codes:

- VOLUME_BAD_NAME
  Volume name does not exist

Listing Reservation Keys

Lists reservation keys.

```
reservation_key_list  [ vol=VolName ]
```

Parameters:

Name	Type	Description	Mandatory	Default
vol	Object name	Name of the volume to list reservation keys.	N	

Lists reservation keys.

Example:

```
xcli  -u -c XIV1 reservation_key_list vol=Vol2
```

Output:

```
Initiator Port     Volume Name        Reservation Key
------------------ ------------------ -----------------
100000062B151C3C   vol-dmathies-0a7   2
100000062B151C3C   vol-dobratz-23a    3
```

Id	Name	Description	Default Position
initiator_port	Initiator Port		1

Id	Name	Description	Default Position
vol_name	Volume Name		2
reg_key	Reservation Key		3

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Allowed	
Read-only users	Allowed	
Technicians	Disallowed	

Completion Codes:

- VOLUME_BAD_NAME
  Volume name does not exist

Clearing Reservations of a Volume

Clear reservations of a volume.

```
reservation_clear  vol=VolName
```

Parameters:

Name	Type	Description	Mandatory	Default
vol	Object name	Name of the volume to clear reservations of.	Y	

Clear reservations of a volume.

Example:

```
xcli  -u -c XIV1 reservation_clear vol=Vol1
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Allowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- VOLUME_BAD_NAME
  Volume name does not exist

Chapter 4. LUN Mapping Management

All the LUN mapping commands are obsolete and were replaced by the clustering commands.

The sections are listed as follows:

The following commands are no longer in use:

- map_create (The concept of mapping has been replaced by clusters. See map_volume and cluster_create.)
- map_add_vol (The concept of mapping has been replaced by clusters. See map_volume.)
- map_remove_vol (The concept of mapping has been replaced by clusters. See unmap_volume.)
- map_link_host (The concept of mapping has been replaced by clusters. See map_volume.)
- map_unlink_host (The concept of mapping has been replaced by clusters.)
- map_list (The concept of mapping has been replaced by clusters.)
- map_list_luns (The concept of mapping has been replaced by clusters. See mapping_list)
- map_rename (The concept of mapping has been replaced by clusters.)
- map_duplicate (The concept of mapping has been replaced by clusters.)
- map_delete (The concept of mapping has been replaced by clusters. See map_volume.)
- map_set_special_type (The concept of mapping has been replaced by clusters. See special_type_set.)

Chapter 5. Volume Snapshot Management

The following sections describe the XIV Command Line Interface (XCLI) for snapshot management.

See also:

- Volume Management
- Consistency Group Management
- Storage Pool Management

The sections are listed as follows:

- snapshot_create(Creates a snapshot of an existing volume.)
- snapshot_duplicate(Duplicates an existing snapshot.)
- snapshot_restore(Restores a master volume or a snapshot from one of its associated snapshots.)
- snapshot_change_priority(Changes a snapshot's deletion priority.)
- snapshot_list(Lists snapshot information.)
- snapshot_delete(Deletes a snapshot.)

Creating a Snapshot

Creates a snapshot of an existing volume.

```
snapshot_create vol=VolName  < [name=SnapshotName]
[delete_priority=del_value] > | < overwrite=SnapshotName >
```

Parameters:

Name	Type	Description	Mandatory	Default
vol	Object name	Name of the volume to snapshot.	Y	
name	Object name	Name of the new snapshot.	N	Auto-generated name.
delete_priority	Integer	The deletion priority of the volume's snapshot.	N	1
overwrite	Object name	Name of an existing snapshot to be overwritten with the current volume content.	N	

This command creates a new snapshot for an existing volume, which is referred to as the snapshot's master volume. The snapshot's content is the same as the master volume at the exact point in time that the snapshot was created. The snapshot remains unchanged, although the master volume keeps changing after the

snapshot is created. Upon a successful completion of this command, the snapshot is created and assigned a name that can later be used by other commands. The name doesn't have to be new. It could be of an already existing snapshot (in such a case, the already existing snapshot is overriden).

A write operation can be processed at the exact time of the snapshot creation, meaning that the write operation request was sent to the system before the command was executed, while the write was acknowledged after the command was executed. In this case, the content of the snapshot is not deterministic and may either contain the original value before the write operation or the new value after the write operation. In fact, the snapshot's data may even contain a mixture of the two, where some blocks are equal to the volume before the write operation and other blocks are equal to the value after the write operation.

The new snapshot is initially locked for changes.

The snapshot that is created acts like a regular volume, except for the differences described below:

- The snapshot's name is either automatically generated from its master volume's name or given as a parameter to the command. It can later be changed without altering the snapshot's modification state.
- Upon successful completion of the command, the system assigns a unique SCSI ID to the snapshot. The creation time of the snapshot is set to the current time and is never changed until the snapshot is deleted.
- The size of the snapshot is the same as its master volume's size, but no storage space is reserved for the snapshot. This means that the functionality of the snapshot is not guaranteed. When the snapshot's Storage Pool is exhausted, the snapshot may be deleted.
- The snapshot's lock state is initially set to "locked", and as long as it is not "unlocked", the snapshot remains an exact image of the master volume at creation time and can be the source for a restore operation. The modification state of the snapshot is initially set to "unmodified".

During creation, the snapshot's deletion priority can be set explicitly, or it is automatically set to the default value. The deletion priority determines which snapshots will be deleted first when the storage pool runs out of snapshot storage. This may happen due to the redirect-on-write mechanisms which share unchanged data between volumes and their snapshots, as well as between snapshots of the same volume.

The Auto Delete Priority is from the list (1-4), as follows:

- 1 = Is last to be deleted automatically ("1" is the default set by the system)
- ...
- 4 = Is first to be deleted automatically

The snapshot is associated with its master volume and this association cannot be broken or changed as long as the snapshot exists.

The `overwrite` option copies the current content of the volume into one of its existing snapshots (set as an input argument). The overwritten snapshot keeps the same SCSI serial number and same mapping, so hosts maintain a continuous mapping to the snapshot, without any need for a rescan or similar operation. The overwritten snapshot must be an existing snapshot of the given volume. The overwritten snapshot can't be part of a Snapshot Group.

This command fails when no snapshot space is defined in the Storage Pool the master volume belongs to.

This command fails if the master volume is a slave of a synchronous mirroring, and it is currently inconsistent due to either a re-synchronization or an initialization process.

Example:

```
xcli -c Nextra1 snapshot_create vol=DBVolume name=DBVolume.snapshot1
                              delete_priority=2
```

Output:

```
Command executed successfully.
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Conditionally Allowed	The volume is mapped to a host or a cluster associated with the user. If a snapshot overwrite is used, the target snapshot must be one created by a server administrator.
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- VOLUME_BAD_NAME
  Volume name does not exist
- MAX_VOLUMES_REACHED
  Maximum number of volumes already defined
  Maximum number of volumes already defined
- SNAPSHOT_ILLEGAL_PRIORITY
  Snapshot priorities allowed are an integer between 1 and 4, inclusive
  Snapshot deletion priority value is illegal
- VOLUME_IS_SNAPSHOT
  Operation is not permitted on snapshots
  Operation is not permitted on snapshots
- VOLUME_EXISTS
  Volume name already exists
- VOLUME_BAD_PREFIX
  Volume name has a reserved prefix
- VOLUME_DATA_MIGRATION_UNSYNCHRONIZED
  Data Migration has not completed to this volume
- OVERWRITE_SNAPSHOT_BAD_NAME

Target overwrite snapshot name does not exist
Snapshot name does not exist

- OVERWRITE_SNAPSHOT_IS_MASTER
  Target overwrite must be a snapshot
  Cannot override Master Volume
- SNAPSHOT_OVERWRITE_MISMATCH
  Specified snapshot is not a snapshot of the specified volume
- SNAPSHOT_IS_PART_OF_SNAPSHOT_GROUP
  Snapshot is part of a Snapshot Group
- SNAPSHOT_IS_INTERNAL
  Target snapshot cannot be internal
  Internal snapshots cannot be mapped, modified or deleted
- POOL_SNAPSHOT_LIMIT_REACHED
  Maximum number of snapshots already created
  Storage Pool snapshot usage has reached its limit
- VOLUME_IS_NOT_CONSISTENT_SLAVE
  Operation not allowed on slave volume that is not consistent.

Duplicating a Snapshot

Duplicates an existing snapshot.

```
snapshot_duplicate snapshot=SnapshotName [ name=SnapshotName ]
```

Parameters:

Name	Type	Description	Mandatory	Default
snapshot	Object name	The name of the snapshot to duplicate.	Y	
name	Object name	Name of the new snapshot to be generated.	N	Automatically generated name.

This command duplicates an existing snapshot. The newly created snapshot is initially locked for changes and is associated with the master volume of the existing snapshot. The content of the newly created snapshot is identical to the content of the source snapshot.

It is useful to duplicate a snapshot before unlocking it for write operations. The duplicate snapshot can be used as a logical backup of the data in case the write operation caused logical data corruption.

Upon successful completion of the command, a new duplicate snapshot is created.

The duplicated snapshot is identical to the source snapshot. It has the same creation time and behaves as if it was created at the exact same moment that the source snapshot was created from the same master volume.

The duplicate snapshot's name is either automatically generated from its master volume's name or provided as a parameter. It can later be changed without altering its modification state.

A snapshot can be duplicated multiple times. A duplicated snapshot can be the source for further duplications.

Example:

```
xcli -c Nextra1 snapshot_duplicate
snapshot=DBVolume.snapshot1 name=DBVolume.snapshot1.copy
```

Output:

```
Command executed successfully.
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Conditionally Allowed	The master volume of the snapshot is mapped to a host or cluster associated with the user and the snapshot was created by the application administrator.
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- VOLUME_BAD_NAME
  Volume name does not exist
- MAX_VOLUMES_REACHED
  Maximum number of volumes already defined
  Maximum number of volumes already defined
- VOLUME_IS_NOT_A_SNAPSHOT
  Operation is permitted only on snapshots
- VOLUME_EXISTS
  Volume name already exists
- SNAPSHOT_IS_PART_OF_SNAPSHOT_GROUP
  Snapshot is part of a Snapshot Group
- VOLUME_BAD_PREFIX
  Volume name has a reserved prefix

Restoring a Volume from a Snapshot

Restores a master volume or a snapshot from one of its associated snapshots.

```
snapshot_restore snapshot=SnapshotName [ target_snapshot=SnapshotName ]
```

Parameters:

Name	Type	Description	Mandatory	Default
snapshot	Object name	Name of the snapshot with which to restore its master volume, or snapshot.	Y	
target_snapshot	Object name	Snapshot to be restored.	N	Restore the master volume.

This command restores the data of a master volume from one of its associated snapshots.

Issuing a restore command logically copies the data of the source snapshot onto its volume. The volume's data is therefore restored to the state that it was at the time that the snapshot was created. If the volume was resized after the snapshot was created, the restore operation resizes the volume back to its original size.

All the snapshots associated with the volume are left unchanged during a restore operation.

It is possible to snapshot the volume before restoring it, so that the generated snapshot can be used and the data is not lost.

It is impossible to restore a master or slave volume of a mirroring definition from a snapshot.

It is possible to restore another snapshot (the target snapshot) from the source snapshot. The target snapshot must be a snapshot of the same volume as the source snapshot. The target snapshot's content and size will be identical to the source snapshot's content and size. The target snapshot's lock/unlock status will remain as it was.

Example:

```
xcli -c Nextra1 snapshot_restore snapshot=DBVolume.snapshot1
```

Output:

```
Command completed successfully
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	

User Category	Permission	Condition
Application administrator	Conditionally Allowed	Both target and source are snapshots of the same master volume. This master volume is mapped to a host or cluster associated with the user, and the target snapshot was created by an application administrator.
Read-only users	Disallowed	
Technicians	Disallowed	

Warnings:

- ARE_YOU_SURE_YOU_WANT_TO_RESTORE_SNAPSHOT
  Are you sure you want to restore the volume from snapshot *Snapshot*?

Completion Codes:

- VOLUME_HAS_DATA_MIGRATION
  Data Migration is defined for this volume
- VOLUME_BAD_NAME
  Volume name does not exist
- VOLUME_IS_NOT_A_SNAPSHOT
  Operation is permitted only on snapshots
- NOT_ENOUGH_SPACE
  No space to allocate volume
- SNAPSHOT_IS_PART_OF_SNAPSHOT_GROUP
  Snapshot is part of a Snapshot Group
- VOLUME_HAS_MIRROR
  Mirror is defined for this volume
- VOLUME_LOCKED
  Volume is locked
- SNAPSHOTS_BELONG_TO_DIFFERENT_MASTERS
  Target snapshot and source snapshot should be snapshots of the same volume
- TARGET_SNAPSHOT_BAD_NAME
  Target snapshot name does not exist
- TARGET_SNAPSHOT_IS_PART_OF_SNAPSHOT_GROUP
  Target snapshot is part of a Snapshot Group
- TARGET_SNAPSHOT_IS_MASTER
  Target snapshot is a master volume
- TARGET_SNAPSHOT_SAME_AS_SNAPSHOT
  Source snapshot cannot be the target snapshot

Changing a Snapshot Deletion Priority

Changes a snapshot's deletion priority.

```
snapshot_change_priority snapshot=SnapshotName  delete_priority=del_value
```

Parameters:

Name	Type	Description	Mandatory	Default
snapshot	Object name	Name of the snapshot whose delete_priority is to be changed.	Y	
delete_priority	Integer	The priority for deleting the volume's snapshot.	Y	

This command changes the priority of the deletion of an existing snapshot. The deletion priority determines which snapshots are deleted first when the system runs out of snapshot storage.

The Auto Delete Priority can have a value between 1 and 4, as follows:

- 1 = Is the last to be deleted automatically ("1" is the default set by the system)
- ...
- 4 = Is the first to be deleted automatically

Example:

```
xcli -c Nextra1 snapshot_change_priority snapshot=DBVolume.snapshot1 delete_priority=4
```

Output:

```
Command completed successfully
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Conditionally Allowed	The master volume of the snapshot is mapped to a host or cluster associated with the user and the snapshot was created by the application administrator.
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- VOLUME_IS_NOT_A_SNAPSHOT
  Operation is permitted only on snapshots
- SNAPSHOT_ILLEGAL_PRIORITY
  Snapshot priorities allowed are an integer between 1 and 4, inclusive
  Snapshot deletion priority value is illegal
- SNAPSHOT_IS_INTERNAL
  Target snapshot cannot be internal

Internal snapshots cannot be mapped, modified or deleted

- SNAPSHOT_IS_PART_OF_SNAPSHOT_GROUP
  Snapshot is part of a Snapshot Group
- VOLUME_BAD_NAME
  Volume name does not exist

Listing Snapshot Information

Lists snapshot information.

```
snapshot_list  vol=VolName
```

Parameters:

Name	Type	Description	Mandatory	Default
vol	Object name	List of all the snapshots of this volume.	Y	

This command lists snapshot information for all the snapshots of a specified volume.

Id	Name	Description	Default Position
name	Name		1
size	Size (GB)		2
size_MiB	Size (MiB)		
master_name	Master Name		3
cg_name	Consistency Group		4
pool_name	Pool		5
creator	Creator		6
capacity	Capacity (blocks)		
modified	Modified		
sg_name	Snapshot Group Name		
delete_priority	Deletion Priority		
locked	Locked		
serial	Serial Number		
snapshot_time	Snapshot Creation Time		
snapshot_of	Snapshot of		
sg_snapshot_of	Snapshot of Snap Group		
wwn	WWN		
locked_by_pool	Locked by Pool		
used_capacity	Used Capacity (GB)		7

Example:

```
xcli -c Nextra1 snapshot_list vol=DBVolume
```

Output:

```
Name                  Size (GB)  Master Name  Consistency Group  Pool
DBVolume.clone1       2508       DBVolume                        default
DBVolume.clone1.copy  2508       DBVolume                        default
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Allowed	
Read-only users	Allowed	
Technicians	Disallowed	

Deleting a Snapshot

Deletes a snapshot.

```
snapshot_delete snapshot=SnapshotName
```

Parameters:

Name	Type	Description	Mandatory	Default
snapshot	Object name	Snapshot to be deleted.	Y	

This command deletes a snapshot. It cannot be used to delete a master volume or to delete a snapshot which is mapped to a host or cluster.

Example:

```
xcli -c Nextra1 snapshot_delete snapshot=DBVolume.snapshot1
```

Output:

```
Command completed successfully
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Conditionally Allowed	The master volume of the snapshot is mapped to a host or cluster associated with the user and the snapshot was created by the application administrator.

User Category	Permission	Condition
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- VOLUME_BAD_NAME
  Volume name does not exist
- VOLUME_IS_NOT_A_SNAPSHOT
  Operation is permitted only on snapshots
- SNAPSHOT_IS_INTERNAL
  Target snapshot cannot be internal
  Internal snapshots cannot be mapped, modified or deleted
- SNAPSHOT_IS_PART_OF_SNAPSHOT_GROUP
  Snapshot is part of a Snapshot Group
- SNAPSHOT_IS_MAPPED
  Snapshot that is mapped to a host cannot be deleted

Chapter 6. Consistency Group Management

The following sections describe the XIV Command Line Interface (XCLI) for Consistency Group management. In addition to the commands listed below, see also: Snapshotting a Consistency Group on the Snapshot Set Management chapter.

- Volume Management
- Volume Snapshot Management
- Storage Pool Management

The sections are listed as follows:

- cg_create(Creates a Consistency Group.)
- cg_add_vol(Adds a volume to a Consistency Group.)
- cg_remove_vol(Removes a volume from a Consistency Group.)
- cg_list(Lists Consistency Groups.)
- cg_rename(Renames Consistency Groups.)
- cg_delete(Deletes a Consistency Group.)

The following commands are no longer in use:

- cg_restore (The command has been replaced by the snap_group_restore command)

Creating Consistency Groups

Creates a Consistency Group.

```
cg_create cg=CgName  pool=PoolName
```

Parameters:

Name	Type	Description	Mandatory	Default
cg	Object name	Name of the Consistency Group.	Y	
pool	Object name	Storage Pool of the Consistency Group.	Y	

This command creates a Consistency Group. The Storage Pool of the Consistency Group must be specified. A Consistency Group is a group of volumes that can be snapshotted at the same point in time. This is essential when snapshotting several volumes that are used by the same application or by applications that interact with each other in order to generate a consistent set of snapshots.

The name of the Consistency Group must be unique in the system.

The Consistency Group is initially empty, containing no volumes.

A Consistency Group always belongs to a specific Storage Pool. All the volumes in the Consistency Group belong to the same Storage Pool as the Consistency Group.

Example:

```
xcli -c Nextra1 cg_create cg=DBgroup
```

Output:

```
Command completed successfully
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- CONS_GROUP_NAME_EXISTS
  Consistency Group name already exists
- MAX_CONS_GROUPS_REACHED
  Maximum number of Consistency Groups already defined
- POOL_DOES_NOT_EXIST
  Storage Pool does not exist

Adding a Volume to a Consistency Group

Adds a volume to a Consistency Group.

```
cg_add_vol cg=CgName vol=VolName
```

Parameters:

Name	Type	Description	Mandatory	Default
cg	Object name	Name of a Consistency Group.	Y	
vol	Object name	Name of the volume to add.	Y	

This command adds a volume to a Consistency Group.

Future snapshot groups created from this Consistency Group also include a snapshot of this added volume.

This command fails if the volume and the Consistency Group belong to different Storage Pools.

This command fails if the volume already belongs to a Consistency Group.

This command fails if the newly added volume is mirrored on a different system than the remote system of one of the mirrored volumes in the Consistency Group.

Example:

```
xcli -c Nextra1 cg_add_vol cg=DBGroup vol=DBLog
```

Output:

```
Command completed successfully
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- CONS_GROUP_BAD_NAME
  Consistency Group name does not exist
  Consistency Group name does not exist
- VOLUME_BAD_NAME
  Volume name does not exist
- VOLUME_IS_SNAPSHOT
  Operation is not permitted on snapshots
  Operation is not permitted on snapshots
- VOLUME_BELONGS_TO_CG
  Volume belongs to a Consistency Group
- MAX_VOLUMES_IN_CG_REACHED
  Consistency Group contains maximum number of volumes
- VOLUME_BAD_POOL
  Volume belongs to a different Storage Pool
- VOLUME_DATA_MIGRATION_UNSYNCHRONIZED
  Data Migration has not completed to this volume
- LOCAL_IS_SLAVE
  Local mirror peer is not the master
- TARGET_NOT_CONNECTED
  There is currently no connection to the target system
- MIRROR_IS_NOT_SYNCHRONIZED
  Mirror is not synchronized

Removing a Volume from a Consistency Group

Removes a volume from a Consistency Group.

```
cg_remove_vol vol=VolName
```

Parameters:

Name	Type	Description	Mandatory	Default
vol	Object name	Name of the volume to be removed.	Y	

This command removes a volume from a Consistency Group.

A Consistency Group's name is deduced from the volume name. A unique name is ensured because each volume belongs to only a single Consistency Group. Future snapshot groups created from this Consistency Group will not include a snapshot which is associated with the removed volume.

All the snapshots of the removed volume that were created as part of this Consistency Group remain associated with the Consistency Group and its snapshot groups. Therefore, if the user re-adds the volume to the Consistency Group, it will seem as is if the volume was never removed from the Consistency Group.

This command succeeds even if the volume is not included in any Consistency Group.

Example:

```
xcli -c Nextra1 cg_remove_vol vol=DBLog
```

Output:

```
Command completed successfully
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- VOLUME_BAD_NAME
  Volume name does not exist
- VOLUME_IS_SNAPSHOT
  Operation is not permitted on snapshots
  Operation is not permitted on snapshots

- TARGET_NOT_CONNECTED
  There is currently no connection to the target system
- LOCAL_IS_SLAVE
  Local mirror peer is not the master

Listing Consistency Groups

Lists Consistency Groups.

```
cg_list [ cg=CgName ]
```

Parameters:

Name	Type	Description	Mandatory	Default
cg	Object name	Name of a Consistency Group.	N	All

This command lists all Consistency Groups.

When a specific Consistency Group name is specified, then only that Consistency Group is listed.

Id	Name	Description	Default Position
name	Name		1
pool	Pool Name		2

Example:

```
xcli -c Nextra1 cg_list cg=DBgroup
```

Output:

```
Name     Pool Name
DBgroup  default
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Allowed	
Read-only users	Allowed	
Technicians	Disallowed	

Renaming Consistency Groups

Renames Consistency Groups.

```
cg_rename cg=CgName new_name=NewName
```

Parameters:

Name	Type	Description	Mandatory	Default
cg	Object name	The name of the Consistency Group to be renamed.	Y	
new_name	Object name	The new name of the Consistency Group.	Y	

This command renames a Consistency Group.

The new name of the Consistency Group must be unique in the system.

This command succeeds even if the new name is identical to the current name.

Example:

```
xcli -c Nextra1 cg_rename cg=DBgroup new_name=DBvolumes
```

Output:

```
Command completed successfully
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- CONS_GROUP_BAD_NAME
  Consistency Group name does not exist
  Consistency Group name does not exist
- CONS_GROUP_NAME_EXISTS
  Consistency Group name already exists

Deleting a Consistency Group

Deletes a Consistency Group.

```
cg_delete cg=CgName
```

Parameters:

Name	Type	Description	Mandatory	Default
cg	Object name	Name of the Consistency Group to delete.	Y	

This command deletes a Consistency Group.

This command fails if the Consistency Group is not empty, meaning that it still contains volumes.

All snapshot groups associated with the Consistency Group are disbanded, meaning that the snapshots contained in these snapshot groups become independent snapshots.

Example:

```
xcli -c Nextra1 cg_delete cg=DBvolumes
```

Output:

```
Command completed successfully
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- CONS_GROUP_BAD_NAME
  Consistency Group name does not exist
  Consistency Group name does not exist
- CONS_GROUP_NOT_EMPTY
  Local Consistency Group is not empty and cannot be mirrored
  Operation is only allowed on an empty Consistency Group
- CONS_GROUP_HAS_MIRROR
  Consistency Group has mirroring defined for it.

Chapter 7. Snapshot Set Management

The following sections describe the XIV Command Line Interface (XCLI) for Snapshot Set management. Use the cg_snapshot_create command to create the snapshot group.

See also:

- Volume Management
- Volume Snapshot Management
- Consistency Group Management

The sections are listed as follows:

- cg_snapshots_create(Creates a snapshot group of a Consistency Group.)
- snap_group_duplicate(Duplicates an existing snapshot group.)
- snap_group_restore(Restores the master volumes of a Consistency Group, or a snapshot group from one of its associated snapshot groups.)
- snap_group_change_priority(Changes the deletion priority of a snapshot group.)
- snap_group_disband(Disbands a snapshot group into independent snapshots.)
- snap_group_lock(Locks a snapshot group by locking all its snapshot.)
- snap_group_unlock(Unlocks a snapshot group by unlocking all its snapshots.)
- snap_group_list(Lists all snapshot groups or a specific one.)
- snap_group_rename(Renames a snapshot group.)
- snap_group_delete(Deletes a snapshot group and all its snapshots.)

Snapshotting a Consistency Group

Creates a snapshot group of a Consistency Group.

```
cg_snapshots_create
cg=CgName < [ snap_group=SnapshotGroupName ]
[ delete_priority=del_value ] > | <overwrite=SnapshotGroupName>
```

Parameters:

Name	Type	Description	Mandatory	Default
cg	Object name	Name of a Consistency Group.	Y	
snap_group	Object name	Name for the new snapshot group.	N	Automatically generated name.
delete_priority	Integer	The priority for deleting this volume when the system runs out of snapshot space.	N	1

Name	Type	Description	Mandatory	Default
overwrite	Object name	Existing snapshot group which is overwritten with current content.	N	

This command creates a consistent snapshot group from a Consistency Group, which includes a snapshot for each of the volumes contained in the Consistency Group.

Logically, this command is comprised of the following steps:

- Suspending all I/O activity on all the volumes in the group and waiting for all pending I/Os to complete
- Creating a snapshot for each of the volumes in the group
- Resuming I/O activity on all the volumes

The main advantage of using this command (as opposed to a manual procedure) is that all snapshots are taken at the same point in time, thus ensuring that they are consistent with each other.

The snapshots in the created snapshot group are consistent with respect to each other in the following manner:

- All snapshots are created synchronously at the same point in time.
- With respect to this point in time, all I/Os to the Consistency Group's volumes that were completed previously are recorded in the snapshot's image.
- All I/Os that were completed afterwards are not recorded in the snapshot's image.

All the snapshots in the snapshot group are also associated with the Consistency Group in addition to having their regular attributes.

The name of the snapshot group is either automatically generated or provided in the command line.

The delete priority of the snapshots in the snapshot group can also be provided (see Creating a Snapshot). The delete priority controls which snapshots or snapshot groups are deleted first when the system runs out of space for snapshots.

The `overwrite` option causes the current content of the Consistency Group to be copied into one of its existing snapshot groups (the argument of overwrite). The snapshots of the overwritten snapshot group keep the same SCSI serial number and same mapping, so hosts maintain a continuous mapping of the snapshots, without any need for a rescan or similar operation. The overwrite snapshot group must be an existing snapshot group of the given Consistency Group.

This command fails if no snapshot space is defined for the Storage Pool containing the Consistency Group.

This command fails if one or more of the volumes in the Consistency Group is a slave of a synchronous mirroring, and it is currently inconsistent due to either a re-synchronization or an initialization process.

Example:

```
xcli -c Nextra1 cg_snapshots_create cg=DBgroup snap_group=DBbackupdaily
```

Output:

```
Command completed successfully.
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Conditionally Allowed	At least one of the volumes in the group is mapped to a host or cluster associated with the user. If a Snapshot Group overwrite is used, then the target Snapshot Group must be one created by a server administrator.
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- MAX_VOLUMES_REACHED
  Maximum number of volumes already defined
  Maximum number of volumes already defined
- CONS_GROUP_BAD_NAME
  Consistency Group name does not exist
  Consistency Group name does not exist
- SNAPSHOT_GROUP_BAD_NAME
  Snapshot Group name does not exist
- SNAPSHOT_GROUP_NAME_EXISTS
  Snapshot Group name already exists
- CONS_GROUP_EMPTY
  Operation is not allowed on an empty Consistency Group
- CONS_GROUP_MISMATCH
  Snapshot Group does not match Consistency Group volumes
- OVERWRITE_SNAPSHOT_GROUP_DOES_NOT_BELONG_TO_GIVEN_GROUP
  Snapshot Group belongs to another Consistency Group
- POOL_SNAPSHOT_LIMIT_REACHED
  Maximum number of snapshots already created
  Storage Pool snapshot usage has reached its limit
- VOLUME_IS_NOT_CONSISTENT_SLAVE
  Operation not allowed on slave volume that is not consistent.

Duplicating a Snapshot Group

Duplicates an existing snapshot group.

```
snap_group_duplicate snap_group=SnapshotGroupName [ new_snap_group=NewName ]
```

Parameters:

Name	Type	Description	Mandatory	Default
snap_group	Object name	Name of the snapshot group to be duplicated.	Y	
new_snap_group	Object name	Name of the newly generated snapshot group.	N	Autogenerated name.

This command duplicates the specified snapshot group. This is functionally equivalent to duplicating all the snapshots in the snapshot group using Duplicating a Snapshot and creating a new snapshot group that contains all the generated snapshots.

The name of the new snapshot group is either specified as a parameter or generated automatically.

You may refer to Duplicating a Snapshot for more details about the snapshot duplication operation.

Example:

```
xcli -c Nextra1 snap_group_duplicate snap_group=DBbackup new_snap_group=DBbackup_copy
```

Output:

```
Command completed successfully
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Conditionally Allowed	At least one of the volumes in the master Consistency Group is mapped to a host or cluster associated with the user and Snapshot Group was created by a server administrator
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- SNAPSHOT_GROUP_BAD_NAME
  Snapshot Group name does not exist
- MAX_VOLUMES_REACHED
  Maximum number of volumes already defined
  Maximum number of volumes already defined
- SNAPSHOT_GROUP_NAME_EXISTS
  Snapshot Group name already exists

Restoring a Volume from a Snapshot Group

Restores the master volumes of a Consistency Group, or a snapshot group from one of its associated snapshot groups.

```
snap_group_restore snap_group=SnapshotGroupName [ target_snap_group=SnapGroupName ]
```

Parameters:

Name	Type	Description	Mandatory	Default
snap_group	Object name	Name of the snapshot group from which to restore its master volumes.	Y	
target_snap_group	Object name	Snapshot group to be restored.	N	

Using this command is equivalent to restoring all the volumes in the Consistency Group or all the snapshots in the target snapshot group from their snapshots in the snapshot group.

It is possible to restore a snapshot group from a snapshot group.

The Consistency Group or the target Snapshot Group must contain the exact same volumes that it contained when the snapshot group was generated. Each volume that is added to the Consistency Group after the creation of the snapshot group must be removed from the Consistency Group before restoration is performed.

See Restoring a Volume from a Snapshot for more information about the restoring.

Example:

```
xcli -c Nextra1 snap_group_restore snap_group=DBbackup_copy
```

Output:

```
Command completed successfully.
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Conditionally Allowed	Both target and source are snapshots groups of the same master Consistency Group, where at least one of the master volumes in this Consistency Group is mapped to a host or cluster associated with the user, and the target Snapshot Group was created by an application administrator.
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- VOLUME_HAS_DATA_MIGRATION
  Data Migration is defined for this volume
- SNAPSHOT_GROUP_BAD_NAME
  Snapshot Group name does not exist
- CONS_GROUP_MISMATCH
  Snapshot Group does not match Consistency Group volumes
- NOT_ENOUGH_SPACE
  No space to allocate volume
- VOLUME_HAS_MIRROR
  Mirror is defined for this volume
- VOLUME_LOCKED
  Volume is locked
- TARGET_SNAPSHOT_GROUP_BAD_NAME
  Target Snapshot Group name does not exist
- SNAPSHOT_GROUP_MISMATCH
  Snapshot Group does not match target Snapshot Group
- TARGET_SNAPSHOT_GROUP_SAME_AS_SOURCE
  Target Snapshot Group is the same as Snapshot Group

Changing Snapshot Group Deletion Priority

Changes the deletion priority of a snapshot group.

```
snap_group_change_priority snap_group=SnapshotGroupName  delete_priority=del_value
```

Parameters:

Name	Type	Description	Mandatory	Default
snap_group	Object name	Name of the snapshot group whose delete_priority is to be changed.	Y	
delete_priority	Integer	Priority according to which this snapshot group is deleted.	Y	

This command changes the priority of the deletion of an existing snapshot group. Similarly to snapshots, the system determines which of the snapshot groups is deleted first when it runs out of snapshot storage, in accordance with the redirect-on-write mechanism. When the system runs out of space, it deletes the snapshot or snapshot group with the highest deletion priority, and among them the unmapped snapshots or snapshot groups, and among them the snapshot or snapshot group which was created first.

See Changing a Snapshot Deletion Priority for more details about the valid deletion priority values and their meaning.

Example:

```
xcli -c Nextra1 snap_group_change_priority snap_group=DBbackup delete_priority=4
```

Output:

```
Command completed successfully.
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Conditionally Allowed	At least one of the volumes in the master Consistency Group is mapped to a host or cluster associated with the user and Snapshot Group was created by a server administrator
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- SNAPSHOT_GROUP_BAD_NAME
  Snapshot Group name does not exist
- SNAPSHOT_ILLEGAL_PRIORITY
  Snapshot priorities allowed are an integer between 1 and 4, inclusive

Snapshot deletion priority value is illegal

Disbanding a Snapshot Group

Disbands a snapshot group into independent snapshots.

```
snap_group_disband snap_group=SnapshotGroupName
```

Parameters:

Name	Type	Description	Mandatory	Default
snap_group	Object name	Snapshot group to be disbanded.	Y	

This command disbands the snapshot group into independent snapshots. After executing this command the snapshots can be individually deleted, restored, unlocked, duplicated and so on. The snapshot group does not exist anymore after this command. The snapshots retain the same names (snap_group_name.volumename).

Example:

```
xcli -c Nextra1 snap_group_disband snap_group=DBbackup_copy
```

Output:

```
Command completed successfully.
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Conditionally Allowed	At least one of the volumes in the master Consistency Group is mapped to a host or cluster associated with the user and Snapshot Group was created by a server administrator
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- SNAPSHOT_GROUP_BAD_NAME
  Snapshot Group name does not exist

Locking a Snapshot Group

Locks a snapshot group by locking all its snapshot.

```
snap_group_lock snap_group=CloneSetName
```

Parameters:

Name	Type	Description	Mandatory	Default
snap_group	Object name	Name of the snapshot group to be locked.	Y	

This command locks the snapshot group by locking all its snapshots. This is functionally equivalent to locking all snapshots individually (through executing Locking a Volume on each snapshot). You may refer to the documentation of Locking a Volume for a description of locking behavior.

Example:

```
xcli -c Nextra1 snap_group_lock snap_group=DBbackup
```

Output:

```
Command completed successfully
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Conditionally Allowed	At least one of the volumes in the master Consistency Group is mapped to a host or cluster associated with the user and Snapshot Group was created by a server administrator
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- SNAPSHOT_GROUP_BAD_NAME
  Snapshot Group name does not exist

Unlocking a Snapshot Group

Unlocks a snapshot group by unlocking all its snapshots.

```
snap_group_unlock snap_group=SnapshotGroupName
```

Parameters:

Name	Type	Description	Mandatory	Default
snap_group	Object name	Name of the snapshot group to be unlocked.	Y	

This command unlocks a snapshot group by unlocking all its snapshots. This is functionally equivalent to executing Unlocking a Volume on each snapshot. You may refer to the documentation of Unlocking a Volume for a description of unlocking behavior.

Example:

```
xcli -c Nextra1 snap_group_unlock snap_group=DBbackup
```

Output:

```
Command completed successfully
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Conditionally Allowed	At least one of the volumes in the master Consistency Group is mapped to a host or cluster associated with the user and Snapshot Group was created by a server administrator
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- SNAPSHOT_GROUP_BAD_NAME
  Snapshot Group name does not exist

Listing Snapshot Groups

Lists all snapshot groups or a specific one.

```
snap_group_list [ snap_group=CloneSetName | cg=CgName ]
```

Parameters:

Name	Type	Description	Mandatory	Default
snap_group	Object name	Name of a specific snapshot group to be listed.	N	All snapshot groups.

Name	Type	Description	Mandatory	Default
cg	Object name	List all the snapshot groups of this Consistency Group.	N	All snapshot groups.

This command lists snapshot groups. When a snapshot group name is specified, then only that specific snapshot group is listed. When a Consistency Group name is specified, then the snapshot groups of this Consistency Group are listed.

Id	Name	Description	Default Position
name	Name		1
cg	CG		2
snapshot_time	Snapshot Time		3
locked	Locked		
modified	Modified		
delete_priority	Deletion Priority		4

Example:

```
xcli -c Nextra1 snap_group_list cg=DBvolumes
```

Output:

```
Name           CG         Clone Time
DBbackup       DBvolumes  2007-01-03 17:46:29
DBbackupdaily  DBvolumes  2007-01-03 17:49:36
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Allowed	
Read-only users	Allowed	
Technicians	Disallowed	

Completion Codes:

- CONS_GROUP_DOES_NOT_EXIST
  Consistency Group name does not exist
- SNAPSHOT_GROUP_BAD_NAME
  Snapshot Group name does not exist

Renaming a Snapshot Group

Renames a snapshot group.

```
snap_group_rename snap_group=SnapshotGroupName new_name=NewName
```

Parameters:

Name	Type	Description	Mandatory	Default
snap_group	Object name	Name of the snapshot group to be renamed.	Y	
new_name	Object name	New name for the snapshot group.	Y	

This command renames a snapshot group.

Example:

```
xcli -c Nextra1 snap_group_rename snap_group=DBbackup new_name=DBBackupweekly
```

Output:

```
Command completed successfully
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Conditionally Allowed	At least one of the volumes in the master Consistency Group is mapped to a host or cluster associated with the user and Snapshot Group was created by a server administrator
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- SNAPSHOT_GROUP_BAD_NAME
  Snapshot Group name does not exist
- SNAPSHOT_GROUP_NAME_EXISTS
  Snapshot Group name already exists

Deleting a Snapshot Group

Deletes a snapshot group and all its snapshots.

```
snap_group_delete snap_group=SnapGroupName
```

Parameters:

Name	Type	Description	Mandatory	Default
snap_group	Object name	Name of the snapshot group to be deleted.	Y	

This command deletes the snapshot group, as well as all the snapshots contained in the snapshot group. You may refer to the documentation of Deleting a Snapshot for more information about deleting snapshots.

If one of the members of the snapshot group is mapped to a host, then the snapshot group cannot be deleted.

Example:

```
xcli -c Nextra1 snap_group_delete snap_group=DBBackupweekly
```

Output:

```
Command completed successfully.
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Conditionally Allowed	At least one of the volumes in the master Consistency Group is mapped to a host or cluster associated with the user and Snapshot Group was created by a server administrator
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- SNAPSHOT_GROUP_BAD_NAME
  Snapshot Group name does not exist
- SNAPSHOT_IS_MAPPED
  Snapshot that is mapped to a host cannot be deleted

Chapter 8. Storage Pool Management

The following sections describe the XIV Command Line Interface (XCLI) for Storage Pool management.

See also:

- Volume Management
- Volume Snapshot Management
- Consistency Group Management

The sections are listed as follows:

- pool_create(Creates a Storage Pool.)
- pool_delete(Deletes a Storage Pool.)
- pool_list(Lists all Storage Pools or the specified one.)
- vol_move(Moves a volume and all its snapshot from one Storage Pool to another.)
- cg_move(Moves a Consistency Group, all its volumes and all their snapshots and Snapshot Sets from one Storage Pool to another.)
- pool_rename(Renames a specified Storage Pool.)
- pool_resize(Resizes a Storage Pool.)
- pool_change_config(Changes the lock behavior of the storage Pool. See a detailed description in .)

Creating Storage Pools

Creates a Storage Pool.

```
pool_create pool=PoolName < size=GB | < hard_size=GB soft_size=GB > >
snapshot_size=GB [ lock_behavior=<read_only|no_io> ]
```

Parameters:

Name	Type	Description	Mandatory	Default
pool	Object name	Name of the Storage Pool.	Y	
size	Positive integer	Size of the Storage Pool (in gigabytes).	N	
hard_size	Positive integer	Hard size of the Storage Pool (actual physical capacity).	N	
soft_size	Positive integer	Soft size of the Storage Pool. Soft_size cannot be less than hard_size.	N	
snapshot_size	Positive integer	Space allocated for snapshots.	Y	

Name	Type	Description	Mandatory	Default
lock_behavior	Enumeration	Sets whether and how the Pool is locked upon space depletion.	N	read_only

This command creates a Storage Pool. The name of the Storage Pool must be unique in the system. Upon creation, the Storage Pool is initially empty and does not contain volumes.

The size of the Storage Pool is specified as an integer multiple of 10^9 bytes, but the actual size of the created Storage Pool is rounded up to the nearest integer multiple of $16x2^{30}$ bytes. The Size parameter is used when hard size and soft size are identical (no thin provisioning). If only the size is specified, then hard_size and soft_size are identical to the size. Otherwise, a Storage Pool with thin provisioning is created.

The created pool has the following values:

- create_last_consistent_snapshot=yes - meaning the volumes of this pool can be mirrored
- protected_snapshot_priority=2 - managing the way last_consistent snapshot are preserved

When a Storage Pool is defined, the new Storage Pool's capacity is reduced from the system's free space (hard and soft). This operation fails if the system hard or soft free space does not have free capacity of at least the size of the new Storage Pool. The sum of the capacities of all the Storage Pools in the system, together with the free space, is always equal to the entire system capacity available for the user.

The system allows the assignment of the entire available capacity to user created Storage Pools, while leaving the free space at zero size.

Both hard and soft sizes are subtracted from the free hard/soft space.

For thin provisioned Storage Pools, the lock_behavior parameter sets whether and how the Pool is locked upon space depletion. The Pool can be locked for write, or for both read and write.

Example:

```
xcli -c Nextra1 pool_create pool=DBPool size=1000
```

Output:

```
Command completed successfully.
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	

User Category	Permission	Condition
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- POOL_NAME_EXISTS

  Storage Pool name already assigned to another Storage Pool
- MAX_POOLS_REACHED

  Maximum number of Storage Pools already defined
- NO_HARD_SPACE

  The system does not have enough free hard space for the requested Storage Pool hard size
- NO_SOFT_SPACE

  The system does not have enough free soft space for the requested Storage Pool soft size
- SOFT_SIZE_SMALLER_THAN_HARD_SIZE

  Soft size must be equal or larger than hard size
- HARD_SIZE_SMALLER_THAN_SNAPSHOT_SIZE

  Snapshot size must be equal or smaller than hard size
- NO_SPACE

  The system does not have enough free space for the requested Storage Pool size

Deleting a Storage Pool

Deletes a Storage Pool.

```
pool_delete pool=PoolName
```

Parameters:

Name	Type	Description	Mandatory	Default
pool	Object name	The name of the Storage Pool to delete.	Y	

This command deletes a Storage Pool.

This command fails if the Storage Pool is not empty, meaning that it still contains volumes.

The capacity of the deleted Storage Pool is added to the free space.

Example:

```
xcli -c Nextra1 pool_delete pool=ERPPool
```

Output:

```
Command completed successfully
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Warnings:

- ARE_YOU_SURE_YOU_WANT_TO_DELETE_POOL
  Are you sure you want to delete Storage Pool *Pool*?

Completion Codes:

- POOL_DOES_NOT_EXIST
  Storage Pool does not exist
- POOL_HAS_CG
  Storage Pool has Consistency Groups defined
- POOL_IN_USE
  Storage Pool has volumes allocated in it

Listing Storage Pools

Lists all Storage Pools or the specified one.

```
pool_list [ pool=PoolName ]
```

Parameters:

Name	Type	Description	Mandatory	Default
pool	Object name	Name of a Storage Pool.	N	All pools.

This command lists all Storage Pools.

When the `pool` parameter is provided, only the specified Storage Pool is listed.

Example:

```
xcli -c Nextra1 pool_list
```

Output:

```
Name     Size (GB)  Empty Space (GB)
default  24292      9225
DBPool   1013       1013
```

Id	Name	Description	Default Position
name	Name		1
soft_size	Size (GB)		2
soft_size_MiB	Size (MiB)		
hard_size	Hard Size (GB)		3
hard_size_MiB	Hard Size (MiB)		
snapshot_size	Snapshot Size (GB)		4
snapshot_size_MiB	Snapshot Size (MiB)		
empty_space_soft	Empty Space (GB)		5
empty_space_soft_MiB	Empty Space (MiB)		
empty_space_hard	Empty Hard Space (GB)		
empty_space_hard_MiB	Empty Hard Space (MiB)		
used_by_volumes	Used by Volumes (GB)		6
used_by_volumes_MiB	Used by Volumes (MiB)		
used_by_snapshots	Used by Snapshots (GB)		7
used_by_snapshots_MiB	Used by Snapshots (MiB)		
creator	Creator		
locked	Locked		8
lock_behavior	Lock Behavior		
create_last_consistent_snapshot	Create Last Consistent Snapshot		
protected_snapshot_priority	Protected Snapshots Priority		

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Allowed	
Read-only users	Allowed	
Technicians	Disallowed	

Moving a Volume between Storage Pools

Moves a volume and all its snapshot from one Storage Pool to another.

```
vol_move vol=VolName pool=PoolName
```

Parameters:

Name	Type	Description	Mandatory	Default
vol	Object name	Name of the volume to move.	Y	
pool	Object name	Name of the Storage Pool to which to move.	Y	

This command moves a volume and all its snapshots from one Storage Pool to another.

When moving a master volume from one Storage Pool to another, all of its snapshots are moved together with it to the destination Storage Pool.

This command fails when trying to move a snapshot of a volume on its own. This command can fail as a result of either a lack of soft or of hard space.

This command only succeeds if the destination Storage Pool has enough free storage capacity to accommodate the volume and its snapshots. The exact amount of storage capacity allocated from the destination Storage Pool is released at the source Storage Pool.

A volume which belongs to a Consistency Group cannot be moved without the entire Consistency Group. You may use Moving Consistency Groups between Storage Pools to move a complete Consistency Group from one Storage Pool to another.

Example:

```
xcli -c Nextra1 vol_move vol=DBLog pool=DBPool
```

Output:

```
Command completed successfully.
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- VOLUME_BAD_NAME
  Volume name does not exist
- POOL_DOES_NOT_EXIST
  Storage Pool does not exist
- NOT_ENOUGH_SPACE

No space to allocate volume

- NOT_ENOUGH_HARD_SPACE
  No space to allocate for volume's current usage
- VOLUME_IS_SNAPSHOT
  Operation is not permitted on snapshots
  Operation is not permitted on snapshots
- VOLUME_BELONGS_TO_CG
  Volume belongs to a Consistency Group
- NOT_ENOUGH_SNAPSHOT_SPACE
  Snapshot usage will exceed snapshot limit

Moving Consistency Groups between Storage Pools

Moves a Consistency Group, all its volumes and all their snapshots and Snapshot Sets from one Storage Pool to another.

```
cg_move cg=CgName pool=PoolName
```

Parameters:

Name	Type	Description	Mandatory	Default
cg	Object name	Name of the Consistency Group to move.	Y	
pool	Object name	Name of the Storage Pool to which to move.	Y	

This command moves a Consistency Group from one Storage Pool to another.

All volumes in the Consistency Group are moved, all snapshot groups of this Consistency Group are moved and all snapshots of the volumes are moved.

This command can fail as a result of either a lack of soft or of hard space.

Example:

```
xcli -c Nextra1 cg_move cg=DBGroup pool=DBPool
```

Output:

```
Command completed successfully.
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	

User Category	Permission	Condition
Technicians	Disallowed	

Completion Codes:

- CONS_GROUP_BAD_NAME
  Consistency Group name does not exist
  Consistency Group name does not exist
- POOL_DOES_NOT_EXIST
  Storage Pool does not exist
- NOT_ENOUGH_SPACE
  No space to allocate volume
- NOT_ENOUGH_HARD_SPACE
  No space to allocate for volume's current usage
- NOT_ENOUGH_SNAPSHOT_SPACE
  Snapshot usage will exceed snapshot limit

Renaming a Storage Pool

Renames a specified Storage Pool.

```
pool_rename pool=PoolName new_name=NewName
```

Parameters:

Name	Type	Description	Mandatory	Default
pool	Object name	Name of the Storage Pool.	Y	
new_name	Object name	New name of the Storage Pool.	Y	

This command renames a Storage Pool.

The new name of the Storage Pool must be unique in the system.

This command succeeds even if the new name is identical to the current name.

Example:

```
xcli -c Nextra1 pool_rename pool=DBPool new_name=ERPPool
```

Output:

```
Command completed successfully.
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	

User Category	Permission	Condition
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- POOL_DOES_NOT_EXIST
  Storage Pool does not exist
- POOL_NAME_EXISTS
  Storage Pool name already assigned to another Storage Pool

Resizing a Storage Pool

Resizes a Storage Pool.

```
pool_resize pool=PoolName [size=GB | < hard_size=GB soft_size=GB >]
[snapshot_size=GB]
```

Parameters:

Name	Type	Description	Mandatory	Default
pool	Object name	Name of the Storage Pool to resize.	Y	
size	Positive integer	New size of the Storage Pool (in gigabytes). Using this option specifies that the hard_size and the soft_size are identical.	N	
hard_size	Positive integer	Hard size of the Storage Pool (actual physical capacity).	N	
soft_size	Positive integer	Soft size of the Storage Pool (maximal size of capacity seen by the hosts, used for thin provisioning).	N	
snapshot_size	Integer	Sets a new value for the limit on snapshot capacity usage of the Storage Pool.	N	Leave unchanged.

This command resizes a Storage Pool.

The new size of the Storage Pool is specified as an integer multiple of 10^9 bytes, but the actual size of the created Storage Pool is rounded up to the nearest integer multiple of $16x2^{30}$ bytes.

This command can be used to both increase and to decrease a Storage Pool size.

You have the option to either specify both hard_size and soft_size or to only specify size (which specifies that the hard_size and the soft_size are identical).

Capacity accounting is performed in respect to the free space.

When increasing a Storage Pool size, the command succeeds only if the free space holds enough free capacity to allow such an increase in size.

When decreasing a Storage Pool size, the command succeeds only if the Storage Pool itself holds enough free capacity to allow such a reduction in size.

If the new size equals the current size, the command succeeds without changes to the Storage Pool.

This command fails if either the current Storage Pool's size (hard or soft) cannot be decreased or if free space (hard or soft) cannot be decreased.

Example:

```
xcli -c Nextra1 pool_resize pool=DBPool size=1300
```

Output:

```
Command Executed Successfully.
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- POOL_DOES_NOT_EXIST

  Storage Pool does not exist
- NO_SOFT_SPACE

  The system does not have enough free soft space for the requested Storage Pool soft size
- SOFT_SIZE_SMALLER_THAN_HARD_SIZE

  Soft size must be equal or larger than hard size
- HARD_SIZE_SMALLER_THAN_SNAPSHOT_SIZE

  Snapshot size must be equal or smaller than hard size
- POOL_SOFT_TOO_SMALL

Requested soft size is smaller than the sum of sizes of volumes in the Storage Pool

- POOL_TOO_SMALL

  Storage Pool usage exceeds requested size
- POOL_HARD_TOO_SMALL

  Storage Pool usage exceeds requested hard size
- NO_SPACE

  The system does not have enough free space for the requested Storage Pool size
- NO_HARD_SPACE

  The system does not have enough free hard space for the requested Storage Pool hard size
- POOL_SNAPSHOT_SIZE_TOO_SMALL

  Storage Pool snapshot usage exceeds requested snapshot size

Changing Pool Limitation

Changes the lock behavior of the storage Pool. See a detailed description in .

```
pool_change_config pool=PoolName
lock_behavior=<read_only|no_io>
```

Parameters:

Name	Type	Description	Mandatory	Default
pool	Object name	Name of the Storage Pool.	Y	
lock_behavior	Enumeration	Sets whether and how the Pool is locked upon space depletion.	Y	

This command changes the Storage Pool snapshot limitation policy.

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- POOL_DOES_NOT_EXIST

  Storage Pool does not exist

Chapter 9. System Management

The following sections describe the XIV Command Line Interface (XCLI) for system management.

The sections are listed as follows:

- shutdown(Shuts down the system.)
- state_list(Shows the current operational state of the system.)
- version_get(Prints the current version of the system.)
- help(Prints system help.)
- dns_test(Tests the DNS (Domain Naming Service).)
- config_get(Shows the values of configuration parameters.)
- config_set(Sets configuration parameters.)
- vpd_config_get(Shows the values of VPD parameters.)
- vpd_config_set(Sets the values of VPD parameters.)
- cod_list(Displays current consumed capacity of the system.)
- system_capacity_list(Lists the system's capacities (both hard and soft).)
- time_set(Sets the system's time in YYYY-MM-DD.HH:MM:SS format.)
- time_list(Shows the current system time.)
- timezone_list(Lists all optional time zones.)
- timezone_set(Sets the time zone of the system.)
- upgrade_download(Initiates the downloading of a new software version.)
- upgrade_download_cancel(Cancels an upgrade download process.)
- upgrade_system(Upgrades the software version of the system.)
- upgrade_force_on(Forces the system to continue the upgrade process.)
- upgrade_get_status(Displays status of upgrade process.)
- upgrade_validate_prerequisites(Validates the ability to upgrade to a specified system version)
- upgrade_abort_ongoing(Aborts a system upgrade process.)

The following commands are no longer in use:

- upgrade_download_status (The data will be incorporated in upgrade_get_status.)

Shutting Down

Shuts down the system.

```
shutdown [ emergency=<yes|no> ]
```

Parameters:

Name	Type	Description	Mandatory	Default
emergency	Enumeration	Sets the system to shut down within a timeout even if some of the disks could not be saved, much like in an emergency shutdown performed when the system loses power.	N	no

This command shuts down the system.

The system stops serving hosts, de-stages all information to disks and then turns itself off. If the "emergency" option is specified, the system will shut down within a timeout period. *NOTE: USING THIS OPTION MAY CAUSE DATA LOSS.*

Example:

```
xcli -c XIV1 shutdown
```

Output:

```
Command Executed Successfully.
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Warnings:

- ARE_YOU_SURE_YOU_WANT_TO_SHUT_DOWN
  Are you sure you want to shut down the machine and all its components?

Completion Codes:

- COMMAND_IS_NOT_VALID_IN_CURRENT_SYSTEM_STATE
  The requested command can not be invoked in the current system state
- SHUTDOWN_CANNOT_HAPPEN_DURING_SYSTEM_UPGRADE
  The system cannot shut down while an upgrade is being performed.
  Troubleshooting: Contact support
- SYSTEM_UPGRADE_CANNOT_BE_CANCELED
  Upgrade is already underway and cannot be canceled.

- SYSTEM_UPGRADE_CANCELED_BECAUSE_OF_NODE_FAILURE_DURING_UPGRADE
  Last upgrade was canceled because a node failed while the upgrade process was running
- HOT_UPGRADE_IS_NOT_ONGOING
  Hot upgrade is not currently ongoing
- SYSTEM_UPGRADE_NOT_RUNNING
  Upgrade is not underway

Listing Operational State

Shows the current operational state of the system.

```
state_list
```

This command shows the current operational state of the system.

Id	Name	Description	Default Position
category	Category		1
value	Value		2

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Allowed	
Read-only users	Allowed	
Technicians	Allowed	

Printing the Current System Version

Prints the current version of the system.

```
version_get
```

This command prints the current version of the system.

Id	Name	Description	Default Position
system_version	Version		1

Example:

```
xcli -c Nextra1 version_get
```

Output:

```
Version
10.1
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Allowed	
Read-only users	Allowed	
Technicians	Allowed	

Printing Help

Prints system help.

```
help [ category=CategoryName | search=SearchString | command=CommandName ]
     [ format=<short|full> ]
```

Parameters:

Name	Type	Description	Mandatory	Default
category	String	Category name.	N	
search	String	Search string.	N	
command	String	Command name.	N	
format	Enumeration	Output format for command help.	N	short.

This command prints the help as follows:

- No parameters - Lists all the commands with their short descriptions, grouped by categories.
- Category - Lists all the commands in the category, with their short descriptions.
- Search - Lists the short descriptions of all the commands in which the search string appears in their name or short description.
- Command with short output (default for command) - Prints the command name and short description.
- Command with full output (default when used in XIV-internal mode) - Prints the command name, short description, syntax, list of parameters and their description, types and default values. If output is table, prints all possible table columns.

Example:

```
xcli -c Nextra1 help category=volume
```

Output:

```
Category  Name       Description

volume    vol_by_id  Prints the volume name given its SCSI
                     serial number

volume    vol_copy   Copies a source Volume onto a target Volume.

volume    vol_create Creates a new volume.

volume    vol_delete Deletes a Volume

volume    vol_format Formats a Volume.

volume    vol_list   Lists all Volumes, or a specific one.

volume    vol_lock   Locks a Volume, so that it is read-only.

volume    vol_rename Renames a Volume

volume    vol_resize Resize a Volume

volume    vol_unlock Unlocks a Volume, so that it is no longer read-only,
                     and can be written to.
```

Id	Name	Description	Default Position
category	Category		1
name	Name		2
access_control	Access Control		
syntax	Syntax		
fields	Fields		
description	Description		3
example	Example		

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Allowed	
Read-only users	Allowed	
Technicians	Allowed	

Testing the DNS

Tests the DNS (Domain Naming Service).

```
dns_test name=HostName
```

Parameters:

Name	Type	Description	Mandatory	Default
name		Name of the host to be resolved.	Y	

This command attempts to translate the DNS name into an IP address. Translation is attempted through each of the defined DNS servers.

This command fails if no DNS servers are defined. A failure of the translation from a name to an IP address is not considered a failure of the command.

The result of each defined DNS server is displayed.

Id	Name	Description	Default Position
name	Name		1
primary_ip	IP (Primary DNS)		2
secondary_ip	IP (Secondary DNS)		3

Example:

```
xcli -c Nextra1 dns_test name=hermes.xiv
```

Output:

```
Name         IP (Primary DNS)   IP (Secondary DNS)
------------ ------------------ --------------------
hermes.xiv   212.143.102.243    Not Found
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Allowed	

Completion Codes:

- DNS_SERVER_NOT_DEFINED
  No DNS servers are defined

Printing Configuration Parameters

Shows the values of configuration parameters.

```
config_get [ name=ParamName ]
```

Parameters:

Name	Type	Description	Mandatory	Default
name	String	Name of parameter to print.	N	All parameters.

Id	Name	Description	Default Position
name	Name		1
value	Value		2

This command shows the name and value of the specified configuration parameter or of all of them, if no parameter is provided.

The values of the following parameters can be shown:

- *dns_primary*- IP address of the master DNS server.
- *dns_secondary* - IP address of the slave DNS server.
- *email_reply_to_address* - Reply-to address to be used when sending emails. This is useful for troubleshooting errors in email addresses.
- *email_sender_address* - Email address used as the sender's address when sending email messages.
- *email_subject_format* - Controls the formatting of the email subject line. The tags {severity}, {description} and {system_name} can be used to insert the event's data. System default is "{severity}: {description}"
- *iscsi_name* - iSCSI initiator name. Used when configuring non-XIV system for Data Migration over iSCSI.
- *machine_model* -
- *machine_serial_number* -
- *machine_type* -
- *ntp_server* - IP address or DNS name of the NTP server.
- *snmp_community* -
- *snmp_contact* -
- *snmp_location* - SNMP location as shown in the SNMP MIB. (.1.3.6.1.2.1.1.4.0).
- *snmp_contact* - SNMP contact as shown in the SNMP MIB. (.1.3.6.1.2.1.1.4.0).
- *snmp_trap_community* -
- *support_center_port_type* -
- *system_id* - Unique system identifier (equivalent to a serial number).
- *system_name* -

Example:

```
xcli -c Nextra1 config_get
```

Output:

```
Name                          Value
------------------------ -----------------------------------
email_sender_address     support@ibm.com
email_reply_to_address   storage@ibm.com
dns_primary              10.0.0.10
dns_secondary
iscsi_name               iqn.2005-10.com.xivstorage:010140
system_name              IBM Storage System
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Allowed	
Read-only users	Allowed	
Technicians	Allowed	

Completion Codes:

- UNRECOGNIZED_CONFIG_PARAMETER
  Unrecognized configuration parameter: *'argument'*
  Troubleshooting: Use a valid configuration parameter as an input.

Setting Configuration Parameters

Sets configuration parameters.

```
config_set name=ParamName value=ParamValue
```

Parameters:

Name	Type	Description	Mandatory	Default
name	String	Name of the parameter to set.	Y	
value	String	Value of the parameter.	Y	

This command sets the values of configuration parameters.

The values of the following parameters can be set:

- *dns_master* - IP address of the master DNS server.
- *dns_slave* - IP address of the slave DNS server.
- *email_sender_address* - Email address used as the sender's address when sending email messages. Once set, this parameter can not be set to a null value.
- *email_reply_to_address* - Reply-to address to be used when sending emails. This is useful for troubleshooting errors in email addresses.
- *system_name* - Name used as the sender's name when sending email messages.
- *defaultuser* - Default user to be used if no user is specified for the CLI. If null, a user must be specified.
- *snmp_sysname* - SNMP system name as shown in the SNMP MIB. (.1.3.6.1.2.1.1.5.0)
- *snmp_location* - SNMP location as shown in the SNMP MIB. (.1.3.6.1.2.1.1.6.0)
- *snmp_contact* - SNMP contact as shown in the SNMP MIB. (.1.3.6.1.2.1.1.4.0)
- *email_subject_format* - Controls the formatting of the email subject line. The tags {severity}, {description} and {system_name} can be used to insert the event's data. System default is "{severity}: {description}".
- *ntpserver* - IP address or DNS name of the NTP server.
- *snmp_community* - community used for SNMP queries of the system.

- *snmp_trap_community* - community used for SNMP traps sent by the system.

Example:

```
xcli -c Nextra1 config_set name=dns_secondary value=10.0.0.119
```

Output:

```
Command completed successfully
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Allowed	

Completion Codes:

- UNRECOGNIZED_CONFIG_PARAMETER
  Unrecognized configuration parameter: *'argument'*
  Troubleshooting: Use a valid configuration parameter as an input.
- READ_ONLY_CONFIG_PARAMETER
  Configuration parameter: *'argument'* is read-only
  Troubleshooting: You can't modify read-only parameters.

Showing Values of VPD Parameters.

Shows the values of VPD parameters.

```
vpd_config_get  [ name=ParamName ]
```

Parameters:

Name	Type	Description	Mandatory	Default
name	String	Name of the parameter to print.	N	All parameters.

Id	Name	Description	Default Position
name	Name		1
value	Value		2

Shows the values of VPD parameters. See Setting VPD Parameters for a full list of available settings.

Example:

```
xcli  -u -c XIV1 vpd_config_get name=site.city
```

Output:

```
Name        Value
----------- --------
site.city   Gotham
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Setting VPD Parameters

Sets the values of VPD parameters.

```
vpd_config_set  name=ParamName value=ParamValue
```

Parameters:

Name	Type	Description	Mandatory	Default
name	String	Name of the parameter to set.	Y	
value	String	Value of the parameter.	Y	

This command sets the following values of VPD parameters, where only the `name` is mandatory.:

- customer.name
- customer.primary_contact.calling_hours
- customer.primary_contact.email
- customer.primary_contact.mobile_phone
- customer.primary_contact.name
- customer.primary_contact.office_phone
- customer.primary_contact.time_zone
- customer.secondary_contact.calling_hours
- customer.secondary_contact.email
- customer.secondary_contact.mobile_phone
- customer.secondary_contact.name
- customer.secondary_contact.office_phone
- customer.secondary_contact.time_zone
- hardware_info.hw_ats_monitoring
- hardware_info.hw_ats_type

- hardware_info.hw_cable_bundle
- hardware_info.hw_door
- hardware_info.hw_patch_panel
- hardware_info.hw_patch_panel_label
- hardware_info.hw_power_cable_config
- hardware_info.hw_rps
- interface_config.model
- machine_model
- machine_type
- main_ibm_contact.calling_hours
- main_ibm_contact.email
- main_ibm_contact.mobile_phone
- main_ibm_contact.name
- main_ibm_contact.office_phone
- main_ibm_contact.time_zone
- non_mutable_vpd_info.original_flashed_version
- remote_support.customer_contact.calling_hours
- remote_support.customer_contact.email
- remote_support.customer_contact.mobile_phone
- remote_support.customer_contact.name
- remote_support.customer_contact.office_phone
- remote_support.customer_contact.time_zone
- remote_support.modem_phone_number
- remote_support.primary_ibm_ip
- remote_support.secondary_ibm_ip
- remote_support.special_instructions
- remote_support.vpn_ip_1
- remote_support.vpn_ip_2
- site.building_location
- site.city site.country
- site.name
- site.postal_code
- site.state
- site.street_address
- system_info.sys_ec_level
- system_info.sys_hw_level

Example:

```
xcli  -u -c XIV1 vpd_config_set name= value=
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	

User Category	Permission	Condition
Read-only users	Disallowed	
Technicians	Disallowed	

Displaying Current Consumed Capacity of the System

Displays current consumed capacity of the system.

```
cod_list  [ name=ParamName ]
```

Parameters:

Name	Type	Description	Mandatory	Default
name	String	Name of parameter to display.	N	All parameters

Id	Name	Description	Default Position
name	Name		1
value	Value		2

This command displays current consumed capacity of a given system.

Example:

```
xcli  -u -c XIV1 cod_list
```

Output:

```
Name                    Value
----------------------- ----------------
consumed_capacity       1039
date                    2009-05-27
dst                     yes
machine_model           A14
machine_serial_number   MN00013
machine_type            2810
system_id               13
system_name             XIV MN00013a
time                    10:13:31
timezone                Asia/Jerusalem
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Allowed	
Read-only users	Allowed	
Technicians	Allowed	

Showing System Capacity, Free Space and Spares

Lists the system's capacities (both hard and soft).

```
system_capacity_list
```

This command shows the various types of hard and soft system capacities, free space and spare capacities.

Id	Name	Description	Default Position
soft	Soft		1
hard	Hard		2
free_hard	Free Hard		3
free_soft	Free Soft		4
spare_modules	Spare Modules		5
spare_disks	Spare Disks		6
target_spare_modules	Target Spare Modules		7
target_spare_disks	Target Spare Disks		8

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Allowed	
Read-only users	Allowed	
Technicians	Disallowed	

Setting the System's Time

Sets the system's time in YYYY-MM-DD.HH:MM:SS format.

```
time_set time=Timestamp
```

Parameters:

Name	Type	Description	Mandatory	Default
time		New current time.	Y	

This command sets the system's time.

Example:

```
xcli  -u  -c Nextra1 time_set time=2006-05-04.03:02:01
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Allowed	

Completion Codes:

- SYSTEM_TIME_NOT_CHANGED
  System time was not changed
  Troubleshooting: Please try again.
- BAD_TIMESTAMP
  Timestamp cannot be deciphered

Showing the Current Time

Shows the current system time.

```
time_list
```

This command shows the current time, date and time zone.

Id	Name	Description	Default Position
time	Time		1
date	Date		2
timezone	Time Zone		3
dst	Daylight Saving Time		4

Example:

```
xcli -u -c Nextra1 time_list
```

Output:

```
Time       Date         Time Zone        Daylight Saving Time
---------- ------------ ---------------- ----------------------
10:09:47   2008-02-19   Asia/Jerusalem   no
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Allowed	
Read-only users	Allowed	
Technicians	Allowed	

Listing Optional Time Zones

Lists all optional time zones.

```
timezone_list
```

This command lists the optional time zones.

Standard POSIX time zones are used. http://www.timeanddate.com/worldclock/ provides a full description of all time zones.

Id	Name	Description	Default Position
name	Timezone		1

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Allowed	
Read-only users	Allowed	
Technicians	Allowed	

Setting the Time Zone

Sets the time zone of the system.

```
timezone_set timezone=TimeZone
```

Parameters:

Name	Type	Description	Mandatory	Default
timezone	String	New time zone of the system.	Y	

This command sets the new time zone of the system. See Listing Optional Time Zones for a complete list of optional time zones.

Standard POSIX time zones are used. http://www.timeanddate.com/worldclock/ provides a full description of all time zones.

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Allowed	

Completion Codes:

- BAD_TIMEZONE_NAME
  Timezone is not recognized by the system

Initiating Download of a New Software Version

Initiates the downloading of a new software version.

```
upgrade_download version=VersionId interface_type=<laptop|management|remote_support_module|vpn>
[ repository_ip=DownloadServer ]
```

Parameters:

Name	Type	Description	Mandatory	Default
version	String	Version number of the new software to be downloaded.	Y	
interface_type	Enumeration	Type of IP interface where the repository IP resides.	Y	
repository_ip		Network server used as the source for the new version.	N	

This command initiates the downloading of a new software version.

The command only fails if there is another download process in effect. All other failures are reported asynchronously in upgrade_status command.

Access Control:

User Category	Permission	Condition
Storage administrator	Disallowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Allowed	

Completion Codes:

- SYSTEM_UPGRADE_IS_ALREADY_RUNNING
  Upgrade is already running
- NO_ACTIVE_PORTS_OF_SPECIFIED_ROLE
  None of the ports of the specified role is active.
- UPGRADE_DOWNLOAD_COULD_NOT_BE_STARTED
  Failed starting upgrade download of an unknown reason.
- REPOSITORY_IP_MUST_BE_SUPPLIED_FOR_DOWNLOAD

Repository IP must be supplied for download if the interface type is Management or VPN.

- NO_PORTS_OF_SPECIFIED_ROLE

  The system does not have any ports of the specified roles.
- PORT_ROLE_IS_INVALID

  Specified port role is invalid.

Canceling an Upgrade Download Process

Cancels an upgrade download process.

```
upgrade_download_cancel
```

This command cancels an upgrade download process.

Access Control:

User Category	Permission	Condition
Storage administrator	Disallowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Allowed	

Completion Codes:

- SYSTEM_UPGRADE_CANNOT_BE_CANCELED

  Upgrade is already underway and cannot be canceled.
- SYSTEM_UPGRADE_NOT_RUNNING

  Upgrade is not underway

Upgrading a System

Upgrades the software version of the system.

```
upgrade_system upgrade_type=<hot|cold|utilities_only> [ allow_downtime=<yes|no> ]
```

Parameters:

Name	Type	Description	Mandatory	Default
upgrade_type	Enumeration	The type can can be ″hot″ (for hot upgrade), ″cold″ (for cold upgrade), and ″utilities_only″ (for warm upgrade). If the requested upgrade type is ″cold″, another argument has to be supplied: ″allow_downtime″ with the value of ″yes″.	Y	
allow_downtime	Boolean	If the requested upgrade type is ″cold″, this argument has with the value of ″yes″.	N	no

This command upgrades the software version of the system.

The command fails whenever:

- The download has not been initiated
- The download has not been completed
- A specific module or disk has failed - in this case, the upgrade operation stops in maintenance mode (″safemode″).

Upgrade is not allowed (to some versions) whenever:

- There are active data migration processes.
- The system contains a volume that is mirrored, set to primary and in mandatory mode.

Both cases return UPGRADE_CAN_NOT_START.

Access Control:

User Category	Permission	Condition
Storage administrator	Disallowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Allowed	

Completion Codes:

- SYSTEM_UPGRADE_NOT_FINISHED_DOWNLOADING

  System cannot switch to new version until the new version is downloaded to all modules of the system.

Troubleshooting: Make sure that all nodes finished downloading the new software version

- PRE_UPGRADE_VALIDATIONS_ALREADY_RUNNING

  Pre-upgrade validations cannot be invoked since the system is already in the process of performing those validations
- NO_LIVE_ADMIN_SERVER_FOUND

  Could not invoke pre-upgrade script because no module can run cli commands
- PRE_UPGRADE_VALIDATION_FAILED

  One or more of the conditions for starting an upgrade sequence failed
- UPGRADE_RELATED_SCRIPT_ALREADY_RUNNING

  Pre-upgrade or post-upgrade script cannot be invoked since an upgrade related script is currently running
- SYSTEM_UPGRADE_COLD_UPGRADE_WHEN_DOWNTIME_DISALLOWED

  Cold upgrade was requested but system downtime was not allowed

  Troubleshooting: Specify another upgrade type, or allow downtime with the proper argument
- SYSTEM_UPGRADE_INCOMPATIBLE_UPGRADE_TYPE_SPECIFIED

  The specified upgrade type specified is incompatibe with the kind of change imposed by the new version's files

  Troubleshooting: Specify the correct upgrade type

Forcing a Continuation of the Upgrade Process

Forces the system to continue the upgrade process.

```
upgrade_force_on
```

This command forces the system to continue the upgrade process.

Access Control:

User Category	Permission	Condition
Storage administrator	Disallowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Allowed	

Warnings:

- ARE_YOU_SURE_YOU_WANT_TO_SHUT_DOWN

  Are you sure you want to shut down the machine and all its components?

Completion Codes:

- COMMAND_IS_NOT_VALID_IN_CURRENT_SYSTEM_STATE

  The requested command can not be invoked in the current system state
- SYSTEM_UPGRADE_NOT_RUNNING

  Upgrade is not underway
- SYSTEM_UPGRADE_CANNOT_BE_CANCELED

  Upgrade is already underway and cannot be canceled.

- HOT_UPGRADE_IS_NOT_ONGOING
  Hot upgrade is not currently ongoing
- SHUTDOWN_CANNOT_HAPPEN_DURING_SYSTEM_UPGRADE
  The system cannot shut down while an upgrade is being performed.
  Troubleshooting: Contact support
- SYSTEM_UPGRADE_CANCELED_BECAUSE_OF_NODE_FAILURE_DURING_UPGRADE
  Last upgrade was canceled because a node failed while the upgrade process was running

Displaying Status of Upgrade Process

Displays status of upgrade process.

```
upgrade_get_status
```

Parameters:

Name	Type	Description	Mandatory	Default

The output of this command displays the status of the upgrade process. The status can be any of the following:

- Downloading - the upgrade package
- Ready for upgrade
- Preparing
- Finalizing upgrade (after the I/Os resumes)

Additional upgrade-related values:

- Requires a reboot
- Update is required
- The number of times the system attempted to stop I/Os
- Time to the next retry
- Abort reason (in case the upgrade was aborted)
- Failed to communication with server
- Server does not have the required software version
- No upgrade path from the current version to the new version
- The new version is a downgrade
- Download done
- Limitations on the upgrade (for example: upgrade to this version is not allowed if data migration is in progress, or if mirroring of a primary volume is defined as mandatory).

In addition, once the download is complete, then a message is displayed on screen, telling whether the upgrade would be hot (no I/O interrupted) or cold (interrupting I/Os).

Example:

```
xcli  -u -c XIV1 upgrade_get_status
```

Output:

```
Name                        Value
--------------------------- ---------------------------------------
io_stopping_attempts_num    -1
is_restart_needed           Unknown
last_upgrade_result         System has never performed an upgrade
last_upgrade_start_time
seconds_for_next_attempt    -1
upgrade_state               Upgrade Not Underway
upgrade_substate            NO_UPGRADE
was_firmware_updated        Unknown
```

Id	Name	Description	Default Position
name	Name		1
value	Value		2

Access Control:

User Category	Permission	Condition
Storage administrator	Disallowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Allowed	

Validating the Prerequisites of an Upgrade to a New Software Version

Validates the ability to upgrade to a specified system version

```
upgrade_validate_prerequisites [ upgrade_type=<hot|cold|utilities_only> ]
```

Parameters:

Name	Type	Description	Mandatory	Default
upgrade_type	Enumeration	The type can can be ″hot″ (for hot upgrade), ″cold″ (for cold upgrade), and ″utilities_only″ (for warm upgrade).	N	

This command runs the prerequisites validation script of the upgrade, and returns its result.

Access Control:

User Category	Permission	Condition
Storage administrator	Disallowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Allowed	

Completion Codes:

- PRE_UPGRADE_VALIDATIONS_ALREADY_RUNNING

  Pre-upgrade validations cannot be invoked since the system is already in the process of performing those validations
- UPGRADE_RELATED_SCRIPT_ALREADY_RUNNING

  Pre-upgrade or post-upgrade script cannot be invoked since an upgrade related script is currently running
- SYSTEM_UPGRADE_NOT_FINISHED_DOWNLOADING

  System cannot switch to new version until the new version is downloaded to all modules of the system.

  Troubleshooting: Make sure that all nodes finished downloading the new software version
- PRE_UPGRADE_VALIDATION_FAILED

  One or more of the conditions for starting an upgrade sequence failed
- NO_LIVE_ADMIN_SERVER_FOUND

  Could not invoke pre-upgrade script because no module can run cli commands

Aborting the Upgrade to a New Software Version

Aborts a system upgrade process.

```
upgrade_abort_ongoing
```

This command aborts the upgrade process of a new software version. In case that I/O are not stopped, the command aborts the upgrade and returns the system to full operation.

Access Control:

User Category	Permission	Condition
Storage administrator	Disallowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Allowed	

Completion Codes:

- HOT_UPGRADE_IS_NOT_ONGOING

  Hot upgrade is not currently ongoing

Chapter 10. Remote Target Connectivity

The following sections describe the XIV Command Line Interface (XCLI) for defining remote target connectivity.

The sections are listed as follows:

- target_define(Defines a new remote target for remote mirroring or data migration.)
- target_config_sync_rates(Changes the target's mirroring configuration.)
- target_rename(Renames a remote target.)
- target_list(Lists a specified remote target definition, or all the target definitions.)
- target_update(Updates the target's configuration.)
- target_delete(Deletes the definition of a specified remote target.)
- target_mirroring_allow(Allows remote mirroring operations initiated from the remote target.)
- target_port_add(Adds a port to a remote target.)
- target_port_delete(Deletes a port from a specified remote target.)
- target_port_activate(Activates a port of a remote target.)
- target_port_deactivate(Deactivates a port of a remote target.)
- target_port_list(List all ports of a target.)
- target_connectivity_define(Defines connectivity between a port on the local storage system and a port on a remote target.)
- target_connectivity_delete(Deletes connectivity between a port on the local storage system and a port on a remote target.)
- target_connectivity_list(Lists all the connectivity definitions of a remote target.)
- target_connectivity_activate(Activates connectivity between a port on the local storage system and a port on a remote target.)
- target_connectivity_deactivate(Deactivates connectivity between a port on the local storage system and a port on a remote target.)

The following commands are no longer in use:

- target_connectivity_test (Connectivity is reported in the target_connectivity_list command.)
- target_portset_create (Port Sets are no longer in use in this version. Ports are associated directly with targets.)
- target_portset_rename (Port Sets are no longer in use in this version. Ports are associated directly with targets.)
- target_portset_delete (Port Sets are no longer in use in this version. Ports are associated directly with targets.)
- target_portset_port_add (Port sets are no longer used. Use target_port_add instead.)
- target_portset_port_delete (Port sets are not in use. Use target_port_delete.)
- target_portset_port_activate (Port sets are not in use any more. Use target_port_activate instead.)
- target_portset_port_deactivate (Portsets are not in use any more. Use target_port_deactivate.)

- target_portset_list (Port Sets are no longer in use in this version. Ports are associated directly with targets.)

Defining a Remote Target

Defines a new remote target for remote mirroring or data migration.

```
target_define target=TargetName  protocol=<FC|iSCSI> [ iscsi_name=iSCSIName ]
[ xiv_features=<yes|no> ]  [ system_id=SystemId ]
```

Parameters:

Name	Type	Description	Mandatory	Default
target	Object name	Local name of the remote target.	Y	
protocol	Enumeration	FC (Fiber Channel) or iSCSI, depending on the communication protocol supported by the remote host.	Y	
iscsi_name	iSCSI initiator name	iSCSI name of the remote target. This field is mandatory for iSCSI hosts.	N	
system_id	String	ID of the remote system. Should be the same as the output of Printing Configuration Parameters of the *system_id* variable on the remote system.	N	
xiv_features	Boolean	Defines the remote system as an XIV system. Non-XIV systems are used only for data migration.	N	Yes

This command defines the communication topology between a local storage system and a remote storage system in order to enable various features, such as remote mirroring. The local storage system can write to or read from the remote storage system or allow the target storage system to write to or read from it.

The first step when defining a new Target Connectivity is to specify the name of the remote storage system and the protocol used to communicate with it. There are

two possible protocols: Fiber Channel (FC) and iSCSI. Each remote target is available through only one of these protocols.

This step only defines the remote system object. No connectivity definitions are defined yet and no communications are performed yet.

Note:

Once you have defined a remote target, the only way to change its protocol type is to delete the remote target and define it again.

Example:

```
xcli -c Nextra1 target_define target=Nextra2 protocol=FC
```

Output:

```
Command executed successfully.
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- MAX_TARGETS_REACHED
  Maximum number of targets already defined
- TARGET_NAME_EXISTS
  Target name is already assigned to another target
- TARGET_ISCSI_MUST_HAVE_A_NAME
  iSCSI Target must have an iscsi_name
- ISCSI_NAME_NOT_ALLOWED_FOR_FC
  FC Target does not have an iscsi_name
- TARGET_BAD_SCSI_TYPE
  Target SCSI type does not exist

Updating the Target Mirroring Configuration

Changes the target's mirroring configuration.

```
target_config_sync_rates target=TargetName [ max_intialization_rate=MaxInitializationRate ]
[ max_resync_rate=MaxResyncRate ]
```

Parameters:

Name	Type	Description	Mandatory	Default
target	Object name	The updated target.	Y	
max_intialization_rate	Positive integer	Specifies the maximum rate for initial synchronization	N	Unchanged
max_resync_rate	Positive integer	Specifies the maximum rate for re-synchronization	N	Unchanged

This command changes the system ID of the remote target. The synchronization rate units are Megabytes per second. The defeault rates are: 30 MB/s for initialization rate, 100 MB/s for resync rate. The default system_id is the value that is set with the config_set command.

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- TARGET_BAD_NAME

  Target name does not exist

Renaming a Remote Target

Renames a remote target.

```
target_rename target=TargetName new_name=NewName
```

Parameters:

Name	Type	Description	Mandatory	Default
target	Object name	Target that is renamed.	Y	
new_name	Object name	New name of the target.	Y	

This command renames an existing target.

Example:

```
xcli -c Nextra1 target_rename target=Nextra2 new_name=Nextra-DRP
```

Output:

```
Command Executed Successfully.
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- TARGET_BAD_NAME
  Target name does not exist
- TARGET_NAME_EXISTS
  Target name is already assigned to another target

Listing Remote Targets

Lists a specified remote target definition, or all the target definitions.

```
target_list [ target=TargetName ]
```

Parameters:

Name	Type	Description	Mandatory	Default
target	Object name	Target name that is listed.	N	All targets

This command lists a specified remote target definition, or all the target definitions. The following is listed for each target: port groups, ports, active/inactive status for each port, and the following mirroring-related values - max initialization rate, max resync rate, and max sync job rate.

Id	Name	Description	Default Position
name	Name		1
scsi_type	SCSI Type		2
connected	Connected		3
xiv_target	XIV Target		
iscsi_name	iSCSI Name		
system_id	System ID		
num_ports	Number of Ports		
creator	Creator		
max_initialization_rate	Max Initialization Rate		
max_resync_rate	Max Resync Rate		

Id	Name	Description	Default Position
max_syncjob_rate	Max Syncjob Rate		

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Allowed	
Read-only users	Allowed	
Technicians	Disallowed	

Updating the Target Configuration

Updates the target's configuration.

```
target_update target=TargetName system_id=SystemId
```

Parameters:

Name	Type	Description	Mandatory	Default
target	Object name	Target to be updated.	Y	
system_id	String	ID of the remote system. Should be the same as the output of Printing Configuration Parameters of the *system_id* variable on the remote system.	Y	

This command changes the system ID of the remote target.

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- TARGET_BAD_NAME
  Target name does not exist

Deleting a Remote Target

Deletes the definition of a specified remote target.

```
target_delete target=TargetName
```

Parameters:

Name	Type	Description	Mandatory	Default
target	Object name	Target that is deleted.	Y	

This command deletes an existing target. A target that contains port definitions cannot be deleted. A target with remote mirroring or Data Migration definitions cannot be deleted.

Example:

```
xcli  -c Nextra1 target_delete target=Nextra2
```

Output:

```
Command executed successfully.
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- TARGET_BAD_NAME
  Target name does not exist
- TARGET_HAS_PORTS
  Ports are defined for this target
- TARGET_HAS_ASSOCIATIONS
  Remote volumes are defined on this target

Allowing Remote Mirroring Access

Allows remote mirroring operations initiated from the remote target.

```
target_mirroring_allow target=TargetName
```

Parameters:

Name	Type	Description	Mandatory	Default
target	Object name	Remote target name.	Y	

This command is performed on a local storage system in order to allow the target storage system permission to read, write, view, create volumes and define existing volumes as slaves. This command is used when allowing remote mirroring operations. Otherwise, the target storage system cannot access the local storage system. This command also allows a remote target to read and write through the SCSI interface.

Once mirroring is allowed, this permission cannot be revoked.

This operation should also be run on the target storage system so that it gives permission to the local storage system to access it.

Note:

This step must be performed before mirroring is defined (mirror_create).

Example:

```
xcli -c Nextra1 target_mirroring_allow target=Nextra2
```

Output:

```
Command Executed Successfully.
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- TARGET_BAD_NAME
  Target name does not exist
- TARGET_BAD_TYPE
  Target machine is not XIV machine

Adding a New Port to a Remote Target

Adds a port to a remote target.

```
target_port_add target=TargetName < ipaddress=IPaddress | fcaddress=wwpn >
```

Parameters:

Name	Type	Description	Mandatory	Default
target	Object name	Remote target to which to add the port.	Y	
ipaddress		IP address of the port on the remote target (for iSCSI type targets only).	N	
fcaddress		FC address of the remote port (for FC type targets only).	N	

This command adds a new port to a specified target. A port can be either FC or iSCSI, and its type must conform to the remote target's communication protocol type.

Specify the IP address or the FC address according to communication protocol of the target.

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- TARGET_BAD_NAME
  Target name does not exist
- PORT_EXISTS
  Port is already defined
- MAX_PORTS_REACHED
  Maximum number of ports already defined in the system
- TARGET_PORT_BAD_ADDRESS
  Remote port address is illegal or does not belong to the remote target

Deleting a Port from a Remote System

Deletes a port from a specified remote target.

```
target_port_delete target=TargetName < ipaddress=IPaddress | fcaddress=wwpn >
```

Parameters:

Name	Type	Description	Mandatory	Default
target	Object name	Remote target from which the port is that is deleted.	Y	
ipaddress		IP address of the port (for iSCSI targets only).	N	
fcaddress		FC address of the remote port (for FC targets only).	N	

This command deletes a port from a specified remote target.

Example:

```
xcli -c Nextra1 target_port_delete
target=Nextra2
fcaddress=10:00:00:17:38:27:ec:11
```

Output:

```
Command executed successfully.
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- TARGET_BAD_NAME
  Target name does not exist
- TARGET_PORT_BAD_ADDRESS
  Remote port address is illegal or does not belong to the remote target
- TARGET_PORT_HAS_CONNECTIVITY
  Port has connectivity defined to it

Activating a Port

Activates a port of a remote target.

```
target_port_activate target=TargetName < ipaddress=IPaddress | fcaddress=wwpn >
```

Parameters:

Name	Type	Description	Mandatory	Default
target	Object name	Remote target of the port.	Y	
ipaddress		IP address of the port on the remote target (iSCSI targets only).	N	
fcaddress		FC address of the port on the remote target (FC targets only).	N	

This command activates a port of a remote target.

Each port in a remote system can be configured as either active or inactive. The system does not use an inactive port. After a port is defined, it is active by default. This command reactivates a port if it was de-activated (by using target_port_deactivate).

This command has no effect, if the port is already active.

Example:

```
xcli -c Nextra1 target_port_activate
target=Nextra2 fcaddress=10:00:00:17:38:27:ec:11
```

Output:

```
Command completed successfully
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- TARGET_PORT_BAD_ADDRESS
  Remote port address is illegal or does not belong to the remote target
- TARGET_BAD_PORT_STATE
  Port is already in requested activation state
- TARGET_BAD_NAME
  Target name does not exist

Deactivating a Port

Deactivates a port of a remote target.

```
target_port_deactivate target=TargetName
            < ipaddress=IPaddress | fcaddress=wwpn >
```

Parameters:

Name	Type	Description	Mandatory	Default
target	Object name	Remote target of the port.	Y	
ipaddress		IP address of the port on the remote target (iSCSI targets only).	N	
fcaddress		FC address of the port on the remote target (FC targets only).	N	

This command deactivates a port.

Each port in a remote system can be configured as either active or in-active. The system does not use an inactive port. After a port is defined, it is active by default. To re-activate a port use Activating a Port.

Example:

```
xcli -c XIV1 target_port_deactivate
target=XIV2
fcaddress=10:00:00:17:38:27:ec:11
```

Output:

```
Command completed successfully
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- TARGET_BAD_NAME
  Target name does not exist
- TARGET_PORT_BAD_ADDRESS

Remote port address is illegal or does not belong to the remote target

- TARGET_BAD_PORT_STATE
  Port is already in requested activation state

Listing the Ports of a Remote Target

List all ports of a target.

```
target_port_list  [ target=TargetName ]
```

Parameters:

Name	Type	Description	Mandatory	Default
target	Object name	Target for which all ports should be listed.	N	All systems

Lists all ports of a remote target.

Id	Name	Description	Default Position
target_name	Target Name		1
scsi_type	Port Type		2
active	Active		3
fc_wwpn	WWPN		4
iscsi_ip_addr	iSCSI Address		5
iscsi_ip_port	iSCSI Port		6

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Allowed	
Read-only users	Allowed	
Technicians	Disallowed	

Defining Connectivity to a Remote Target

Defines connectivity between a port on the local storage system and a port on a remote target.

```
target_connectivity_define target=TargetName
< ipaddress=IPaddress local_ipinterface=IPInterface > |
< fcaddress=wwpn local_port=ComponentId >
```

Parameters:

Name	Type	Description	Mandatory	Default
target	Object name	Remote target of the connectivity definition.	Y	
ipaddress		IP address of the port on the remote target (iSCSI targets only).	N	
local_ipinterface	Object name	Local IP interface to be connected to the remote port (iSCSI only).	N	
fcaddress		FC address of the port on the remote target (FC targets only).	N	
local_port		FC port (FC only).	N	

This command defines connectivity between a port on the local storage system and a port on a remote target.

Connectivity between a local and a target storage system is defined between a specific port on a local storage system and a port on the target storage system.

Each connectivity definition can be either active or inactive. The system does not use an inactive connectivity definition. Target Connectivity is active by default. An option is provided to de-activate (target_connectivity_deactivate) and then re-activate (target_connectivity_activate) it, if required. Target Connectivity can be deleted (Deleting Connectivity to a Remote Target) and a list of Target Connectivity definitions (Listing Target Connectivity Definitions) can be displayed.

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- TARGET_BAD_NAME
  Target name does not exist
- CONN_EXISTS
  Remote port is already connected through this local port
- MAX_CONNECTIONS_REACHED

Maximum number of connections already defined

- COMPONENT_IS_NOT_AN_FC_PORT
  Component must specify an FC port
- BAD_LOCAL_IP_PORT
  An ID of a local IP port must be specified
- COMMAND_NOT_ALLOWED_ON_MANAGEMENT_INTERFACE
  Operation is not allowed on Management IP Interface
- IPINTERFACE_DOES_NOT_EXIST
  IP Interface name does not exist
- TARGET_PORT_BAD_ADDRESS
  Remote port address is illegal or does not belong to the remote target

Deleting Connectivity to a Remote Target

Deletes connectivity between a port on the local storage system and a port on a remote target.

```
target_connectivity_delete target=TargetName
< ipaddress=IPaddress local_ipinterface=IPInterface > |
< fcaddress=wwpn local_port=ComponentId >
```

Parameters:

Name	Type	Description	Mandatory	Default
target	Object name	Remote target of the connectivity definition.	Y	
ipaddress		IP address of the port on the remote target (iSCSI targets only).	N	
local_ipinterface	Object name	Local IP interface that is connected to the remote port (iSCSI only).	N	
fcaddress		FC address of the port on the remote target (FC targets only).	N	
local_port		Port number on the local module (FC only).	N	

This command deletes a Target Connectivity definition. Only a previously defined connectivity definition can be deleted.

Example:

```
xcli -c XIV1 target_connectivity_delete
target=XIV2 local_module=101
```

Output:

```
Command completed successfully
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- TARGET_BAD_NAME
  Target name does not exist
- COMMAND_NOT_ALLOWED_ON_MANAGEMENT_INTERFACE
  Operation is not allowed on Management IP Interface
- CONNECTIVITY_NOT_DEFINED
  Remote port is not connected through this local port
- COMPONENT_IS_NOT_AN_FC_PORT
  Component must specify an FC port
- TARGET_PORT_BAD_ADDRESS
  Remote port address is illegal or does not belong to the remote target
- BAD_LOCAL_IP_PORT
  An ID of a local IP port must be specified
- IPINTERFACE_DOES_NOT_EXIST
  IP Interface name does not exist

Listing Target Connectivity Definitions

Lists all the connectivity definitions of a remote target.

```
target_connectivity_list [ target=TargetName ]
```

Parameters:

Name	Type	Description	Mandatory	Default
target	Object name	Target name that is listed.	N	All targets

This command lists a specified remote target connectivity definition or all the target definitions.

Id	Name	Description	Default Position
target_name	Target Name		1
remote_port_address	Remote Port		2
local_fc_port	FC Port		3
local_ip_port	IP Interface		4
active	Active		5
up	Up		6

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Allowed	
Read-only users	Allowed	
Technicians	Disallowed	

Activating Connectivity to a Remote Target

Activates connectivity between a port on the local storage system and a port on a remote target.

```
target_connectivity_activate target=TargetName
      < ipaddress=IPaddress local_ipinterface=IPInterface > | <
      fcaddress=wwwpn local_port=PortId >
```

Parameters:

Name	Type	Description	Mandatory	Default
target	Object name	Remote target of the connectivity definition.	Y	
ipaddress		IP address of the port on the remote target (iSCSI targets only).	N	
local_ipinterface	Object name	Local IP interface to be connected to the remote port (iSCSI only)	N	
fcaddress		FC address of the port on the remote target (FC targets only).	N	
local_port		Port identifier.	N	

This command activates connectivity after it has been deactivated, if required.

Each connectivity definition can be either active or in-active. The system does not use an inactive connectivity definition. Target Connectivity is active by default.

This command has no effect if the connectivity is already active.

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- TARGET_BAD_NAME
  Target name does not exist
- CONNECTIVITY_NOT_DEFINED
  Remote port is not connected through this local port
- COMPONENT_IS_NOT_AN_FC_PORT
  Component must specify an FC port
- COMMAND_NOT_ALLOWED_ON_MANAGEMENT_INTERFACE
  Operation is not allowed on Management IP Interface
- IPINTERFACE_DOES_NOT_EXIST
  IP Interface name does not exist
- TARGET_PORT_BAD_ADDRESS
  Remote port address is illegal or does not belong to the remote target
- BAD_LOCAL_IP_PORT
  An ID of a local IP port must be specified

Deactivating Connectivity to a Remote Target

Deactivates connectivity between a port on the local storage system and a port on a remote target.

```
target_connectivity_deactivate target=TargetName
< ipaddress=IPaddress local_ipinterface=IPInterface > |
< fcaddress=wwpn  local_port=PortId >
```

Parameters:

Name	Type	Description	Mandatory	Default
target	Object name	Remote target of the connectivity definition.	Y	
ipaddress		IP address of the port on the remote target (iSCSI targets only).	N	

Name	Type	Description	Mandatory	Default
local_ipinterface	Object name	Local IP interface that is connected to the remote port (iSCSI only).	N	
fcaddress		FC address of the port on the remote target (FC targets only).	N	
local_port		Port identifier.	N	

This command deactivates connectivity.

Each connectivity definition can be either active or inactive. The system does not use an inactive connectivity definition. Target Connectivity is active by default. Connectivity can be reactivated using Activating Connectivity to a Remote Target.

This command has no effect if the connectivity is already deactivated.

Example:

```
xcli -c Nextra1 target_connectivity_deactivate
target=Nextra2 local_module=101
```

Output:

```
Command completed successfully
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- TARGET_BAD_NAME
  Target name does not exist
- COMMAND_NOT_ALLOWED_ON_MANAGEMENT_INTERFACE
  Operation is not allowed on Management IP Interface
- CONNECTIVITY_NOT_DEFINED
  Remote port is not connected through this local port
- COMPONENT_IS_NOT_AN_FC_PORT
  Component must specify an FC port
- TARGET_PORT_BAD_ADDRESS
  Remote port address is illegal or does not belong to the remote target

- BAD_LOCAL_IP_PORT
  An ID of a local IP port must be specified
- IPINTERFACE_DOES_NOT_EXIST
  IP Interface name does not exist

Chapter 11. Remote Mirroring

The following sections describe the XIV Command Line Interface (XCLI) for remote mirroring.

The sections are listed as follows:

- mirror_create(Creates remote mirroring coupling.)
- mirror_list(Lists the status and configuration of mirroring couplings)
- mirror_delete(Deletes a remote mirroring coupling definition.)
- mirror_activate(Activates mirroring for a defined mirror coupling.)
- mirror_deactivate(Deactivates mirroring for a defined mirror coupling.)
- mirror_switch_roles(Switches roles between master and slave volumes.)
- mirror_change_role(Changes the role of a local mirroring peer between Master and Slave)

The following commands are no longer in use:

- mirror_change_config (Command is no longer in use in this version. Parameter create_last_consistent_snapshot is associated directly with pools)
- sync_rates_set (Command is no longer in use in this version. Supplanted by command target_config_sync_rates.)

Creating a Mirroring Definition

Creates remote mirroring coupling.

```
mirror_create vol=VolName slave_vol=SlaveVolumeName [ create_slave=<Yes|No>
  [ remote_pool=RemotePoolName ] ] target=TargetName
```

Parameters:

Name	Type	Description	Mandatory	Default
vol	Object name	Local volume to be mirrored (the master).	Y	
target	Object name	Remote target to contain the slave volume.	Y	
slave_vol	Object name	Name of the slave volume on the remote storage system.	Y	
remote_pool	Object name	The Storage Pool on the remote system. Relevant only if creating a slave.	N	

Name	Type	Description	Mandatory	Default
create_slave	Boolean	Determines whether to create a new Slave volume or to use an existing one. Cannot be specified if validate_slave_data is specified. If both parameters are specified - the command fails.	N	no

Mirroring is the process of ensuring that both volumes contain identical data at all times. This command defines a new mirroring coupling between a master volume and a slave volume.

Creating a mirroring coupling, an existing master volume is specified together with a slave volume. This slave volume either already exists or it is created by this command. If the slave volume already exists, the command receives its name along with the remote system name. If it is created by this command, the input parameters specify the remote storage system name, the name of the slave volume that is created and the Storage Pool that will contain the newly created slave volume.

Mirroring is created in the Inactive state. The mirroring coupling must then be activated in order to start the initialization process, which copies the data from the master volume to the slave volume.

A storage system can have multiple mirroring definitions between pairs of volumes on various remote systems.

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- VOLUME_BAD_NAME
  Volume name does not exist
- VOLUME_IS_MASTER
  Local volume is already defined as a master volume
- REMOTE_VOLUME_IS_MASTER
  Slave volume is currently defined as a master volume
- REMOTE_VOLUME_IS_SLAVE
  Slave volume is already defined as a slave volume

- MAX_MIRRORS_REACHED
  Maximum number of mirrors already defined
- REMOTE_MAX_VOLUMES_REACHED
  Maximum number of volumes already defined on remote machine
- TARGET_BAD_TYPE
  Target machine is not XIV machine
- VOLUME_BAD_PREFIX
  Volume name has a reserved prefix
- NOT_ENOUGH_SPACE_ON_REMOTE_MACHINE
  Not enough free space to set requested size of slave volume
- BAD_REMOTE_VOLUME_SIZE
  Master and slave volumes contain a different number of blocks
- REMOTE_POOL_DOES_NOT_EXIST
  Pool does not exist on remote machine
- REMOTE_POOL_NOT_SPECIFIED
  A Pool on remote machine must be specified when a slave volume is to be created
- REMOTE_TARGET_NOT_CONNECTED
  There is currently no connection from the target system
- REMOTE_VOLUME_EXISTS
  Slave volume name already exists and cannot be created
- VOLUME_IS_SNAPSHOT
  Operation is not permitted on snapshots
  Operation is not permitted on snapshots
- REMOTE_VOLUME_IS_SNAPSHOT
  Slave volume is a snapshot
- TARGET_BAD_NAME
  Target name does not exist
- TARGET_NO_ACCESS
  No access permissions to slave machine
- TARGET_NOT_CONNECTED
  There is currently no connection to the target system
- REMOTE_MAX_MIRRORS_REACHED
  Maximum number of mirrors already defined on remote machine
- REMOTE_VOLUME_HAS_DATA_MIGRATION
  Data Migration is defined for slave volume
- VOLUME_DATA_MIGRATION_UNSYNCHRONIZED
  Data Migration has not completed to this volume
- REMOTE_VOLUME_LOCKED
  Slave volume is locked
- TIMEOUT
  Remote operation did not complete in time
- VOLUME_HAS_MIRRORING_SNAPSHOTS
  Volume has snapshots created by previous mirroring process.
- SLAVE_VOLUME_NOT_FORMATTED

Slave volume is not formatted
- TARGET_DOES_NOT_ACCEPT_XIV_COMMANDS
  Target system does not accept XIV management commands
- LOCAL_VOLUME_IS_SLAVE
  Local volume is already defined as a slave volume
- BAD_REMOTE_VOLUME_NAME
  Slave volume name does not exist

Viewing Mirroring Status

Lists the status and configuration of mirroring couplings

```
mirror_list       [ vol=VolName ]
```

Parameters:

Name	Type	Description	Mandatory	Default
vol	Object name	Local volume name.	N	[none]

Id	Name	Description	Default Position
local_peer_name	Name		1
mirror_object	Mirror Object		2
designation	Designation		
current_role	Role		3
target_name	Remote System		4
remote_peer_name	Remote Peer		5
active	Active		6
sync_state	Status		7
connected	Target Connected		8
size_to_synchronize	Size To Sync (MB)		
sync_progress	Sync Progress (%)		
mirror_error	Mirror Error	No Error, Secondary pool exhausted, Configuration error or No thin provisioning resources	
schedule_name	Schedule Name		

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Deleting a Remote Mirroring Definition

Deletes a remote mirroring coupling definition.

```
mirror_delete vol=VolName
```

Parameters:

Name	Type	Description	Mandatory	Default
vol	Object name	Local Master volume name.	N	

This command deletes the definition of a remote mirroring coupling.

When a coupling is initially created or after it is deactivated, it is in *Standby* mode. Only a Standby coupling can be deleted.

After the remote mirroring is deleted, both peers are configured as *None*, meaning that they are no longer configured as either a master or a slave.

Note:

Only the remote mirroring coupling definition is deleted. The volumes themselves are not deleted.

The local object specified in the *vol*parameter, must be a master.

The remote mirroring coupling is deleted on both sides when communication is functioning. If there is no communication, mirroring is only deleted on the master, and a configuration error appears on the slave once communication resumes.

The command can only be issued on the Master, and when the link is *OK*.

Command outcome:

- Event is generated
- Overall coupling statistics are captured
- Outstanding pertinent Sync Jobs are deleted
- Completion of process is recorded in log

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- VOLUME_BAD_NAME
  Volume name does not exist
- VOLUME_NO_MIRROR
  Local volume does not have remote mirroring definitions
- **Target event not found. href = VOLUME_IS_NOT_MASTER**
- MIRROR_IS_ACTIVE
  Remote mirroring is currently active
- FORCE_DELETE_NOT_ALLOWED_ON_MASTER
  Only slave mirrors need to be forced to be deleted
- LOCAL_PEER_IS_NOT_MASTER
  Local peer is not the master

Activating Mirroring

Activates mirroring for a defined mirror coupling.

```
mirror_activate vol=VolName
```

Parameters:

Name	Type	Description	Mandatory	Default
vol	Object name	Master volume.	Y	

This command activates the coupling and switches it to Active state. In the Active state, the slave volume is updated together with the master volume, in contrast to the Inactive state, where only the master is updated. If mirroring is already active, this command has no effect and a success code is returned.

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- VOLUME_BAD_NAME
  Volume name does not exist
- VOLUME_NO_MIRROR
  Local volume does not have remote mirroring definitions
- LOCAL_PEER_IS_NOT_MASTER
  Local peer is not the master
- MIRROR_CONFIGURATION_ERROR
  Mirror local configuration does not match remote configuration

- REMOTE_MAX_VOLUMES_REACHED
  Maximum number of volumes already defined on remote machine
- SYNC_ALREADY_ACTIVE
  Synchronization is already active

Deactivating Mirroring

Deactivates mirroring for a defined mirror coupling.

```
mirror_deactivate  vol=<VolName [ ,VolName... ] >
```

Parameters:

Name	Type	Description	Mandatory	Default
vol	Object name	Master volume name or a list of master volumes.	Y	

This command deactivates a coupling and switches it to an Inactive state. While in the Inactive state, only the master volume is updated. This is in contrast to the Active state, where the slave volume is updated together with the master volume.

If the mirroring is already inactive, this command has no effect and a success code is returned.

If more than one volume is specified, mirroring on all the volumes is deactivated. Furthermore, the deactivation of all the volumes is performed as an atomic operation, so that the slave volumes remain consistent with each other.

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- VOLUME_BAD_NAME
  Volume name does not exist
- VOLUME_NO_MIRROR
  Local volume does not have remote mirroring definitions
- SYNC_ALREADY_INACTIVE
  Synchronization is already inactive
- LOCAL_PEER_IS_NOT_MASTER
  Local peer is not the master

Switching Roles between Master and Slave

Switches roles between master and slave volumes.

```
mirror_switch_roles vol=VolName
```

Parameters:

Name	Type	Description	Mandatory	Default
vol	Object name	Local volume name.	Y	

This command switches between the roles of the master and the slave volumes. It can only be issued when the coupling is operational and synchronized, and only on the master.

After the command is executed, the volume that was previously the master volume becomes the slave volume and the volume that was previously the slave volume becomes the master volume.

Before this command switches roles, the system stops accepting new writes to the local volume and performs all pending writes. Only after all pending writes have been committed, the roles are switched.

After the command is executed, the coupling is deactivated and the user has to activate it in order to restart synchronization. It is advised to create a snapshot before deactivating the coupling in order to enable recovery from logical errors due to incorrect server configurations.

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- VOLUME_BAD_NAME
  Volume name does not exist
- VOLUME_NO_MIRROR
  Local volume does not have remote mirroring definitions
- MIRROR_IS_NOT_SYNCHRONIZED
  Mirror is not synchronized
- VOLUME_HAS_DATA_MIGRATION
  Data Migration is defined for this volume
- REMOTE_TARGET_NOT_CONNECTED
  There is currently no connection from the target system
- LOCAL_PEER_IS_NOT_MASTER

Local peer is not the master

Changing the Roles of a Mirrored Volume

Changes the role of a local mirroring peer between Master and Slave

```
mirror_change_role vol=VolName [ new_role=<Master|Slave> ]
```

Parameters:

Name	Type	Description	Mandatory	Default
vol	Object name	Local volume name. Must be specified if the command concerns a volume.	Y	
new_role	Enumeration	Role name of the peer If not specified, the command will act as a toggle - changing the role of the peer between Master and Slave.	N	none

This command changes the role of the local volume from master to slave or from slave to master when the coupling is non-operational. It is assumed that the command will be issued on both volumes of the coupling before the coupling becomes operational again, so that upon reconnection there will still be one master and one slave volume.

When applied to a master volume, the mirroring has to be non-operational. All changes made to the master since the last time the volumes were synchronized will be reverted to their original value. This is done in order to allow the current slave volume to become the master.

It is impossible to apply the command while the coupling is in the Initialization state.

When applied to the slave volume, the command can even be issued when the mirroring is active, because there is no way to deactivate the mirroring on the slave. The slave volume will now become the master volume.

If the slave volume has a last_consistent snapshot, the mirroring was broken in the middle of the synchronization process and the slave volume could be inconsistent. In this case, the administrator must choose whether to use the most updated volume, which might be inconsistent or the last_consistent snapshot.

Reverting the volume to the last_consistent snapshot can only be performed by deleting the mirroring, reverting the volume and creating a new mirroring

definition. Either way, if a last_consistent snapshot exists, a most-updated snapshot is be created, keeping a copy of the information at the time of the role change.

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- VOLUME_BAD_NAME
  Volume name does not exist
- VOLUME_NO_MIRROR
  Local volume does not have remote mirroring definitions
- MIRROR_IS_INITIAL
  Mirror has not completed initial synchronization
  Operation is not permitted during initialization
- MIRROR_IS_ACTIVE
  Remote mirroring is currently active
- VOLUME_HAS_DATA_MIGRATION
  Data Migration is defined for this volume

Chapter 12. Data Migration

The following sections describe the XIV Command Line Interface (XCLI) for data migration.

The sections are listed as follows:

- dm_define(Defines a Data Migration configuration.)
- dm_test(Tests the Data Migration configuration.)
- dm_activate(Activates the Data Migration process.)
- dm_list(Lists Data Migration configuration and status.)
- dm_deactivate(Deactivates the Data Migration process.)
- dm_delete(Deletes the Data Migration process.)

Defining Data Migration Configuration

Defines a Data Migration configuration.

```
dm_define vol=VolName target=TargetName lun=SourceLUN
 source_updating=<yes|no> [ create_vol=<yes|no> ] [ pool=PoolName ]
```

Parameters:

Name	Type	Description	Mandatory	Default
vol	Object name	Data Migration destination volume on the local system.	Y	
target	Object name	Remote system containing the source volume.	Y	
lun	Integer	LUN of the source volume.	Y	
source_updating	Boolean	Specifies whether to use source volume updating.	Y	
create_vol	Boolean	A Boolean that determines whether to create a new volume or to use an existing one.	N	No

Name	Type	Description	Mandatory	Default
pool	Object name	Name of the Storage Pool to contain the volume. Used only when creating a volume. Mandatory when creating a volume.	N	

This command defines a Data Migration relationship between a local volume and a remote volume. This definition defines that the local volume should reflect the remote volume.

After this configuration has been defined, it can be tested using the Testing the Data Migration Definition command and then activated using the Activating Data Migration command. After this activation, hosts can read and write to this volume, and these operations are reflected on the remote volume.

The remote volume may be inaccessible at the time that the command is executed. In this case, the definition is only used when Data Migration is tested.

The local system acts as a host to the remote system. The remote system should be configured to make the remote volume accessible to the local system through the specified LUN.

If *source updating* is specified, each write to the local volume is reflected as a write to the remote volume. Otherwise, writes on the local volume are not reflected and the remote volume is not changed.

The local volume must be formatted.

If *create_vol* is specified as `yes`, the volume is created. In this case the size of the newly created volume is identical to the size of the source volume. When creating a volume, a pool name must be specified. Creating a volume fails if there is no connectivity to the target since the volume's size is unknown.

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- VOLUME_BAD_NAME
  Volume name does not exist
- TARGET_BAD_NAME
  Target name does not exist

- VOLUME_IS_SNAPSHOT
  Operation is not permitted on snapshots
  Operation is not permitted on snapshots
- VOLUME_HAS_MIRROR
  Mirror is defined for this volume
- VOLUME_BELONGS_TO_CG
  Volume belongs to a Consistency Group
- VOLUME_HAS_DATA_MIGRATION
  Data Migration is defined for this volume
- VOLUME_HAS_SNAPSHOTS
  Volume has snapshots
- VOLUME_NOT_FORMATTED
  Local volume is not formatted
- TOO_MANY_MIRRORS
  Maximum number of remote volumes (mirror/migration) is already defined
  Troubleshooting: Delete remote mirrors or Data Migration objects
- VOLUME_EXISTS
  Volume name already exists
- POOL_DOES_NOT_EXIST
  Storage Pool does not exist
- VOLUME_BAD_PREFIX
  Volume name has a reserved prefix
- NOT_ENOUGH_SPACE
  No space to allocate volume
- NOT_ENOUGH_HARD_SPACE
  No space to allocate for volume's current usage
- MAX_VOLUMES_REACHED
  Maximum number of volumes already defined
  Maximum number of volumes already defined
- ILLEGAL_VOLUME_SIZE
  Illegal volume size
- REMOTE_VOLUME_NO_LUN
  Remote volume's LUN is unavailable
- TARGET_NOT_CONNECTED
  There is currently no connection to the target system
- VOLUME_CANNOT_HAVE_ZERO_SIZE
  Volume size cannot be zero
- ILLEGAL_LUN
  LUN is out of range
- TARGET_IS_MIRRORING
  Target machine is defined only for remote mirroring
- NO_ONLINE_MIGRATION_WITHOUT_SOURCE_UPDATING
  Data Migration without automatic migration must be defined as source-updating

Testing the Data Migration Definition

Tests the Data Migration configuration.

```
dm_test vol=VolName
```

Parameters:

Name	Type	Description	Mandatory	Default
vol	Object name	Destination volume for Data Migration testing.	Y	

This command tests the Data Migration configuration. Completion codes indicate the types of test failures that may occur. Once a test is successful, then Data Migration can be activated.

If source updating is not defined for this Data Migration, writing is not tested.

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- VOLUME_BAD_NAME
  Volume name does not exist
- VOLUME_NO_DM
  Local volume does not have Data Migration definitions
- TARGET_NOT_CONNECTED
  There is currently no connection to the target system
- REMOTE_VOLUME_NO_LUN
  Remote volume's LUN is unavailable
- REMOTE_VOLUME_NO_READ_ACCESS
  Remote volume cannot be read
- REMOTE_VOLUME_NO_WRITE_ACCESS
  Remote volume is write protected
- BAD_REMOTE_VOLUME_SIZE
  Master and slave volumes contain a different number of blocks

Activating Data Migration

Activates the Data Migration process.

```
dm_activate vol=VolName
```

Parameters:

Name	Type	Description	Mandatory	Default
vol	Object name	Destination volume for Data Migration activation.	Y	

This command activates the Data Migration process. This is either an initial activation or an activation after de-activation.

Upon activation the Data Migration is tested in the same way as when using Testing the Data Migration Definition and this command fails if the Data Migration test fails.

This command has no effect if the process is already active.

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- VOLUME_BAD_NAME
  Volume name does not exist
- VOLUME_NO_DM
  Local volume does not have Data Migration definitions
- TARGET_NOT_CONNECTED
  There is currently no connection to the target system
- REMOTE_VOLUME_NO_LUN
  Remote volume's LUN is unavailable
- REMOTE_VOLUME_NO_READ_ACCESS
  Remote volume cannot be read
- REMOTE_VOLUME_NO_WRITE_ACCESS
  Remote volume is write protected
- BAD_REMOTE_VOLUME_SIZE
  Master and slave volumes contain a different number of blocks

Listing Data Migration Statuses

Lists Data Migration configuration and status.

```
dm_list [ vol=VolName ]
```

Parameters:

Name	Type	Description	Mandatory	Default
vol	Object name	Name of the volume to be listed.	N	All Data Migration volumes.

This command lists all the Data Migration configuration and statuses, including the following:

- Volume name
- Target name
- LUN
- Volume size (GB)
- Migration completed (GB)
- Migration activation (active/inactive)
- Migration status (synchronized, unsynchronized)
- Migration remaining (GB)
- Migration remaining (%)
- Estimated time to completion

Id	Name	Description	Default Position
local_volume_name	Local Volume		1
target_name	Remote System		2
remote_volume_lun	Remote LUN		3
active	Active		4
sync_state	Status		5
connected	Target Connected		6
size_to_synchronize	Size To Sync (MB)		
sync_progress	Sync Progress (%)		
start_migration_automatically	Start Data Migration Automatically		

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Allowed	
Read-only users	Allowed	
Technicians	Disallowed	

Deactivating Data Migration

Deactivates the Data Migration process.

```
dm_deactivate vol=VolName
```

Parameters:

Name	Type	Description	Mandatory	Default
vol	Object name	Local volume for Data Migration deactivation.	Y	

This command deactivates the process of Data Migration. Hosts are not served while the Data Migration process is in-active.

This command has no effect if the Data Migration is already inactive.

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Warnings:

- ARE_YOU_SURE_YOU_WANT_TO_DEACTIVATE_DATA_MIGRATION
  Deactivation will stop all applications, data migration can be deleted if it is done

Completion Codes:

- VOLUME_BAD_NAME
  Volume name does not exist
- VOLUME_NO_DM
  Local volume does not have Data Migration definitions

Deleting the Data Migration Process

Deletes the Data Migration process.

```
dm_delete vol=VolName
```

Parameters:

Name	Type	Description	Mandatory	Default
vol	Object name	Volume name for deleting the Data Migration process.	Y	

This command deletes the Data Migration configuration and stops the Data Migration process.

This command can only be executed if the Data Migration has reached the state of synchronization.

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- VOLUME_BAD_NAME
  Volume name does not exist
- VOLUME_NO_DM
  Local volume does not have Data Migration definitions
- DM_IS_NOT_SYNCHRONIZED
  Data Migration process has not been completed

Chapter 13. Event Handling

The following sections describe the XIV Command Line Interface (XCLI) for event handling, including listing events, filtering and sending notifications.

The sections are listed as follows:

- event_list(Lists system events.)
- event_list_uncleared(Lists uncleared alerting events.)
- event_clear(Clears alerting events.)
- smtpgw_define(Defines an SMTP gateway.)
- smtpgw_prioritize(Sets the priority of which SMTP gateway should be used to send emails.)
- smtpgw_delete(Deletes a specified SMTP gateway.)
- smtpgw_update(Updates the configuration of an SMTP gateway.)
- smtpgw_rename(Renames an SMTP gateway.)
- smtpgw_list(Lists SMTP gateways.)
- smsgw_define(Defines an SMS gateway.)
- smsgw_update(Updates an SMS gateway.)
- smsgw_prioritize(Sets the priorities of the SMS gateways for sending SMS messages.)
- smsgw_delete(Deletes an SMS gateway.)
- smsgw_rename(Renames an SMS gateway.)
- smsgw_list(Lists SMS gateways.)
- dest_define(Defines a new destination for event notifications.)
- dest_update(Updates a destination.)
- dest_rename(Renames an event notification destination.)
- dest_delete(Deletes an event notification destination.)
- dest_list(Lists event notification destinations.)
- dest_test(Sends a test message to an event notification destination.)
- destgroup_create(Creates an event notification destinations group.)
- destgroup_add_dest(Adding an event notification destination to a destination group.)
- destgroup_remove_dest(Removes an event notification destination from a destination group.)
- destgroup_rename(Renames an event notification destination group.)
- destgroup_delete(Deletes an event notification destination group.)
- destgroup_list(Lists destination groups.)
- rule_create(Creates an event notification rule.)
- rule_update(Updates an event notification rule.)
- rule_rename(Renames an event notification rule.)
- rule_delete(Deletes an event notification rule.)
- rule_activate(Activates an event notification rule.)
- rule_deactivate(Deactivates an event notification rule.)
- rule_list(Lists event notification rules.)

- event_redefine_threshold(Redefines the threshold of a parameterized event.)
- custom_event(Generates a custom event.)

Listing Events

Lists system events.

```
event_list [ max_events=MaxEventsToList ]
     [ after=<afterTimeStamp|ALL> ]
     [ before=<beforeTimeStamp|ALL> ]
     [ min_severity=<INFORMATIONAL|WARNING|MINOR|MAJOR|CRITICAL> ]
     [ alerting=<yes|no|all> ]
     [ cleared=<yes|no> ] [ code=EventCode ]
     [ object_type=<cons_group|destgroup|dest|dm|host|map|
       mirror|pool|rule|smsgw|smtpgw|target|volume|
       cluster|ip_interface|ldap_conf|meta_data_object|
       sync_schedule|user|user_group> ]
     [ beg=BeginIndex ] [ end=EndIndex ]
     [ internal=<yes|no|all> ]
```

Parameters:

Name	Type	Description	Mandatory	Default
max_events	Positive integer	Maximum number of events to list.	N	100.
after		Earliest time/date.	N	no filter.
before		Latest time/date.	N	no filter.
min_severity	Enumeration	Minimum severity.	N	no filter.
alerting	Boolean	Filter alerting events.	N	no filter.
cleared	Boolean	Filter cleared events.	N	no filter.
code		Filter by a specific event code.	N	no filter.
object_type	Enumeration	Filter events by the type of the related system object.	N	no filter
internal	Boolean	Filter XIV internal events.	N	no
beg	Integer	Index of the first event. If negative, then counts from the end.	N	Use end definition.
end	Integer	End index. If negative, then counts from the end.	N	last event

This command lists system events according to specified criteria, such as minimum severity, event type and so on. The event list displays the following information for each event: timestamp, severity, code, user and description.

Events are listed and sorted by time of creation, where the latest events are listed last. Events are listed by default in their user-readable textual form. Alternatively, the XCLI option for comma separated values can be used to generate output that can serve as input for other applications.

The syntax for the `before` and `after` fields is as follows: `Y-M-D[.[h[:m[:s]]]]`, where the ranges are as follows:

- Y - year (four digit)
- M - month (1-12)
- D - day (1-31)
- h - hour (0-23, with 0 as default)
- m - minute (0-59, with 0 as default)
- s - second (0-59, with 0 as default)

Note:

The year, month and day are separated by dashes, while the optional hour, minute and second are separated by colons.

Id	Name	Description	Default Position
timestamp	Timestamp		1
severity	Severity		2
code	Code		3
user_name	User		4
description	Description		5
index	Index		
alerting	Alerting		
cleared	Cleared		
tshooting	Trouble Shooting		

Example:

```
xcli  -u -c Nextra1 event_list max_events=10
```

Output:

```
Timestamp             Severity        Code
2009-05-12 15:10:16   Informational   START_WORK
2009-05-12 15:16:11   Informational   POOL_CREATE
2009-05-12 15:16:22   Critical        WOULD_BE_EMERGENCY_SHUTDOWN
2009-05-12 15:16:23   Informational   VOLUME_CREATE

Additional output fields
(lines are broken to fit the page width of this Guide):

User              Description
                  System has entered ON state.
xiv_development   Storage Pool of size 171GB was created with name
                  'p1_m'.
                  An emergency shutdown has been detected, but UPS control
                  is disabled.
xiv_development   Volume was created with name 'master' and size 17GB in
                  Storage Pool with name 'p1_m'.
```

Access Control:

User Category	Permission	Condition
Storage administrator	Conditionally Allowed	Only the technician can run this command with the internal parameter
Application administrator	Conditionally Allowed	Only the technician can run this command with the internal parameter
Read-only users	Conditionally Allowed	Only the technician can run this command with the internal parameter
Technicians	Allowed	

Completion Codes:

- UNRECOGNIZED_EVENT_CODE
  'String' is not a recognized event code
  Troubleshooting: Consult the manual for the list of event codes
- CANNOT_READ_EVENTS
  Cannot read events.
  Troubleshooting: Contact support

Listing Uncleared Alerting Events

Lists uncleared alerting events.

```
event_list_uncleared
```

Parameters:

Name	Type	Description	Mandatory	Default

This command lists uncleared alerting events.

Example:

```
xcli  -u  -c Nextra1 event_list_uncleared
```

Output:

```
Index   Code            Severity
------- --------------- ---------------
318     VOLUME_CREATE   Informational
666     VOLUME_DELETE   Informational
```

Id	Name	Description	Default Position
index	Index		1
code	Code		2
severity	Severity		3

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Allowed	
Read-only users	Allowed	
Technicians	Allowed	

Clearing Alerting Events

Clears alerting events.

```
event_clear event_id=EventId [ all_preceding=<yes|no> ] [ internal=<yes|no|all> ]
```

Parameters:

Name	Type	Description	Mandatory	Default
event_id	Positive integer	ID number of the event to clear.	Y	
all_preceding	Boolean	Clears all events preceding the specified event.	N	no.
internal	Boolean	Clears XIV-internal events.	N	no

This command clears alerting events.

In order to ensure that an event was indeed received, an event notification may be sent repeatedly until it is cleared by a CLI command or the GUI. Such events are called *alerting* events. An event is defined as *alerting* if at the time of the event's generation it was matched by an *alerting* rule, meaning a rule that either has snooze or escalation definitions.

Notifications for the alerting event are sent until it is cleared by this command. The clearing operation does not imply that the problem has been solved. It only implies that the event has been noted by the relevant person who takes responsibility for fixing the problem.

A user may either clear a specific event or clear all alerting events.

Example:

```
xcli  -u -c Nextra1 event_clear event_id=87
```

Output:

```
Command executed successfully.
```

Access Control:

User Category	Permission	Condition
Storage administrator	Conditionally Allowed	Only the technician can run this command with the internal parameter
Application administrator	Conditionally Allowed	Only the technician can run this command with the internal parameter
Read-only users	Disallowed	
Technicians	Allowed	

Completion Codes:

- ONLY_TECHNICIAN_CAN_REFER_TO_INTERNAL_EVENT_OBJECTS
  Only technician can refer to internal event objects

Defining a New SMTP Gateway

Defines an SMTP gateway.

```
smtpgw_define smtpgw=SMTPGatewayName address=Address
              [ from_address=<SenderEmailAddress|DEFAULT> ]
              [ reply_to_address=<ReplyToAddress|DEFAULT> ]
```

Parameters:

Name	Type	Description	Mandatory	Default
smtpgw	Object name	SMTP gateway name.	Y	

Name	Type	Description	Mandatory	Default
address		SMTP gateway address (IP or DNS name).	Y	
from_address		Sender's email address used for out-going emails sent through this SMTP server.	N	DEFAULT (system-wide sender's address that applies to all servers).
reply_to_address		The reply to address used for outgoing emails sent through this SMTP server.	N	DEFAULT (system-wide reply-to address that applies to all servers).

This command defines an SMTP gateway.

Several email gateways can be defined to enable notification of events by email or to enable the sending of SMS messages via Email-to-SMS gateways. By default, XIV attempts to send each email notification through the first gateway according to the order that you specify. Subsequent gateways are only tried if the first in line returns an error. A specific email destination, or a specific SMS gateway may be defined to use only specific SMTP gateways.

The SMTP protocol dictates that every email message must specify the email address of the sender. This sender address must be a valid address for two reasons:

- Many SMTP gateways require a valid sender address, otherwise they will not forward the email, as a security measure in order to prevent unauthorized usage of the SMTP server. Often this sender address must be limited to a specific domain.
- The sender's address is used as the destination for error messages generated by the SMTP gateways, such as: incorrect email address, full email mailbox and so on.

If the sender's address is not specified for a specific SMTP gateway, a global system-wide sender's address specified in Setting Configuration Parameters is used.

The user can also configure a reply-to address which is different from the sender's address, if it is required that the return emails are sent to another destination.

Example:

```
xcli -c Nextra1 smtpgw_define smtpgw=mailserver1 address=smtp.yourcompany.com
from_address=nextra@yourcompany.com
reply_to_address=nextraerrors@yourcompany.com
```

Output:

```
Command executed successfully.
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- CANNOT_CHANGE_EVENT_CONF_WITH_ALERTING_EVENTS
  Cannot change event configuration while there are alerting events
  Troubleshooting: Clear all alerting events before changing event configuration
- FROM_ADDRESS_NOT_DEFINED
  Neither the gateway's From Address nor the default From Address is defined
- GATEWAY_MAX_REACHED
  Maximum number of gateways already defined
- GATEWAY_NAME_ALREADY_EXISTS
  Gateway name already exists

Prioritizing SMTP Gateways

Sets the priority of which SMTP gateway should be used to send emails.

```
smtpgw_prioritize order=<gw1[,gw2]...>
```

Parameters:

Name	Type	Description	Mandatory	Default
order	Object name	List of all the SMTP gateways in order of their priority.	Y	

This command sets the priority in which SMTP gateway should be used to send emails. Several email gateways can be defined to enable notification of events or the sending of SMS by email. By default, XIV attempts to send each email through the first gateway according to the order that is specified in this command. Only one gateway is used and subsequent gateways are only tried if the preceding ones in this priority list return an error.

These priorities are used only for email destinations and SMS gateways that did not specify their own SMTP gateways.

Example:

```
xcli -u  -c Nextra1 smtpgw_prioritize order=mailserver2,mailserver1
```

Output:

```
Command completed successfully
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- CANNOT_CHANGE_EVENT_CONF_WITH_ALERTING_EVENTS
  Cannot change event configuration while there are alerting events
  Troubleshooting: Clear all alerting events before changing event configuration
- GATEWAY_NAME_APPEARS_TWICE
  Gateway name appears twice in the list
- GATEWAY_NAME_DOES_NOT_EXIST
  Gateway name does not exist
- GATEWAY_NAME_MISSING_FROM_LIST
  Gateway name is missing from the list

Deleting an SMTP Gateway

Deletes a specified SMTP gateway.

```
smtpgw_delete smtpgw=SMTPGatewayName
```

Parameters:

Name	Type	Description	Mandatory	Default
smtpgw	Object name	SMTP gateway to be deleted.	Y	

This command deletes a specified SMTP gateway. A gateway cannot be deleted if it is part of a notification rule, is being used as an SMS gateway or if it belongs to a destination.

An SMTP gateway cannot be deleted while there are uncleared alerting events.

Example:

```
xcli -c Nextra1 smtpgw_delete smtpgw=mailserverbackup
```

Output:

```
Command completed successfully
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Warnings:

- ARE_YOU_SURE_YOU_WANT_TO_DELETE_SMTP_GATEWAY
  Are you sure you want to delete SMTP gateway *Gateway*?

Completion Codes:

- CANNOT_CHANGE_EVENT_CONF_WITH_ALERTING_EVENTS
  Cannot change event configuration while there are alerting events
  Troubleshooting: Clear all alerting events before changing event configuration
- GATEWAY_NAME_DOES_NOT_EXIST
  Gateway name does not exist
- GATEWAY_USED_BY_DESTINATION
  Gateway is used by a destination
- GATEWAY_USED_BY_SMS_GATEWAY
  Gateway is used by an SMS Gateway

Updating an SMTP Gateway

Updates the configuration of an SMTP gateway.

```
smtpgw_update smtpgw=SMTPGatewayName [ address=Address ]
              [ from_address=<SenderEmailAddress|DEFAULT> ]
              [ reply_to_address=<ReplyToAddress|DEFAULT> ] [ internal=<yes|no> ]
```

Parameters:

Name	Type	Description	Mandatory	Default
smtpgw	Object name	SMTP gateway name.	Y	
address		SMTP gateway address (IP or DNS name).	N	Leave unchanged.
internal	Boolean	Should be specified as YES for XIV internal gateway.	N	NO
from_address		Sender's Email address used for out-going emails sent through this SMTP server, or DEFAULT for the system-wide default.	N	Leave unchanged.

Name	Type	Description	Mandatory	Default
reply_to_address		The reply-to address used for outgoing emails sent through this SMTP server, or DEFAULT for the system-wide default.	N	Leave unchanged.

This command updates the configuration of an existing SMTP gateway. Fields which are not specified are not changed.

Example:

```
xcli -c Nextra1 smtpgw_update smtpgw=mailserver1 address=smtp2.yourcompany.com
                              from_address=nextra@yurcompany.com
           reply_to_address=nextraerrors@yourcompany.com
```

Output:

```
Command executed successfully.
```

Access Control:

User Category	Permission	Condition
Storage administrator	Conditionally Allowed	Only the technician can run this command with the internal parameter
Application administrator	Conditionally Allowed	Only the technician can run this command with the internal parameter
Read-only users	Disallowed	
Technicians	Allowed	

Completion Codes:

- GATEWAY_NAME_DOES_NOT_EXIST
  Gateway name does not exist
- CANNOT_CHANGE_EVENT_CONF_WITH_ALERTING_EVENTS
  Cannot change event configuration while there are alerting events
  Troubleshooting: Clear all alerting events before changing event configuration
- FROM_ADDRESS_NOT_DEFINED
  Neither the gateway's From Address nor the default From Address is defined
- ONLY_TECHNICIAN_CAN_REFER_TO_INTERNAL_EVENT_OBJECTS
  Only technician can refer to internal event objects

Renaming an SMTP Gateway

Renames an SMTP gateway.

```
smtpgw_rename smtpgw=SMTPGatewayName new_name=NewSMTPGWName
```

Parameters:

Name	Type	Description	Mandatory	Default
smtpgw	Object name	SMTP gateway to be renamed.	Y	
new_name	Object name	New name for the SMTP gateway.	Y	

This command renames an SMTP gateway.

Example:

```
xcli  -u  -c Nextra1 smtpgw_rename smtpgw=mailserver2 new_name=mailserverbackup
```

Output:

```
Command completed successfully
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- CANNOT_CHANGE_EVENT_CONF_WITH_ALERTING_EVENTS
  Cannot change event configuration while there are alerting events
  Troubleshooting: Clear all alerting events before changing event configuration
- GATEWAY_NAME_ALREADY_EXISTS
  Gateway name already exists
- GATEWAY_NAME_DOES_NOT_EXIST
  Gateway name does not exist

Listing SMTP Gateways

Lists SMTP gateways.

```
smtpgw_list [ smtpgw=SMTPGatewayName ]  [ internal=<yes|no> ]
```

Parameters:

Name	Type	Description	Mandatory	Default
smtpgw	Object name	Name of SMTP gateway to list.	N	no.
internal	Enumeration	Filters gateways by their XIV-internal attribute.	N	no

This command lists defined SMTP gateways and their configuration information.

Id	Name	Description	Default Position
name	Name		1
address	Address		2
priority	Priority		3
from_address	From Address		
reply_to_address	Reply-to Address		
failed	Failed		
port	Port		
creator	Creator		

Example:

```
xcli -u  -c Nextra1 smtpgw_list
```

Output:

```
Name         Email Address         Port  Priority
mailserver1  smtp.yourcompany.com  25    1
mailserver2  smtp.yourcompany.com  25    2
```

Access Control:

User Category	Permission	Condition
Storage administrator	Conditionally Allowed	Only the technician can run this command with the internal parameter
Application administrator	Conditionally Allowed	Only the technician can run this command with the internal parameter
Read-only users	Conditionally Allowed	Only the technician can run this command with the internal parameter
Technicians	Allowed	

Defining an SMS Gateway

Defines an SMS gateway.

```
smsgw_define smsgw=SMSGatewayName email_address=EmailAddressScheme
           subject_line=SubjectLineScheme email_body=EmailBodyScheme
     [ smtpgw=<SMTPGW1,...|ALL> ]
```

Parameters:

Name	Type	Description	Mandatory	Default
smsgw	Object name	SMS gateway name.	Y	
email_address	Token String	Format for the email address.	Y	
subject_line	Token String	Format for the subject line.	Y	
email_body	Token String	Format for the email body.	Y	
smtpgw	Object name	List of SMTP gateways to be used.	N	The SMTP gateways defined in the smtpgw_prioritize command.

SMS gateways are used to send event notifications via SMS messages. SMS messages are sent via SMS-to-Email servers, and when defining a new SMS gateway, we need to know how to SMS messages are encapsulated in the email message.

When the XIV system sends an SMS message it uses the actual message text that describes the event and the destination number. The destination number is comprised from an area code and the local number, both which are specified when a destination is defined as described in the Defining a New Event Notification Destination command.

The message's text and the destination numbers can be embedded into the email message in either the email destination address, email subject line or the email body. This command defines how email messages are formatted, and how they contain the information of the specific SMS.

When defining the SMS gateway, three fields must be specified in order to define the formatting:

- *email_address*: This is the email address used for sending the SMS via the Email-to-SMS gateway.
- *subject_line*: This is the subject line of the outgoing email that would be converted into an SMS.
- *email_body*: This is the body of the outgoing email that would be converted into an SMS.

When specifying each of these fields, the text can be either fixed, contain the event's text or contain the destination phone number. This is done by embedding the following escape sequences into the text:

- *{areacode}*. This escape sequence is replaced by the destination's cellular number area code.
- *{number}*. This escape sequence is replaced by the destination's cellular local number.
- *{message}*. This escape sequence is replaced by the text to be shown to the user.
- \{, \}, \\. These are replaced by the {, } or \ respectively.

By default, the email to the Email-to-SMS server is sent through the defined SMTP servers, prioritized by the Prioritizing SMTP Gateways command. If needed, the user may specify a specific SMTP gateway or gateways to be used for sending email to this Email-to-SMS gateway.

Several SMS gateways can be defined. The system will try the SMS gateways, in the order specified in Prioritizing SMS Gateways until it successfully connects to one of them. It is possible to define that specific SMS destinations will use specific SMS gateways (see Defining a New Event Notification Destination).

Example:

```
xcli -u  -c Nextra1 smsgw_define smsgw=SMSGW1
            email_address={areacode}{number}@sms2emailserver.yourcompany.com
    subject_line=SMS email_body={message}
```

Output:

```
Command executed successfully.
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- CANNOT_CHANGE_EVENT_CONF_WITH_ALERTING_EVENTS
  Cannot change event configuration while there are alerting events
  Troubleshooting: Clear all alerting events before changing event configuration
- GATEWAY_MAX_REACHED
  Maximum number of gateways already defined
- SMSGW_CANNOT_BE_DEFINED_WITHOUT_SMTPGW
  SMS gateways cannot be defined if no SMTP gateways are defined
- GATEWAY_NAME_DOES_NOT_EXIST
  Gateway name does not exist
- GATEWAY_NAME_APPEARS_TWICE
  Gateway name appears twice in the list

- GATEWAY_NAME_ALREADY_EXISTS
  Gateway name already exists

Updating an SMS Gateway

Updates an SMS gateway.

```
smsgw_update smsgw=SMSGatewayName [ email_address=EmailAddressScheme ]
     [ subject_line=SubjectLineScheme ] [ email_body=EmailBodyScheme ]
   [ smtpgw=<SMTPGW1[,SMTPGW2]...|ALL> ]
```

Parameters:

Name	Type	Description	Mandatory	Default
smsgw	Object name	SMS gateway name.	Y	
email_address	Token String	Format for email address.	N	Leave unchanged.
subject_line	Token String	Format for subject line.	N	Leave unchanged.
email_body	Token String	Format for the email's body.	N	Leave unchanged.
smtpgw	Object name	List of SMTP gateways to be used.	N	The SMTP gateways defined in the smtpgw_prioritize command.

This command updates the configuration information of an existing SMS gateway. For the exact description and documentation of each parameter, see the documentation of Defining an SMS Gateway.

This command cannot be executed while there are uncleared alerting events.

Parameters which are not specified will not be changed.

Example:

```
xcli -u  -c Nextra1 smsgw_update smsgw=SMSGW1
            email_address={areacode}{number}@sms2emailserver.yourcompany.com
    subject_line=NextraSMS
    email_body={message}
```

Output:

```
Command executed successfully.
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	

User Category	Permission	Condition
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- CANNOT_CHANGE_EVENT_CONF_WITH_ALERTING_EVENTS
  Cannot change event configuration while there are alerting events
  Troubleshooting: Clear all alerting events before changing event configuration
- GATEWAY_NAME_APPEARS_TWICE
  Gateway name appears twice in the list
- GATEWAY_NAME_DOES_NOT_EXIST
  Gateway name does not exist

Prioritizing SMS Gateways

Sets the priorities of the SMS gateways for sending SMS messages.

```
smsgw_prioritize order=<gw1[,gw2]...>
```

Parameters:

Name	Type	Description	Mandatory	Default
order	Object name	List of all SMS gateways ordered by priority.	Y	

This command sets the priority in which SMS gateways are used to send SMSs. SMS messages can be sent to cell phones through one of the email-to-SMS gateways in this list. This command determines the order in which XIV attempts to use these SMS gateways.

Only one gateway is used and subsequent gateways are only tried if the preceding ones in this priority list return an error.

Specific SMS destinations may define their own SMS gateways to be used when sending SMS to these destinations, regardless of this list.

Example:

```
xcli -u  -c Nextra1 smsgw_prioritize order=SMSGW1,SMSGW2
```

Output:

```
Command completed successfully
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- CANNOT_CHANGE_EVENT_CONF_WITH_ALERTING_EVENTS
  Cannot change event configuration while there are alerting events
  Troubleshooting: Clear all alerting events before changing event configuration
- GATEWAY_NAME_APPEARS_TWICE
  Gateway name appears twice in the list
- GATEWAY_NAME_DOES_NOT_EXIST
  Gateway name does not exist
- GATEWAY_NAME_MISSING_FROM_LIST
  Gateway name is missing from the list

Deleting an SMS Gateway

Deletes an SMS gateway.

```
smsgw_delete smsgw=SMSGatewayName
```

Parameters:

Name	Type	Description	Mandatory	Default
smsgw	Object name	SMS gateway to be deleted.	Y	

This command deletes an SMS gateway. A gateway cannot be deleted if it is part of a notification rule or if it is being used by a destination.

It is impossible to delete an SMS gateway while there are uncleared alerting events.

Example:

```
xcli -u  -c Nextra1 smsgw_delete smsgw=external-SMSGW
```

Output:

```
Command completed successfully
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Warnings:

- ARE_YOU_SURE_YOU_WANT_TO_DELETE_SMS_GATEWAY
  Are you sure you want to delete SMS gateway *Gateway*?

Completion Codes:

- CANNOT_CHANGE_EVENT_CONF_WITH_ALERTING_EVENTS
  Cannot change event configuration while there are alerting events
  Troubleshooting: Clear all alerting events before changing event configuration
- GATEWAY_NAME_DOES_NOT_EXIST
  Gateway name does not exist
- GATEWAY_USED_BY_DESTINATION
  Gateway is used by a destination

Renaming an SMS Gateway

Renames an SMS gateway.

```
smsgw_rename smsgw=SMSGatewayName new_name=NewSMSGWName
```

Parameters:

Name	Type	Description	Mandatory	Default
smsgw	Object name	SMS gateway to be renamed.	Y	
new_name	Object name	New name for the SMS gateway.	Y	

This command renames an SMS gateway.

SMS gateways cannot be renamed while there are uncleared alerting events.

Example:

```
xcli -u  -c Nextra1 smsgw_rename smsgw=SMSGW2 new_name=external-SMSGW
```

Output:

```
Command completed successfully
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- CANNOT_CHANGE_EVENT_CONF_WITH_ALERTING_EVENTS
  Cannot change event configuration while there are alerting events
  Troubleshooting: Clear all alerting events before changing event configuration
- GATEWAY_NAME_ALREADY_EXISTS
  Gateway name already exists
- GATEWAY_NAME_DOES_NOT_EXIST
  Gateway name does not exist

Listing SMS Gateways

Lists SMS gateways.

```
smsgw_list [ smsgw=SMSGatewayName ]
```

Parameters:

Name	Type	Description	Mandatory	Default
smsgw	Object name	Name of SMS gateway to list.	N	All gateways.

Lists all SMS gateways or a specific one. For each SMS gateway all its configuration information is listed.

Id	Name	Description	Default Position
name	Name		1
email_address	Email Address		2
gateways	SMTP Gateways		3
subject_line	Subject Line		
email_body	Email Body		
priority	Priority		

Example:

```
xcli -u  -c Nextra1 smsgw_list
```

Output:

```
Name    Email Address                                        SMTP Gateways
SMSGW1  {areacode}{number}@sms2emailserver.yourcompany.com  all
SMSGW2  {areacode}{number}@sms2emailservice.com             all
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Allowed	
Read-only users	Allowed	
Technicians	Disallowed	

Defining a New Event Notification Destination

Defines a new destination for event notifications.

```
dest_define dest=DestName
type=<SNMP|EMAIL|SMS>
< snmp_manager=SNMPManager | email_address=email |
< area_code=number number=number > | user=UserName >
[smtpgws=<SMTPGW1[,SMTPGW2]...|ALL> | smsgws=<SMSGW1[,SMSGW2]...|ALL>]
[heartbeat_test_hour=HH:MM
 [ heartbeat_test_days=<[sun][,mon][,tue][,wed][,thu][,fri][,sat]> ] ]
```

Parameters:

Name	Type	Description	Mandatory	Default
dest	Object name	Destination name.	Y	
type	Enumeration	Destination type for event notifications, which can be email, SMS or SNMP.	Y	
snmp_manager		IP address or DNS name of the SNMP manager.	N	
email_address		Email address.	N	
smtpgws	Object name	List of SMTP gateways to be used.	N	ALL (all gateways).
area_code		Area code of the cellular number for SMS notification. Use digits, '-' or '.'	N	
number		Cellular number for SMS notification. Use digits, '-' or '.'	N	

Name	Type	Description	Mandatory	Default
smsgws	Object name	SMS gateways to be used for this destination.	N	ALL (all gateways).
user	Object name	User name, where the user's email or phone are used.	N	
heartbeat_test_hour		Hour for periodic heartbeat testing in the format HH:MM	N	No heartbeat
heartbeat_test_days		List of days for heartbeat testing: a comma-separated list of 3-letter day names (such as "Mon,Fri")	N	No heartbeat

This command defines a destination for event notifications. There are three types of destinations: email, SMS and SNMP.

- *Email* destinations are used for sending notifications via email. When defining a new destination of type Email, either the email address of the recipient must be specified in *email_address* or the user name must be specified in *user* (in this case the email address of that user is used).
- *SMS* destinations are used for sending notifications via SMS to cellular phones. When defining a new destination of type SMS, either the cellular phone number of the destination must be specified in *number* or the user name must be specified in *user* (in this case the cellular phone number of that user is used). To allow correct formatting, this number should be separated into the area code and the local number.
- *SNMP* destination are used for sending notifications by SNMP traps to SNMP managers. When defining a new destination of type SNMP, the IP address of the SNMP manager should be specified.

By default, when sending an email notification, all SMTP gateways specified in Prioritizing SMTP Gateways are used, according to the order specified in that command. It is possible to define that sending emails to a specific destination will use specific SMTP gateway or gateways. This is done by specifying the smtpgws parameter.

The same logic applies to sending SMS messages. By default, SMS gateways specified in Prioritizing SMS Gateways are used, according to the order specified in this command. It is possible to define that a sending messages to a specific SMS destination will be done through specific SMS gateway or gateways.

Example:

```
xcli  -u  -c Nextra1 dest_define dest=adminemail type=EMAIL
email_address=storageadmin@yourcompany.com
```

Output:

```
Command executed successfully.
```

Example:

```
xcli -u  -c Nextra1 dest_define dest=monitoringserver type=SNMP
snmp_manager=10.170.68.111
```

Output:

```
Command executed successfully.
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- AREA_CODE_MUST_BE_SPECIFIED_FOR_DEST_TYPE
  Destination must have an area code
- AREA_CODE_NOT_ALLOWED_FOR_DEST_TYPE
  Destination cannot have an area code
- CANNOT_CHANGE_EVENT_CONF_WITH_ALERTING_EVENTS
  Cannot change event configuration while there are alerting events
  Troubleshooting: Clear all alerting events before changing event configuration
- DEST_MAX_REACHED
  Maximum number of destinations already defined
- DEST_NAME_ALREADY_EXISTS
  Destination name already exists
- DEST_NAME_IS_DESTGROUP_NAME
  Destination name already exists as a destination group name
- EMAIL_MUST_BE_SPECIFIED_FOR_DEST_TYPE
  Destination must have an email address
- EMAIL_NOT_ALLOWED_FOR_DEST_TYPE
  Destination cannot have an email address
- GATEWAY_NAME_APPEARS_TWICE
  Gateway name appears twice in the list
- GATEWAY_NAME_DOES_NOT_EXIST
  Gateway name does not exist
- NUMBER_MUST_BE_SPECIFIED_FOR_DEST_TYPE
  Destination must have a number
- NUMBER_NOT_ALLOWED_FOR_DEST_TYPE

Destination cannot have a number

- SMSGWS_NOT_ALLOWED_FOR_DEST_TYPE
  Destination cannot have SMS gateways
- SMTPGWS_NOT_ALLOWED_FOR_DEST_TYPE
  Destination cannot have SMTP gateways
- SNMP_MANAGER_MUST_BE_SPECIFIED_FOR_DEST_TYPE
  Destination must have an SNMP manager
- SNMP_MANAGER_NOT_ALLOWED_FOR_DEST_TYPE
  Destination cannot have an SNMP manager
- NO_SMS_GATEWAYS_ARE_DEFINED
  An SMS Destination cannot be defined if no SMS gateways are defined
- NO_SMTP_GATEWAYS_ARE_DEFINED
  An email destination cannot be defined if no SMTP gateways are defined
- SNMP_DESTS_CANNOT_REFER_TO_USERS
  SNMP destinations cannot refer to users
- USER_EMAIL_ADDRESS_IS_NOT_DEFINED
  User's email address is not defined
- USER_PHONE_NUMBER_IS_NOT_DEFINED
  User's phone number is not defined
- USER_NAME_DOES_NOT_EXIST
  User name does not exist
- INTERNAL_DESTS_CANNOT_REFER_TO_USERS
  Internal destinations cannot refer to users
- DEST_HEARTBEAT_DAYS_BUT_NO_HOUR
  Destination heartbeat days specified with no heartbeat hour
- DEST_CANNOT_HAVE_A_USER_AND_AN_EMAIL_ADDRESS
  Destination cannot simultaneously have an email address and refer to a user
- DEST_CANNOT_HAVE_A_USER_AND_A_PHONE_NUMBER
  Destination cannot simultaneously have a phone number address and refer to a user
- DAY_APPEARS_TWICE
  Day *'Day'* appears twice in the list
  Troubleshooting: Each day must appear at most once.

Updating an Event Notification Destination

Updates a destination.

```
dest_update dest=DestName
   [ snmp_manager=SNMPManager ] [ email_address=email ]
 [ smtpgws=<SMTPGW1[,SMTPGW2]...|ALL> ] [ area_code=number ]
 [ number=number ]  [ smsgws=<SMSGW1[,SMSGW2]...|ALL> ]
 [ user=UserName ] [ heartbeat_test_hour=HH:MM ]
        [ heartbeat_test_days=<[sun][,mon][,tue][,wed][,thu][,fri][,sat]> ]
```

Parameters:

Name	Type	Description	Mandatory	Default
dest	Object name	Destination name.	Y	
snmp_manager		IP address or DNS name of the SNMP manager.	N	Keep unchanged.
email_address		Email address.	N	Keep unchanged.
smtpgws	Object name	List of SMTP gateways to be used.	N	Keep unchanged.
area_code		Area code of the cellular number for SMS notification.	N	Keep unchanged.
number		Cellular number for SMS notification.	N	Keep unchanged.
smsgws	Object name	SMS gateways to be used.	N	Keep unchanged.
user	Object name	User name, where the user's email or phone are used.	N	Keep unchanged.
heartbeat_test_hour		Hour for periodic heartbeat testing	N	Keep unchanged.
heartbeat_test_days		List of days for heartbeat testing	N	Keep unchanged.

This command updates a destination. The parameters of this command are identical to the Defining a New Event Notification Destination command, except that the destination type cannot be changed. All relevant fields must be specified (not only the ones that are being changed).

Example:

```
xcli -u  -c Nextra1 dest_update dest=storagemanager
                    email_address=admin@yourcompany.com
```

Output:

```
Command executed successfully.
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	

User Category	Permission	Condition
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- DEST_NAME_DOES_NOT_EXIST
  Destination name does not exist
- AREA_CODE_MUST_BE_SPECIFIED_FOR_DEST_TYPE
  Destination must have an area code
- AREA_CODE_NOT_ALLOWED_FOR_DEST_TYPE
  Destination cannot have an area code
- CANNOT_CHANGE_EVENT_CONF_WITH_ALERTING_EVENTS
  Cannot change event configuration while there are alerting events
  Troubleshooting: Clear all alerting events before changing event configuration
- EMAIL_MUST_BE_SPECIFIED_FOR_DEST_TYPE
  Destination must have an email address
- EMAIL_NOT_ALLOWED_FOR_DEST_TYPE
  Destination cannot have an email address
- GATEWAY_NAME_APPEARS_TWICE
  Gateway name appears twice in the list
- GATEWAY_NAME_DOES_NOT_EXIST
  Gateway name does not exist
- NUMBER_MUST_BE_SPECIFIED_FOR_DEST_TYPE
  Destination must have a number
- NUMBER_NOT_ALLOWED_FOR_DEST_TYPE
  Destination cannot have a number
- SMSGWS_NOT_ALLOWED_FOR_DEST_TYPE
  Destination cannot have SMS gateways
- SNMP_MANAGER_MUST_BE_SPECIFIED_FOR_DEST_TYPE
  Destination must have an SNMP manager
- SNMP_MANAGER_NOT_ALLOWED_FOR_DEST_TYPE
  Destination cannot have an SNMP manager
- NO_SMS_GATEWAYS_ARE_DEFINED
  An SMS Destination cannot be defined if no SMS gateways are defined
- NO_SMTP_GATEWAYS_ARE_DEFINED
  An email destination cannot be defined if no SMTP gateways are defined
- DEST_CANNOT_HAVE_A_USER_AND_AN_EMAIL_ADDRESS
  Destination cannot simultaneously have an email address and refer to a user
- DEST_CANNOT_HAVE_A_USER_AND_A_PHONE_NUMBER
  Destination cannot simultaneously have a phone number address and refer to a user
- USER_PHONE_NUMBER_IS_NOT_DEFINED
  User's phone number is not defined
- USER_NAME_DOES_NOT_EXIST

User name does not exist

- INTERNAL_DESTS_CANNOT_REFER_TO_USERS
  Internal destinations cannot refer to users
- DEST_HEARTBEAT_DAYS_BUT_NO_HOUR
  Destination heartbeat days specified with no heartbeat hour
- SNMP_DESTS_CANNOT_REFER_TO_USERS
  SNMP destinations cannot refer to users
- USER_EMAIL_ADDRESS_IS_NOT_DEFINED
  User's email address is not defined
- SMTPGWS_NOT_ALLOWED_FOR_DEST_TYPE
  Destination cannot have SMTP gateways
- DAY_APPEARS_TWICE
  Day '*Day*' appears twice in the list
  Troubleshooting: Each day must appear at most once.

Renaming a Destination

Renames an event notification destination.

```
dest_rename dest=DestName new_name=NewDestName
```

Parameters:

Name	Type	Description	Mandatory	Default
dest	Object name	Destination to be renamed.	Y	
new_name	Object name	New name of the destination.	Y	

This command renames an event notification destination.

Example:

```
xcli -u  -c Nextra1 dest_rename dest=adminemail new_name=storagemanager
```

Output:

```
Command completed successfully
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- CANNOT_CHANGE_EVENT_CONF_WITH_ALERTING_EVENTS
  Cannot change event configuration while there are alerting events
  Troubleshooting: Clear all alerting events before changing event configuration
- DEST_NAME_DOES_NOT_EXIST
  Destination name does not exist
- DEST_NAME_IS_DESTGROUP_NAME
  Destination name already exists as a destination group name
- DEST_NAME_ALREADY_EXISTS
  Destination name already exists

Deleting a Destination

Deletes an event notification destination.

```
dest_delete dest=DestName
```

Parameters:

Name	Type	Description	Mandatory	Default
dest	Object name	Name of destination to be deleted.	Y	

The command deletes an event notification destination.

Destinations that are part of a destination group or that are used in a rule cannot be deleted.

Destinations cannot be deleted while there are uncleared alerting events.

Example:

```
xcli -u  -c Nextra1 dest_delete dest=itmanager
```

Output:

```
Command completed successfully
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Warnings:

- ARE_YOU_SURE_YOU_WANT_TO_DELETE_DESTINATION
  Are you sure you want to delete destination *Destination*?

Completion Codes:

- DEST_NAME_DOES_NOT_EXIST
  Destination name does not exist
- CANNOT_CHANGE_EVENT_CONF_WITH_ALERTING_EVENTS
  Cannot change event configuration while there are alerting events
  Troubleshooting: Clear all alerting events before changing event configuration
- DEST_IS_PART_OF_DESTGROUP
  Destination is part of a destination group and hence cannot be deleted
- DEST_APPEARS_IN_RULE
  Destination appears in a rule
  Troubleshooting: To delete the destination, first delete the rule.

Listing Event Notification Destinations

Lists event notification destinations.

```
dest_list [dest=DestName] [type=<SNMP|EMAIL|SMS>] [internal=<yes|no>]
```

Parameters:

Name	Type	Description	Mandatory	Default
dest	Object name	Destinations to be listed.	N	All destinations.
type	Enumeration	Filter only destinations of the specified type.	N	All types.
internal	Enumeration	Filter destinations by their internal XIV attribute.	N	no

This command lists the configuration of all defined destinations or of a specific destination.

Id	Name	Description	Default Position
name	Name		1
type	Type		2
email_address	Email Address		3
area_code	Area Code		4
number	Phone Number		5
snmp_manager	SNMP Manager		6
gateways	Gateways		

Id	Name	Description	Default Position
user	User		7
heartbeat_test_days	Heartbeat Days		
heartbeat_test_hour	Heartbeat Time		
creator	Creator		

Example:

```
xcli  -u  -c Nextra1 dest_list
```

Output:

```
Name              Type  Email Address                 Phone Number  Gateways
storagemanager    EMAIL storageadmin@yourcompany.com                all
monitoringserver  SNMP
```

Access Control:

User Category	Permission	Condition
Storage administrator	Conditionally Allowed	Only the technician can run this command with the internal parameter
Application administrator	Conditionally Allowed	Only the technician can run this command with the internal parameter
Read-only users	Conditionally Allowed	Only the technician can run this command with the internal parameter
Technicians	Allowed	

Testing a Destination

Sends a test message to an event notification destination.

```
dest_test dest=DestName management_ip=IPAdress [smtpgw=SMTPGatewayName]
[smsgw=SMSGatewayName] [internal=<yes|no>]
```

Parameters:

Name	Type	Description	Mandatory	Default
dest	Object name	Name of destination to be tested.	Y	
management_ip		Management IP used for sending the event notification.	Y	
smtpgw	Object name	SMTP Gateway to be tested.	N	Default system choice.

Name	Type	Description	Mandatory	Default
smsgw	Object name	SMS Gateway to be tested.	N	Default system choice.
internal	Boolean	Must be specified for XIV-internal destinations	N	no

This command tests a destination by sending a test message, SMS or SNMP trap. Note that a successful return code from this command does not ensure notification delivery.

For SNMP, email and SMS the system may fail to detect some of the problems.

For email messages, the SMTP gateway to be used should be specified (the destination is only tested through that gateway). The same applies to SMS, so that both the SMS gateways and the SMTP gateways should be specified.

Access Control:

User Category	Permission	Condition
Storage administrator	Conditionally Allowed	Only the technician can run this command with the internal parameter
Application administrator	Conditionally Allowed	Only the technician can run this command with the internal parameter
Read-only users	Disallowed	
Technicians	Allowed	

Completion Codes:

- DEST_NAME_DOES_NOT_EXIST
  Destination name does not exist
- DEST_TEST_NOT_PERFORMED_SYSTEM_BUSY
  Test of destination '*Destination Name*' not performed because the system is busy
  Troubleshooting: Please wait a few seconds and try again
- GATEWAY_NAME_DOES_NOT_EXIST
  Gateway name does not exist
- SMSGWS_MUST_BE_SPECIFIED_FOR_DEST_TYPE
  Destination must have SMS gateways
- SMSGWS_NOT_ALLOWED_FOR_DEST_TYPE
  Destination cannot have SMS gateways
- SMTPGWS_MUST_BE_SPECIFIED_FOR_DEST_TYPE
  Destination must have SMTP gateways
- SMTPGWS_NOT_ALLOWED_FOR_DEST_TYPE
  Destination cannot have SMTP gateways
- DEST_TEST_FAILED
  Test of destination '*Destination Name*' failed

- SYSTEM_HAS_NO_SUCH_EXTERNAL_IP
  The system has no such external IP address
- MODULE_CANNOT_SEND_MESSAGES
  Selected module cannot send messages
  Troubleshooting: Contact support
- ONLY_TECHNICIAN_CAN_REFER_TO_INTERNAL_EVENT_OBJECTS
  Only technician can refer to internal event objects

Creating a Destination Group

Creates an event notification destinations group.

```
destgroup_create destgroup=GroupName
```

Parameters:

Name	Type	Description	Mandatory	Default
destgroup	Object name	Destination group name.	Y	

This command creates a destination group. A destination group is simply a group of destinations, which is used by rules to send notifications to the entire group without specifying all the destinations for each rule. You can also add or remove destinations from the group, which eliminates the need to change the configuration of each rule separately.

A destination group is empty when it is created. To add a destination to a destination group, use Adding a Destination to a Destination Group.

Example:

```
xcli -u  -c Nextra1 destgroup_create destgroup=alladmins
```

Output:

```
Command executed successfully.
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- CANNOT_CHANGE_EVENT_CONF_WITH_ALERTING_EVENTS
  Cannot change event configuration while there are alerting events
  Troubleshooting: Clear all alerting events before changing event configuration

- DESTGROUP_MAX_REACHED
  Maximum number of destination groups already defined
- DESTGROUP_NAME_ALREADY_EXISTS
  Destination group name already exists
- DESTGROUP_NAME_IS_DEST_NAME
  Destination group name already exists as a destination name

Adding a Destination to a Destination Group

Adding an event notification destination to a destination group.

```
destgroup_add_dest destgroup=GroupName dest=DestName
```

Parameters:

Name	Type	Description	Mandatory	Default
destgroup	Object name	Destination group name to which to add the destination.	Y	
dest	Object name	Destination to be added to the group.	Y	

This command adds a destination to a destination group.

Command will fails if the destination group already contains the destination.

The command cannot be executed while there are uncleared alerting events.

Example:

```
xcli -u  -c Nextra1 destgroup_add_dest destgroup=alladmins dest=john
```

Output:

```
Command completed successfully
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- DESTGROUP_NAME_DOES_NOT_EXIST
  Destination group name does not exist

- CANNOT_CHANGE_EVENT_CONF_WITH_ALERTING_EVENTS
  Cannot change event configuration while there are alerting events
  Troubleshooting: Clear all alerting events before changing event configuration
- DEST_NAME_DOES_NOT_EXIST
  Destination name does not exist
- DESTGROUP_MAX_DESTS_REACHED
  Maximum number of destinations already defined in destination groups
- DESTGROUP_ALREADY_INCLUDES_DEST
  Destination group already includes destination name

Removing a Destination from Destination Group

Removes an event notification destination from a destination group.

```
destgroup_remove_dest destgroup=GroupName dest=DestName
```

Parameters:

Name	Type	Description	Mandatory	Default
destgroup	Object name	Group name.	Y	
dest	Object name	Destination to be removed from the group.	Y	

This command removes an event notification destination from a destination group.

This command cannot be executed while there are uncleared alerting events.

Example:

```
xcli -u  -c Nextra1 destgroup_remove_dest destgroup=alladmins dest=john
```

Output:

```
Command completed successfully
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- DESTGROUP_NAME_DOES_NOT_EXIST
  Destination group name does not exist
- DEST_NAME_DOES_NOT_EXIST

Destination name does not exist
- CANNOT_CHANGE_EVENT_CONF_WITH_ALERTING_EVENTS
  Cannot change event configuration while there are alerting events
  Troubleshooting: Clear all alerting events before changing event configuration
- DESTGROUP_DOES_NOT_INCLUDE_DEST
  Destination group does not include destination name

Renaming a Destination Group

Renames an event notification destination group.

```
destgroup_rename destgroup=GroupName new_name=NewGroupName
```

Parameters:

Name	Type	Description	Mandatory	Default
destgroup	Object name	Destination group to be renamed.	Y	
new_name	Object name	New name of the destination group.	Y	

This command renames an event notification destination group.

This command can not be executed while there are uncleared alerting events.

Example:

```
xcli -u  -c Nextra1 destgroup_rename destgroup=alladmins new_name=itstaff
```

Output:

```
Command completed successfully
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- DESTGROUP_NAME_DOES_NOT_EXIST
  Destination group name does not exist
- DESTGROUP_NAME_ALREADY_EXISTS
  Destination group name already exists
- CANNOT_CHANGE_EVENT_CONF_WITH_ALERTING_EVENTS

Cannot change event configuration while there are alerting events
Troubleshooting: Clear all alerting events before changing event configuration

- DESTGROUP_NAME_IS_DEST_NAME
  Destination group name already exists as a destination name

Deleting a Destination Group

Deletes an event notification destination group.

```
destgroup_delete destgroup=GroupName
```

Parameters:

Name	Type	Description	Mandatory	Default
destgroup	Object name	Name of destination group to be deleted.	Y	

This command deletes an event notification destination group.

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Warnings:

- ARE_YOU_SURE_YOU_WANT_TO_DELETE_DESTINATION_GROUP
  Are you sure you want to delete destination group *Destination Group*?

Completion Codes:

- CANNOT_CHANGE_EVENT_CONF_WITH_ALERTING_EVENTS
  Cannot change event configuration while there are alerting events
  Troubleshooting: Clear all alerting events before changing event configuration
- DESTGROUP_APPEARS_IN_RULE
  Destination Group appears in a Rule
  Troubleshooting: To delete the destination group, first delete the rule.
- DESTGROUP_NAME_DOES_NOT_EXIST
  Destination group name does not exist

Listing Destination Groups

Lists destination groups.

```
destgroup_list [ destgroup=GroupName ]
```

Parameters:

Name	Type	Description	Mandatory	Default
destgroup	Object name	Destination group to be listed.	N	All groups.

This command lists all destination groups or a specific one. All the destinations are listed for each destination group.

Id	Name	Description	Default Position
name	Name		1
dests	Destinations		2
creator	Creator		

Example:

```
xcli -u  -c Nextra1 destgroup_list
```

Output:

```
Name     Destinations
itstaff  john,michael,linda,monitoringserver
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Allowed	
Read-only users	Allowed	
Technicians	Disallowed	

Creating Event Notification Rules

Creates an event notification rule.

```
rule_create rule=RuleName
      [ min_severity=<INFORMATIONAL|WARNING|MINOR|MAJOR|CRITICAL|NONE> ]
      [ codes=EventCodes | except_codes=EventCodes ]
      [ escalation_only=<yes|no> ]
  dests=dest1,dest2,...
      [ snooze_time=SnoozeTime ]
      [ escalation_time=EscalationTime escalation_rule=EscalationRule ]
```

Parameters:

Name	Type	Description	Mandatory	Default
rule	Object name	Name of the new rule.	Y	

Name	Type	Description	Mandatory	Default
min_severity	Enumeration	Minimum event severity for rule filtering.	N	All severities.
codes		Filter only events with these codes.	N	All events.
except_codes		Filter only events with other codes.	N	All events.
escalation_only	Boolean	Specifies that this rule can only be used for escalation.	N	no
dests	Object name	Comma separated list of destinations and destination groups for event notification.	Y	
snooze_time	Integer	Snooze time in minutes.	N	No snoozing.
escalation_rule	Object name	Escalation rule.	N	
escalation_time	Integer	Escalation time in minutes. Escalation time should not be defined as shorter than snooze time. Refer to escalation_rule above for more information.	N	No escalation.

This command defines a new event notification rule. An event notification rule is used in order to determine which events should create which notifications. When an event is created, it is checked by all currently defined rules, and based on these rules the notifications are created.

Each rule has filtering and notifications configuration.

The filtering configuration controls which events match this rule. The filtering can be based on the event's code, by specifying a minimum severity. When using this configuration, each event with a severity higher or equal to the rule's `min_severity` parameter match this rule. Alternatively, the rule may match only a specific event code. The two filters can be combined together, when required for events whose severity depends on a run-time parameter.

The second part of the configuration of a rule is a list of destinations and destination groups that receive the notification when an event matches the filtering criteria. If a destination is included both in the rule and in one of the rule's destination groups, it still gets only one notification. The same applies if a

destination is included in two destination groups, or if the event matches the filtering criteria of several rules, all using the same destination.

A rule can be defined as *alerting*, which means that notifications are being sent repeatedly until the matching events are cleared using the Clearing Alerting Events command.

Note:

Clearing the event does not mean that the problem has been solved. It only means that it was noticed and there is no need to continue to send notifications.

The repeated sending of notifications can be defined by two ways:

- The `snooze` parameter causes the notifications to be sent again and again to the same destinations. The time in minutes between the repeated transmissions is determined by the `snooze` value.
- The `escalation_time` and `escalation_rule` parameters causes the notifications to be sent to the destination list of the `escalation_rule` if it is not cleared within `escalation_time` minutes.

Rules can escalate only to alerting rules (meaning to rules that have snooze or escalation definitions) in order to prevent a situation where notifications are stopped from being sent.

A rule cannot escalate to itself, and neither can it be defined in a cyclic escalation of rules.

escalation_only defines a rule without filters, which can only be used as an escalation for other rules.

The snooze time cannot be longer than the escalation time.

It is not permitted to define new rules while there are uncleared alerting events.

The following example sends alerts upon critical events to John's cellular number and to the emails of all the IT staff. The alerts will be resent every 20 minutes until the events are cleared.

Example:

```
xcli -u  -c Nextra1 rule_create rule=critical_alerts min_severity=critical
                    destinations=john-cell,itstaff snooze_time=20
```

Output:

```
Command executed successfully.
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	

User Category	Permission	Condition
Technicians	Disallowed	

Completion Codes:

- CANNOT_CHANGE_EVENT_CONF_WITH_ALERTING_EVENTS
  Cannot change event configuration while there are alerting events
  Troubleshooting: Clear all alerting events before changing event configuration
- EVENT_RULE_CANNOT_ESCALATE_TO_ITSELF
  An event rule cannot be its own escalation rule
- EVENT_RULE_MAX_REACHED
  Maximum number of event rules already defined
- EVENT_RULE_CANNOT_ESCALATE_TO_NON_ALERTING_RULES
  Event rule cannot escalate to non-alerting rule
  Troubleshooting: Alerting rule can only escalate to another escalating rule
- DEST_APPEARS_TWICE
  Destination or destination group appears twice
- EVENT_RULE_MISSING_ESCALATION_RULE
  An alerting event rule must have an escalation rule
  Troubleshooting: If escalation time is specified, then an escalation rule must be specified also.
- EVENT_RULE_MISSING_ESCALATION_TIME
  An alerting event rule must have escalation time
  Troubleshooting: If an escalation rule is specified, then escalation time must be specified also.
- EVENT_RULE_NAME_ALREADY_EXISTS
  Event rule name already exists
- EVENT_RULE_NAME_DOES_NOT_EXIST
  Event rule name does not exist
- NAME_IS_NEITHER_DEST_NOR_GROUP
  Name is neither the name of a destination group nor the name of a destination
- ESCALATION_TIME_MUST_BE_LARGER_THAN_SNOOZE_TIME
  Escalation time must be larger than snooze time
- RULE_MAX_DESTS_REACHED
  Maximum number of destinations and destination groups in a rule already defined
- EVENT_RULE_MUST_HAVE_FILTER
  An alerting event rule must have a filter, either event code or severity
- CYCLIC_ESCALATION_RULES_DEFINITION
  Event rule escalation cannot be cyclic
- EVENT_RULE_CANNOT_REFER_TO_INTERNAL_EVENT_CODES
  A user event rule cannot refer to internal event codes
- ESCALATION_EVENT_RULE_CANNOT_HAVE_FILTER
  An escalation-only event rule cannot have code or min_severity specification
- EVENT_RULE_CANNOT_HAVE_BOTH_CODES_AND_EXCEPTION_CODES
  An event rule cannot have both codes and exception codes

- EVENT_RULE_CANNOT_HAVE_A_CATEGORY
  A user event rule cannot have a category definition
- EVENT_RULE_USED_FOR_ESCALATION_MUST_BE_ALERTING
  Event rule is an escalation rule of another event rule and thus must be an alerting rule
- ESCALATION_EVENT_RULE_MUST_BE_ALERTING
  Escalation-only event rules must be alerting rules
- TOO_MANY_EVENT_CODES
  A maximum of *Maximum* event codes can be specified
- EVENT_CODE_APPEARS_TWICE
  Event code *'Code'* appears twice in the list
  Troubleshooting: Each event code must appear at most once.
- UNRECOGNIZED_EVENT_CODE
  'String' is not a recognized event code
  Troubleshooting: Consult the manual for the list of event codes

Updating an Event Notification Rule

Updates an event notification rule.

```
rule_update rule=RuleName
     [ min_severity=<INFORMATIONAL|WARNING|MINOR|MAJOR|CRITICAL|NONE> ]
     [ codes=EventCodes ] [ except_codes=EventCodes ]
     [ escalation_only=<yes|no> ]
   [ dests=dest1,dest2,...
     ] [ snooze_time=SnoozeTime ]
     [ escalation_time=EscalationTime ] [ escalation_rule=EscalationRule ]
```

Parameters:

Name	Type	Description	Mandatory	Default
rule	Object name	Name of the rule.	Y	
min_severity	Enumeration	Minimum event severity for rule filtering.	N	Leave unchanged.
codes		Filter only events with this code.	N	Leave unchanged.
except_codes		Filter only events with other codes.	N	Leave unchanged.
escalation_only	Boolean	Specifies that this rule can only be used for escalation.	N	no
dests	Object name	Comma separated list of destinations and destination groups for event notification.	N	Leave unchanged.

Name	Type	Description	Mandatory	Default
snooze_time	Integer	Snooze time in minutes.	N	Leave unchanged.
escalation_time	Integer	Escalation time in minutes.	N	Leave unchanged.
escalation_rule	Object name	Escalation rule.	N	Leave unchanged.

This command updates the configuration of an event notification rule. All parameters and their descriptions are identical to the Creating Event Notification Rules command.

Parameters which are not specified are not changed.

Example:

```
xcli -u  -c Nextra1 rule_update rule=critical_alerts min_severity=critical
                    destinations=john-cell,itstaff snooze_time=30
```

Output:

```
Command executed successfully.
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- EVENT_RULE_NAME_DOES_NOT_EXIST
  Event rule name does not exist
- CANNOT_CHANGE_EVENT_CONF_WITH_ALERTING_EVENTS
  Cannot change event configuration while there are alerting events
  Troubleshooting: Clear all alerting events before changing event configuration
- EVENT_RULE_CANNOT_ESCALATE_TO_ITSELF
  An event rule cannot be its own escalation rule
- EVENT_RULE_CANNOT_ESCALATE_TO_NON_ALERTING_RULES
  Event rule cannot escalate to non-alerting rule
  Troubleshooting: Alerting rule can only escalate to another escalating rule
- DEST_APPEARS_TWICE
  Destination or destination group appears twice
- EVENT_RULE_MISSING_ESCALATION_RULE
  An alerting event rule must have an escalation rule

Troubleshooting: If escalation time is specified, then an escalation rule must be specified also.

- EVENT_RULE_MISSING_ESCALATION_TIME

  An alerting event rule must have escalation time

  Troubleshooting: If an escalation rule is specified, then escalation time must be specified also.
- NAME_IS_NEITHER_DEST_NOR_GROUP

  Name is neither the name of a destination group nor the name of a destination
- ESCALATION_TIME_MUST_BE_LARGER_THAN_SNOOZE_TIME

  Escalation time must be larger than snooze time
- RULE_MAX_DESTS_REACHED

  Maximum number of destinations and destination groups in a rule already defined
- EVENT_RULE_MUST_HAVE_FILTER

  An alerting event rule must have a filter, either event code or severity
- CYCLIC_ESCALATION_RULES_DEFINITION

  Event rule escalation cannot be cyclic
- EVENT_RULE_USED_FOR_ESCALATION_MUST_BE_ALERTING

  Event rule is an escalation rule of another event rule and thus must be an alerting rule
- EVENT_RULE_CANNOT_REFER_TO_INTERNAL_EVENT_CODES

  A user event rule cannot refer to internal event codes
- ESCALATION_EVENT_RULE_CANNOT_HAVE_FILTER

  An escalation-only event rule cannot have code or min_severity specification
- EVENT_RULE_CANNOT_HAVE_A_CATEGORY

  A user event rule cannot have a category definition
- EVENT_RULE_CANNOT_HAVE_BOTH_CODES_AND_EXCEPTION_CODES

  An event rule cannot have both codes and exception codes
- ESCALATION_EVENT_RULE_MUST_BE_ALERTING

  Escalation-only event rules must be alerting rules
- TOO_MANY_EVENT_CODES

  A maximum of *Maximum* event codes can be specified
- EVENT_CODE_APPEARS_TWICE

  Event code '*Code*' appears twice in the list

  Troubleshooting: Each event code must appear at most once.
- UNRECOGNIZED_EVENT_CODE

  '*String*' is not a recognized event code

  Troubleshooting: Consult the manual for the list of event codes

Renaming Event Notification Rules

Renames an event notification rule.

```
rule_rename rule=RuleName new_name=NewRuleName
```

Parameters:

Name	Type	Description	Mandatory	Default
rule	Object name	Rule to be renamed.	Y	
new_name	Object name	New name of the rule.	Y	

This command renames an event notification rule.

Example:

```
xcli  -u  -c Nextra1 rule_rename rule=critical_alerts new_name=emergency_alerts
```

Output:

```
Command completed successfully
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- CANNOT_CHANGE_EVENT_CONF_WITH_ALERTING_EVENTS
  Cannot change event configuration while there are alerting events
  Troubleshooting: Clear all alerting events before changing event configuration
- EVENT_RULE_NAME_ALREADY_EXISTS
  Event rule name already exists
- EVENT_RULE_NAME_DOES_NOT_EXIST
  Event rule name does not exist

Deleting Event Notification Rules

Deletes an event notification rule.

```
rule_delete rule=RuleName
```

Parameters:

Name	Type	Description	Mandatory	Default
rule	Object name	Rule to be deleted.	Y	

This command deletes an event notification rule.

Rules that are defined as the escalation of other rules cannot be deleted.

It is not permitted to delete a rule while there are uncleared alerting events.

Example:

```
xcli -u  -c Nextra1 rule_delete rule=emergency_alerts
```

Output:

```
Command completed successfully
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Warnings:

- ARE_YOU_SURE_YOU_WANT_TO_DELETE_RULE
  Are you sure you want to delete Rule *Rule*?

Completion Codes:

- CANNOT_CHANGE_EVENT_CONF_WITH_ALERTING_EVENTS
  Cannot change event configuration while there are alerting events
  Troubleshooting: Clear all alerting events before changing event configuration
- EVENT_RULE_NAME_DOES_NOT_EXIST
  Event rule name does not exist
- EVENT_RULE_USED_FOR_ESCALATION_CAN_NOT_BE_DELETED
  Event rule is an escalation rule of another event rule and thus cannot be deleted
  Troubleshooting: Delete all escalation rules that refer to this rule as their escalation rule

Activating a Rule

Activates an event notification rule.

```
rule_activate rule=RuleName
```

Parameters:

Name	Type	Description	Mandatory	Default
rule	Object name	Name of the rule to be activated.	Y	

This command activates the specified rule. An active rule is matched against events and generates notifications. If the rule is already active, then this command has no effect.

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- EVENT_RULE_NAME_DOES_NOT_EXIST
  Event rule name does not exist

Deactivating a Rule

Deactivates an event notification rule.

```
rule_deactivate rule=RuleName
```

Parameters:

Name	Type	Description	Mandatory	Default
rule	Object name	Name of the rule to be deactivated.	Y	

This command deactivates the specified rule. A deactivated rule is not matched against events and does not generate notifications. If the rule is already inactive, then this command has no effect.

In-active rules can not be used as escalation rules.

escalation_only rules can not be deactivated.

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- EVENT_RULE_NAME_DOES_NOT_EXIST
  Event rule name does not exist
- ESCALATION_ONLY_RULES_ARE_ALWAYS_ACTIVE
  Escalation-only event rules cannot be deactivated or activated

Listing Event Notification Rules

Lists event notification rules.

```
rule_list [ rule=RuleName ]
```

Parameters:

Name	Type	Description	Mandatory	Default
rule	Object name	Rule to be listed.	N	All rules.

This command lists all event notification rules or a specific rule and their configuration.

Id	Name	Description	Default Position
name	Name		1
min_severity	Minimum Severity		2
codes	Event Codes		3
except_codes	Except Codes		4
dests	Destinations		5
active	Active		6
escalation_time	Escalation Time		
snooze_time	Snooze Time		
escalation_rule	Escalation Rule		
escalation_only	Escalation Only		7
category	Category		
creator	Creator		

Example:

```
xcli -u  -c Nextra1 rule_list
```

Output:

```
Name              Minimum Severity  Event Code  Destinations
emergency_alerts  critical          all         john-cell,itstaff
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Allowed	
Read-only users	Allowed	
Technicians	Disallowed	

Setting the Threshold for Events Notification

Redefines the threshold of a parameterized event.

```
event_redefine_threshold code=EventCode
                         severity=<INFORMATIONAL|WARNING|MINOR|MAJOR|CRITICAL|NONE>
                         threshold=<ThresholdValue|NONE>
```

Parameters:

Name	Type	Description	Mandatory	Default
code		Event code.	Y	
severity	Enumeration	Severity.	Y	
threshold	Integer	Threshold value, or NONE to indicate that an event with this severity is not created.	Y	

This command redefines the threshold of a parameterized event.

This command can be applied to parameterized events, that is events that are triggered when a certain parameter crosses a certain threshold. Using this command the user can change the threshold for event notification. Furthermore, multiple thresholds can be defined using multiple invocations of this command, one for each event severity. When the relevant parameter crosses a threshold, an event with the matching severity is created.

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- EVENT_DOES_NOT_HAVE_THRESHOLDS
  Event does not have thresholds
- EVENT_THRESHOLD_IS_ILLEGAL
  Illegal value for event threshold

Troubleshooting: Event threshold values must be monotone

- UNRECOGNIZED_EVENT_CODE

  '*String*' is not a recognized event code

  Troubleshooting: Consult the manual for the list of event codes
- LAST_EVENT_THRESHOLD_CANNOT_BE_DELETED

  Events must have at least one threshold value

Generating a Custom Event

Generates a custom event.

```
custom_event description=Description
[severity=<INFORMATIONAL|WARNING|MINOR|MAJOR|CRITICAL>]
```

Parameters:

Name	Type	Description	Mandatory	Default
description	String	Description of the event.	Y	
severity		Severity of the event.	N	Informational

This command generates a custom event. This can be used for either generating an event from a user application or host side software, or in order to test the event notifications procedures.

Example:

```
xcli -u -c Nextra1 custom_event description="Test started"
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Chapter 14. IP Configuration

The following sections describe the XIV Command Line Interface (XCLI) for IP configuration.

The sections are listed as follows:

- ipinterface_create(Creates a new IP interface for iSCSI.)
- ipinterface_update(Updates the configuration of an IP interface.)
- ipinterface_delete(Deletes an IP interface.)
- ipinterface_add_port(Adds an Ethernet port to the link aggregation group of an IP interface.)
- ipinterface_remove_port(Removes an Ethernet port from the link aggregation group of an IP interface.)
- ipinterface_list(Lists the configuration of a specific IP interface or all IP interfaces.)
- ipinterface_rename(Renames an IP interface.)
- ipinterface_list_ports(Lists all Ethernet ports together with their configuration and status.)
- ipinterface_run_arp(Prints the ARP database of the specified IP interface.)
- ipinterface_run_traceroute(Tests connectivity to a remote IP node using the ICMP trace-route mechanism.)
- support_center_define(This command defines a support center.)
- support_center_list(This command lists support centers.)
- support_center_delete(This command deletes a support center.)
- support_center_connect(This command connects to a support center.)
- support_center_disconnect(This command disconnects the IBM XIV Storage System from a support center.)
- support_center_status(This command lists information about all defined support centers.)

Creating a New IP Interface

Creates a new IP interface for iSCSI.

```
ipinterface_create ipinterface=IPInterfaceName address=IPaddress
netmask=NetworkMask [ gateway=DefaultGateway ] [ mtu=MTU ]
module=ComponentId  ports=P1,...
```

Parameters:

Name	Type	Description	Mandatory	Default
ipinterface	Object name	Name of the IP interface to be created. Do not use the names `Management` or `VPN`.	Y	

Name	Type	Description	Mandatory	Default
address		IP address of the interface.	Y	
netmask		Network mask of the interface.	Y	
gateway		IP address of the default gateway for this interface. This is optional.	N	None
mtu	Integer	Maximum Transmission Unit: The supported packet size by the connecting Ethernet switch. This is optional when the default equals 1536. MTU of up to 8192 is supported.	N	4500 for iSCSI and 1536 for Management and VPN.
module		Component identifier (rack and module) of the module containing the Ethernet ports.	Y	
ports	Integer	Comma-separated list (one or more) of port numbers.	Y	

This command defines a new IP interface for iSCSI traffic. Gateway, MTU, network mask and IP are the standard IP definitions.

The IP interface is defined on a Port Group, where a Port Group is one or more ports (all on the same module) that is linked together as an 802.3ad link aggregation group.

Each iSCSI Ethernet port can be defined as an IP interface, or groups of Ethernet iSCSI ports on the same module can be defined as a single link aggregation group. Ports defined as a link aggregation must be connected to the same Ethernet switch, and the parallel link aggregation must be defined on that switch.

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Allowed	

Completion Codes:

- IPINTERFACE_EXISTS
  IP Interface name already used
- ILLEGAL_PORT_NUMBER
  Port number is out of range
- PORT_IS_USED_IN_ANOTHER_IP_INTERFACE
  One of the physical ports specified is already assigned to an IP Interface
- PORT_REPEATS_TWICE
  Port list contains the same value more than once
- FORCE_NO_AGGREGATION_ALLOWED_FOR_SINGLE_PORT_ONLY
  More than one port specified for non-aggregated IP Interface
- IP_ADDRESS_ALREADY_USED_IN_ANOTHER_INTERFACE
  IP address is already assigned to another interface
- IPADDRESS_AND_GATEWAY_ARE_NOT_ON_SAME_SUBNET
  IP address specified for the default gateway is not in the subnet of the IP Interface
- MTU_TOO_LARGE
  Specified MTU value is too large
- BAD_PORTS_FORMAT
  Port list should be a comma separated list of positive integers
- ILLEGAL_COMPONENT_ID
  Component ID is illegal
- TOO_MANY_PORTS_IN_AGGREGATION_GROUP
  Too many physical ports for one IP interface

Updating an IP Interface

Updates the configuration of an IP interface.

```
ipinterface_update ipinterface=IPInterfaceName
[address=<IPaddress[,IPadress[,IPaddress]]>]
[netmask=NetworkMask] [gateway=DefaultGateway] [mtu=MTU]
```

Parameters:

Name	Type	Description	Mandatory	Default
ipinterface	Object name	Name of the IP interface to be updated.	Y	
address		IP address of the interface or a list of addresses for the Management and VPN interfaces.	N	Leaves the address unchanged.
netmask		Network mask of the interface.	N	Leaves the network mask unchanged.

Name	Type	Description	Mandatory	Default
gateway		IP address of the default gateway for this interface.	N	Leaves unchanged.
mtu	Integer	Maximum Transmission Unit: The packet size that is supported by the connecting Ethernet switch.	N	Keep unchanged.

This command updates the configuration of an existing IP interface.

Fields that are not specified do not change their values.

The name of the interface may either be one of the previously defined IP interfaces for iSCSI, or Management for the management IP interface, or VPN for the VPN interface.

Management ports are dedicated for CLI and GUI communications, as well as for outgoing SNMP and SMTP connections. For management interfaces, the user must specify three IP addresses (equal to the number of potential managers, minus the number of management ports).

For VPN interfaces, the user must specify two IP addresses (equal to the number of VPN ports). All VPN addresses must reside on the same subnet.

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Allowed	

Completion Codes:

- IPINTERFACE_DOES_NOT_EXIST

  IP Interface name does not exist
- IP_ADDRESS_ALREADY_USED_IN_ANOTHER_INTERFACE

  IP address is already assigned to another interface
- IPADDRESS_AND_GATEWAY_ARE_NOT_ON_SAME_SUBNET

  IP address specified for the default gateway is not in the subnet of the IP Interface
- IPINTERFACE_MANAGEMENT_DIFFERENT_SUBNET

  IP addresses management modules must all be in the same subnet
- IPINTERFACE_MANAGEMENT_MISSING_IPS

  Number of IP addresses specified is less than the number of management modules
- IPINTERFACE_MANAGEMENT_TOO_MANY_IPS

Number of IP addresses specified is larger than the number of management modules

- MTU_TOO_LARGE

  Specified MTU value is too large
- ILLEGAL_IPADDRESS

  Illegal IP address was entered

Deleting IP Interfaces

Deletes an IP interface.

```
ipinterface_delete ipinterface=IPInterfaceName
```

Parameters:

Name	Type	Description	Mandatory	Default
ipinterface	Object name	IP interface to be deleted.	Y	

This command deletes the IP interface. Only the interfaces defined for iSCSI traffic can be deleted. Management and VPN interfaces cannot be deleted.

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Allowed	

Completion Codes:

- IPINTERFACE_DOES_NOT_EXIST

  IP Interface name does not exist
- COMMAND_NOT_ALLOWED_ON_MANAGEMENT_INTERFACE

  Operation is not allowed on Management IP Interface
- IPINTERFACE_HAS_CONNECTIVITY

  IP interface has connectivity defined to another machine

Adding Ethernet Ports to IP Interfaces

Adds an Ethernet port to the link aggregation group of an IP interface.

```
ipinterface_add_port ipinterface=IPInteraceName port=P
```

Parameters:

Name	Type	Description	Mandatory	Default
ipinterface	Object name	IP interface to which the port is to be added.	Y	
port	Integer	Number of the port to be added to the group.	Y	

This command adds an Ethernet port to the link aggregation group of an IP interface.

The specified port is added to the link aggregation group of the specified IP interface.

Ports defined as a link aggregation group must be connected to the same Ethernet switch, and a parallel link aggregation group must be defined on that Ethernet switch.

The module is not provided, as it must be the module of the other ports of the interface.

This command cannot be applied to Management or VPN interfaces.

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Allowed	

Completion Codes:

- IPINTERFACE_DOES_NOT_EXIST
  IP Interface name does not exist
- ILLEGAL_PORT_NUMBER
  Port number is out of range
- PORT_ALREADY_IN_INTERFACE
  Port is already part of the specified IP Interface
- PORT_IS_USED_IN_ANOTHER_IP_INTERFACE
  One of the physical ports specified is already assigned to an IP Interface
- COMMAND_NOT_ALLOWED_ON_MANAGEMENT_INTERFACE
  Operation is not allowed on Management IP Interface
- IPINTERFACE_NOT_AGGREGATED
  Ports cannot be added to a non-aggregated IP Interface

Removing Ethernet Ports from IP Interfaces

Removes an Ethernet port from the link aggregation group of an IP interface.

```
ipinterface_remove_port ipinterface=IPInteraceName port=P
```

Parameters:

Name	Type	Description	Mandatory	Default
ipinterface	Object name	IP interface from which the port is to be removed.	Y	
port	Integer	Number of the port to be removed from the group.	Y	

This command removes the specified port from the link aggregation group of the specified IP interface. The module does not need to be specified, because it is the same module as the other ports of the IP interface.

The last port of the IP interface cannot be removed.

If the IP interface must be moved to a different module, first delete the interface and then recreate it. This command cannot be applied to Management or VPN interfaces.

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Allowed	

Completion Codes:

- IPINTERFACE_DOES_NOT_EXIST
  IP Interface name does not exist
- ILLEGAL_PORT_NUMBER
  Port number is out of range
- PORT_NOT_IN_INTERFACE
  Port is not part of the specified IP Interface
- COMMAND_NOT_ALLOWED_ON_MANAGEMENT_INTERFACE
  Operation is not allowed on Management IP Interface
- IPINTERFACE_LAST_PORT
  Last port in IP Interface cannot be removed

Listing IP Interface Configuration

Lists the configuration of a specific IP interface or all IP interfaces.

```
ipinterface_list [ ipinterface=IPInterfaceName | address=IPAddress ]
```

Parameters:

Name	Type	Description	Mandatory	Default
ipinterface	Object name	IP interface to be listed.	N	All interfaces
address		IP address of the interface to be listed.	N	All interfaces

This command lists configuration information for the specified IP interface or for all IP interfaces (including Management). The Management or VPN name can only be used to view the configuration of the management of VPN interfaces.

The following information is listed:

- Name
- Type (iSCSI/management)
- IP address (or comma separated addresses for management and VPN)
- Network mask
- Default gateway
- MTU
- Module (for iSCSI only)
- Comma separated list of ports (for iSCSI only)

Id	Name	Description	Default Position
name	Name		1
type	Type		2
address	IP Address		3
netmask	Network Mask		4
gateway	Default Gateway		5
mtu	MTU		6
module	Module		7
ports	Ports		8

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Allowed	
Read-only users	Allowed	
Technicians	Allowed	

Renaming an IP Interface

Renames an IP interface.

```
ipinterface_rename ipinterface=IPInterfaceName new_name=NewName
```

Parameters:

Name	Type	Description	Mandatory	Default
ipinterface	Object name	Original name of the IP interface.	Y	
new_name	Object name	New name of the IP interface.	Y	

This command renames an IP interface. The IP interface must be unique in the system. This command cannot be applied to Management or VPN interfaces.

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Allowed	

Completion Codes:

- IPINTERFACE_DOES_NOT_EXIST
  IP Interface name does not exist
- IPINTERFACE_EXISTS
  IP Interface name already used
- COMMAND_NOT_ALLOWED_ON_MANAGEMENT_INTERFACE
  Operation is not allowed on Management IP Interface

Showing the Status and Configuration of Ethernet Ports

Lists all Ethernet ports together with their configuration and status.

```
ipinterface_list_ports
```

This command lists all Ethernet ports together with their configuration and status.

All physical Ethernet ports used to connect to the user's network are listed. This list shows the following information:

- Component ID (Module number for iSCSI or switch number for management/field technician port)
- Port number on module/switch

- For management/VPN/field technician: "management"/"VPN"/"field technician"
- IP interface containing the ports (or none, if port is not configured as part of IP interface)
- Status up/down
- Auto-negotiation: Half-full duplex, 1000/100/10

Id	Name	Description	Default Position
index	Index		1
role	Role		2
ip_interface_name	IP Interface		3
connected_component	Connected Component		4
is_link_up	Link Up?		5
negotiated_speed_Mbs	Negotiated Speed (MB/s)		6
is_full_duplex	Full Duplex?		7
module_id	Module		8

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Allowed	
Read-only users	Allowed	
Technicians	Allowed	

Printing the ARP Database of an IP Interface

Prints the ARP database of the specified IP interface.

```
ipinterface_run_arp localipaddress=IPaddress
```

Parameters:

Name	Type	Description	Mandatory	Default
localipaddress		IP address of the IP interface for which the ARP database should be printed.	Y	

This command prints the ARP database of an IP interface, meaning a list of pairs of IP addresses and their associated Ethernet MAC addresses. The IP address must be one of the IP addresses defined for iSCSI IP interfaces, or the Management or VPN name.

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Allowed	

Completion Codes:

- NO_IP_INTERFACE_MATCHES_CRITERIA
  No IP Interface matches given criteria
- MORE_THAN_ONE_IP_INTERFACE_MATCHES
  More than one IP Interface matches given criteria
- TRNS_SUBPROCESS_SYSCALL_FAILED
  System call failed when running a subprocess.
- TRNS_SUBPROCESS_INVALID_USAGE
  Invalid usage of interface
  Troubleshooting: Contact support
- TRNS_SUBPROCESS_NO_MEMORY
  Not enough memory present to complete operation
  Troubleshooting: Contact support
- TRNS_SHELL_COMMAND_SYSCALL_FAILED
  A system call failed for the shell command.
  Troubleshooting: Contact support
- TRNS_SHELL_COMMAND_INTERNAL_FAILURE
  Internal failure prevented shell command from running
  Troubleshooting: Contact support
- TRNS_SHELL_COMMAND_NO_RESOURCES
  Too many shell commands are run concurrently.
  Troubleshooting: Contact support

Testing the Traceroute to a Remote IP

Tests connectivity to a remote IP node using the ICMP trace-route mechanism.

```
ipinterface_run_traceroute localipaddress=IPaddress remote=remoteHost
```

Parameters:

Name	Type	Description	Mandatory	Default
localipaddress		IP address of the IP interface for which the traceroute command is run.	Y	
remote		IP address or DNS for the traceroute test.	Y	

This command runs a route trace to the specified remote host through the specified IP interface. The IP address must be one of the IP addresses defined for iSCSI IP interfaces or the Management or VPN name.

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Allowed	

Completion Codes:

- NO_IP_INTERFACE_MATCHES_CRITERIA
  No IP Interface matches given criteria
- MORE_THAN_ONE_IP_INTERFACE_MATCHES
  More than one IP Interface matches given criteria
- TRNS_SUBPROCESS_SYSCALL_FAILED
  System call failed when running a subprocess.
- TRNS_SUBPROCESS_INVALID_USAGE
  Invalid usage of interface
  Troubleshooting: Contact support
- TRNS_SUBPROCESS_NO_MEMORY
  Not enough memory present to complete operation
  Troubleshooting: Contact support
- TRNS_SHELL_COMMAND_SYSCALL_FAILED
  A system call failed for the shell command.
  Troubleshooting: Contact support
- TRNS_SHELL_COMMAND_INTERNAL_FAILURE
  Internal failure prevented shell command from running
  Troubleshooting: Contact support
- TRNS_SHELL_COMMAND_NO_RESOURCES
  Too many shell commands are run concurrently.
  Troubleshooting: Contact support

Defining a Support Center

This command defines a support center.

```
support_center_define support_center=SupportCenterName address=SupportCenterIPAddress
[ port=port ] [ priority=priority ]
```

Parameters:

Name	Type	Description	Mandatory	Default
support_center	Object name	The name of the support center server	Y	
address		The IP address of the support center server	Y	
port	Positive integer	The TCP port to connect to on the support center	N	22
priority		The priority of the support center (support centers with a higher priority will be connected first)	N	0

This command defines a support center.

Access Control:

User Category	Permission	Condition
Storage administrator	Disallowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Allowed	

Completion Codes:

- MAX_SUPPORT_CENTERS_DEFINED
  Maximum number of support centers is already defined.

Listing Support Centers

This command lists support centers.

```
support_center_list
```

Parameters:

Name	Type	Description	Mandatory	Default

This command displays the following information about all defined support centers:

- Name
- IP Address

- Port
- Priority

Id	Name	Description	Default Position
name	Name		1
address	Address		2
port	Port		3
priority	Priority		4

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Allowed	

Deleting a Support Center

This command deletes a support center.

```
support_center_delete support_center=SupportCenterName
```

Parameters:

Name	Type	Description	Mandatory	Default
support_center	Object name	The name of the support center to delete.	Y	

This command deletes a support center. Sessions that belong to this support center are disconnected, even if they are open at the time of deletion.

Access Control:

User Category	Permission	Condition
Storage administrator	Disallowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Allowed	

Warnings:

- ARE_YOU_SURE_TO_DELETE_THE_SUPPORT_CENTER
  Are you sure you want to delete Support Center?.

Completion Codes:

- SUPPORT_CENTER_NOT_DEFINED
  Support Center is not defined.

Connecting to a Support Center

This command connects to a support center.

```
support_center_connect [ timeout=Timeout [ idle_timeout=IdleTimeout ] ] [ module=module ]
```

Parameters:

Name	Type	Description	Mandatory	Default
timeout		Specified duration of the session. After the duration elapses, the session will be disconnected.	N	none
idle_timeout		Specified idle time for the session after which it will be disconnected.	N	[timeout]
module		The module from which the connection to the support center should be initiated	N	[the module that handled the CLI request]

This command connects to a support center. If the support center does not exist, the command fails.

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Allowed	

Completion Codes:

- NO_SUPPORT_CENTERS_ARE_DEFINED
  No support centers are defined.
- REMOTE_SUPPORT_CONTROL_FAILURE
  Remote support control failure: *Error Message*.
- IDLE_TIMEOUT_MUST_BE_LOWER_THAN_TIMEOUT
  The idle timeout of the connection must be lower than its timeout.
- REMOTE_SUPPORT_CONTROL_ABNORMAL_FAILURE

Unable to perform remote support function.

- MODULE_HAS_NO_VALID_PORT

  Module has no valid port for support center connection.

Disconnecting from a Support Center

This command disconnects the IBM XIV Storage System from a support center.

```
support_center_disconnect
```

Parameters:

Name	Type	Description	Mandatory	Default

This command disconnects the IBM XIV Storage System from a support center.

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Allowed	

Completion Codes:

- **Target event not found. href = NO_SUPPORT_CENTER_DEFINED**
- **Target event not found. href = ARE_YOU_SURE_TO DISCONNECT_FROM_SUPPORT_CENTER**
- REMOTE_SUPPORT_CONTROL_FAILURE

  Remote support control failure: *Error Message*.
- REMOTE_SUPPORT_CONTROL_ABNORMAL_FAILURE

  Unable to perform remote support function.

Presenting Status of a Support Center

This command lists information about all defined support centers.

```
support_center_status
```

Parameters:

Name	Type	Description	Mandatory	Default

This command lists information about all defined support centers.

Id	Name	Description	Default Position
name	Name		1
value	Value		2

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Allowed	

Completion Codes:

- REMOTE_SUPPORT_CONTROL_FAILURE
  Remote support control failure: *Error Message*.
- REMOTE_SUPPORT_CONTROL_ABNORMAL_FAILURE
  Unable to perform remote support function.

Chapter 15. Access Control

The following sections describe the XIV Command Line Interface (XCLI) for user access control.

The sections are listed as follows:

- user_define(Defines a new user.)
- user_update(Updates a user.)
- user_list(Lists all users or a specific user.)
- user_rename(Renames a user.)
- user_delete(Deletes a user.)
- user_group_create(Creates a user group.)
- user_group_update(Updates a user group.)
- user_group_add_user(Adds a user to a user group.)
- user_group_remove_user(Removes a user from a user group.)
- user_group_list(Lists all user groups or a specific one.)
- user_group_rename(Renames a user group.)
- user_group_delete(Deletes a user group.)
- access_define(Defines an association between a user group and a host.)
- access_delete(Deletes an access control definition.)
- access_list(Lists access control definitions.)
- ldap_config_set(Configures general system parameters governing user authentication against LDAP servers)
- ldap_config_get(Lists LDAP configuration parameters.)
- ldap_mode_set(Enables/disables LDAP-based authentication mode.)
- ldap_mode_get(Lists LDAP-based authentication mode.)
- ldap_add_server(Adds an LDAP server definition.)
- ldap_remove_server(Removes an LDAP server definition.)
- ldap_list_servers(Lists LDAP servers defined in the system.)
- ldap_user_test(Tests user credentials in LDAP authentication.)
- smis_add_user(Adds a user to the SMIS agent in the system.Adds a user to the SMIS agent in the system.)
- smis_list_users(Lists users defined in the SMIS agent in the system.Lists users defined in the SMIS agent in the system.)
- smis_remove_user(Removes a user from the SMIS agent in the system.Removes a user from the SMIS agent in the system.)

Defining a New User

Defines a new user.

```
user_define user=UserName password=Password password_verify=Password
 category=<storageadmin|applicationadmin|readonly>
[ email_address=EmailAddress ]
[ area_code=AreaCode number=PhoneNumber ]
```

Parameters:

Name	Type	Description	Mandatory	Default
user	Object name	User name. User names are lower case.	Y	
password		Password of the user to be created. The password must have between 6 and 12 characters consisting of: a-z. A-Z or 0-9. Password is case sensitive.	Y	
password_verify		Password verification, which must be equal to the value of `password`.	Y	
category	Enumeration	The role of the user to be created.	Y	
email_address		Email address of this user. The email address specified here can be used for event notification. Entering this address is optional. The email address format is any legal email address.	N	
number		Cellular phone number of the user for event notification via SMS, excluding the area code. Phone numbers and area codes can be a maximum of 63 digits, dashes (-) and periods (.)	N	

Name	Type	Description	Mandatory	Default
area_code		Area code of the cellular phone number of the user. Phone numbers and area codes can be a maximum of 63 digits, dashes (-) and periods (.)	N	

This command adds a new user. Email address and phone number are optional and can be used for event notification. The category (user role) may be only one of those specified above (other categories contain only a single predefined user).

Access_all can be specified for application administrators only. When it is specified, it means that the user has a application administrator access level for all volumes, and can perform operations on all volumes and not just a subset of the specific volume.

The maximum number of users is 32.

Two predefined users are set system-wide: Admin and Technician.

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- USER_NAME_ALREADY_EXISTS
  User name already exists
- MAX_USERS_REACHED
  Maximum number of users already defined
- PASSWORDS_DO_NOT_MATCH
  Passwords must be identical
- USER_PHONE_NUMBER_MUST_ACCOMPANY_AREA_CODE
  Phone numbers and area code must be defined together
- LDAP_AUTHENTICATION_IS_ACTIVE
  Command is not available while LDAP authentication is active

Updating a User Definition

Updates a user.

```
user_update user=UserName [ password=Password password_verify=Password ]
 [ email_address=EmailAddress ][ area_code=AreaCode ]
 [ number=PhoneNumber ]
```

Parameters:

Name	Type	Description	Mandatory	Default
user	Object name	The name of the user to be updated. User names are lower case.	Y	
password		New password. Users can only change their own passwords. The password consists of 6-12 characters, comprised of a-z, A-Z and 0-9, and is case sensitive.	N	Retains the current password.
password_verify		Verification of the password: Must be equal to `the password`.	N	Retains the current password.
email_address		Email address of this user (for event notification).	N	Leaves the current email address.
number		Cellular phone number of the user (for event notification via SMS) excluding the area code.	N	Leaves the current number.
area_code		Area code of the cellular phone number of the user.	N	Leaves the current area code.

This command updates the configuration of a user.

A user with the predefined password admin can change the passwords of other users. The Category (Role) of a user cannot be changed. The user Technician does not require a phone number and an email address. Limitations on password changes are as follows:

- Any user can change his/her own password (excluding read-only users).
- The predefined admin user can change all passwords, excluding the user Technician.

- Passwords are case sensitive.

Access Control:

User Category	Permission	Condition
Storage administrator	Conditionally Allowed	A user other than admin may only change its won configuration.
Application administrator	Conditionally Allowed	A user of this category may only change its own configuration.
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- USER_NAME_DOES_NOT_EXIST
  User name does not exist
- PASSWORDS_DO_NOT_MATCH
  Passwords must be identical
- USER_PHONE_NUMBER_MUST_ACCOMPANY_AREA_CODE
  Phone numbers and area code must be defined together
- ADMIN_CAN_NOT_CHANGE_TECHNICIAN_USER
  Administrators cannot change the details of the technician user
- SMS_DESTINATION_REFERS_TO_USER
  An SMS destination refers to the user and therefore it must have a phone number and an area code
- EMAIL_DESTINATION_REFERS_TO_USER
  An email destination refers to the user and therefore it must have an email address
- USER_NOT_ALLOWED_TO_CHANGE_OTHER_USERS
  This user is not allowed to change the details of other users
- USER_NOT_ALLOWED_TO_HAVE_PHONE_NUMBER
  User not allowed to have phone number
- USER_NOT_ALLOWED_TO_HAVE_EMAIL_ADDRESS
  User not allowed to have email address
- USER_NOT_ALLOWED_TO_CHANGE_PASSWORDS
  This user cannot change passwords of other users
- USER_CANNOT_BE_UPDATED_WHILE_LDAP_AUTHENTICATION_IS_ACTIVE
  User cannot be updated while LDAP authentication is active

Listing Users

Lists all users or a specific user.

```
user_list [ user=UserName | show_users=<all|active> ]
```

Parameters:

Name	Type	Description	Mandatory	Default
user	Object name	The user to be listed.	N	All users.
show_users	Enumeration	Indicates whether all internal users will be listed, or only internal users that are active.	N	active

This command lists all users or a specific user. The following information is listed:

- User name: Lower case
- Category
- Email address
- Phone number
- Phone area code
- Containing user group

Passwords are not shown in the list.

Id	Name	Description	Default Position
name	Name		1
category	Category		2
group	Group		3
active	Active		4
email_address	Email Address		5
area_code	Area Code		6
number	Phone Number		7
access_all	Access All		8
creator	Creator		

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Allowed	
Read-only users	Allowed	
Technicians	Disallowed	

Renaming Users

Renames a user.

```
user_rename user=UserName new_name=NewName
```

Parameters:

Name	Type	Description	Mandatory	Default
user	Object name	User to be renamed. User names are lower case.	Y	
new_name	Object name	New name of the user.	Y	

This command renames a user.

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- USER_NAME_DOES_NOT_EXIST
  User name does not exist
- USER_NAME_ALREADY_EXISTS
  User name already exists
- USER_CANNOT_BE_RENAMED
  User cannot be renamed
- LDAP_AUTHENTICATION_IS_ACTIVE
  Command is not available while LDAP authentication is active

Deleting a User

Deletes a user.

```
user_delete user=UserName
```

Parameters:

Name	Type	Description	Mandatory	Default
user	Object name	User to be deleted.	Y	

This command deletes a user.

Existing objects created by this user contain an empty reference to the creating user after it has been deleted.

Two predefined users are set system-wide: Admin and Technician. Predefined users cannot be deleted or renamed.

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- LDAP_AUTHENTICATION_IS_ACTIVE
  Command is not available while LDAP authentication is active
- USER_NAME_DOES_NOT_EXIST
  User name does not exist
- USER_CANNOT_BE_DELETED
  User cannot be deleted
- USER_IS_REFERRED_TO_BY_DEST
  User is referred to by an event destination and therefore cannot be deleted

Creating User Groups

Creates a user group.

```
user_group_create user_group=UserGroup  [ access_all=<yes|no> ]   [ ldap_role=LdapRole ]
```

Parameters:

Name	Type	Description	Mandatory	Default
user_group	Object name	Name of the user group to be created.	Y	
access_all	Boolean	Allows application administrators the ability to perform their specified operations on all volumes and not just a subset of the specific volumes	N	no
ldap_role	String	The value representing the user group in LDAP.	N	[none]

This command creates a new user group.

A user group is a group of application administrators who share the same set of snapshot creation limitations. After user groups are created, the limitations of all

the users in a user group can be updated using a single command. These limitations are enforced by associating the user groups with hosts or clusters.

Storage administrators create user groups and control the various application administrator's permissions. Hosts and clusters can be associated with only a single user group. When a user belongs to a user group that is associated with a host, it is possible to manage snapshots of the volumes mapped to that host.

User groups have the following limitations:

- Only users who are defined as application administrators can be assigned to a group.
- A user can belong to only a single user group.
- A user group can contain up to eight users.

User and host associations have the following properties:

- User groups can be associated with both hosts and clusters. This allows limiting application administrator access to specific volumes.
- A host that is part of a cluster cannot also be associated with a user group.
- When a host is added to a cluster the host's associations are broken. Limitations on the management of volumes mapped to the host is controlled by the cluster's association.
- When a host is removed from a cluster, the host's associations become the cluster 's associations, this allows continued mapping of operations so that all scripts continue to work.

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- USER_GROUP_NAME_ALREADY_EXISTS
  User group name already exists
- MAX_USER_GROUPS_REACHED
  Maximum number of user groups already defined
- LDAP_ROLE_ALREADY_USED
  LDAP role is already in use in LDAP configuration or in a user group

Updating a User Group

Updates a user group.

```
user_group_update  user_group=UserGroup [ ldap_role=LdapRole ] [ access_all=<yes|no> ]
```

Parameters:

Name	Type	Description	Mandatory	Default
user_group	Object name	The name of the user group to be updated.	Y	
ldap_role	String	The value representing the user group in LDAP.	N	Keep current LDAP role.
access_all	Boolean	Assigns application administration access level for all volumes.	N	no

Updates a user group.

Example:

```
xcli  -u -c XIV1 user_group_update user_group=Development ldap_role=storagedev
```

Output:

```
Command executed successfully.
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- USER_GROUP_NAME_DOES_NOT_EXIST
  User group name does not exist
- LDAP_ROLE_ALREADY_USED
  LDAP role is already in use in LDAP configuration or in a user group

Adding Users to a User Groups

Adds a user to a user group.

```
user_group_add_user user_group=UserGroup user=UserName
```

Parameters:

Name	Type	Description	Mandatory	Default
user_group	Object name	User group into which the user is to be added.	Y	
user	Object name	User to be added to the user group.	Y	

This command adds a user to a user group.

A user group can contain up to eight users.

A user may belong to only one user group.

Only users defined as Application Administrators can be assigned to a group.

This command fails when the user already belongs to the user group.

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- USER_GROUP_NAME_DOES_NOT_EXIST
  User group name does not exist
- USER_NAME_DOES_NOT_EXIST
  User name does not exist
- USER_ALREADY_INCLUDED_IN_ANOTHER_GROUP
  User is included in another user group
- USER_GROUP_ALREADY_INCLUDES_USER
  User group already includes user
- ONLY_APPLICATION_ADMIN_USERS_CAN_BE_GROUPED
  User groups can only contain application administrators
- USER_GROUP_HAS_MAXIMUM_NUMBER_OF_USERS
  User group already has the maximum number of users
- LDAP_AUTHENTICATION_IS_ACTIVE
  Command is not available while LDAP authentication is active

Removing a User from a User Group

Removes a user from a user group.

```
user_group_remove_user user_group=UserGroup user=UserName
```

Parameters:

Name	Type	Description	Mandatory	Default
user_group	Object name	User group.	Y	
user	Object name	User to be removed.	Y	

This command removes a user from a user group.

This command fails when the user does not belong to the user group.

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Warnings:

- ARE_YOU_SURE_YOU_WANT_TO_REMOVE_USER
  Are you sure you want to remove user from user group?

Completion Codes:

- USER_GROUP_NAME_DOES_NOT_EXIST
  User group name does not exist
- USER_NAME_DOES_NOT_EXIST
  User name does not exist
- USER_GROUP_DOES_NOT_INCLUDE_USER
  User group does not include user
- LDAP_AUTHENTICATION_IS_ACTIVE
  Command is not available while LDAP authentication is active

Listing User Groups

Lists all user groups or a specific one.

```
user_group_list [ user_group=UserGroup ]
```

Parameters:

Name	Type	Description	Mandatory	Default
user_group	Object name	The user group to be listed.	N	All user groups.

This command lists all user groups or a specific one. All the users included in the user group are listed.

Id	Name	Description	Default Position
name	Name		1
access_all	Access All		2
ldap_role	LDAP Role		3
users	Users		4
creator	Creator		

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Allowed	
Read-only users	Allowed	
Technicians	Disallowed	

Renaming User Groups

Renames a user group.

```
user_group_rename user_group=UserGroup new_name=NewName
```

Parameters:

Name	Type	Description	Mandatory	Default
user_group	Object name	User group to be renamed.	Y	
new_name	Object name	New name of the user group.	Y	

This command renames a user group.

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- USER_GROUP_NAME_DOES_NOT_EXIST
  User group name does not exist
- USER_GROUP_NAME_ALREADY_EXISTS
  User group name already exists

Deleting a User Group

Deletes a user group.

```
user_group_delete user_group=UserGroup
```

Parameters:

Name	Type	Description	Mandatory	Default
user_group	Object name	User group to be deleted.	Y	

This command deletes a user group.

A user group can be deleted, even when it is associated with hosts or clusters.

A user group can be deleted, even when it contains users. Deleting the user group does not delete the users contained in this group.

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- USER_GROUP_NAME_DOES_NOT_EXIST
  User group name does not exist
- LDAP_AUTHENTICATION_IS_ACTIVE
  Command is not available while LDAP authentication is active

Adding an Access Control Definition

Defines an association between a user group and a host.

```
access_define user_group=UserGroup < host=Host | cluster=Cluster >
```

Parameters:

Name	Type	Description	Mandatory	Default
user_group	Object name	User group to be associated with the host or cluster.	Y	
host	Object name	Host to be associated with the user group.	N	

Name	Type	Description	Mandatory	Default
cluster	Object name	Cluster to be associated with the user group.	N	

This command associates a user group with a host or a cluster. Hosts and clusters can be associated with only a single user group.

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- USER_GROUP_NAME_DOES_NOT_EXIST
  User group name does not exist
- CLUSTER_BAD_NAME
  Cluster name does not exist
- HOST_BAD_NAME
  Host name does not exist
- HOST_BELONGS_TO_CLUSTER
  Host is part of a cluster

Deleting an Access Control Definition

Deletes an access control definition.

```
access_delete user_group=UserGroup < host=Host | cluster=Cluster >
```

Parameters:

Name	Type	Description	Mandatory	Default
user_group	Object name	User group of the access control definition to be deleted.	Y	
host	Object name	Host of the access control definition to be deleted.	N	
cluster	Object name	Cluster of the access control definition to be deleted.	N	

This command deletes the association between the user group and host or cluster. The operation fails if no such access definition exists. When a host is removed from a cluster, the host's associations become the cluster's associations, thereby allowing continued mapping of operations so that all scripts continue to work.

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- USER_GROUP_NAME_DOES_NOT_EXIST
  User group name does not exist
- USER_GROUP_DOES_NOT_HAVE_ACCESS_TO_CLUSTER
  User Group does not have access to cluster
- CLUSTER_BAD_NAME
  Cluster name does not exist
- HOST_BAD_NAME
  Host name does not exist
- HOST_BELONGS_TO_CLUSTER
  Host is part of a cluster
- USER_GROUP_DOES_NOT_HAVE_ACCESS_TO_HOST
  User Group does not have access to host

Listing Access Control Definitions

Lists access control definitions.

```
access_list [ user_group=UserGroup ] [ host=Host | cluster=Cluster ]
```

Parameters:

Name	Type	Description	Mandatory	Default
user_group	Object name	Filters the access control listing so that it only shows this user group.	N	All user groups.
host	Object name	Filters the access control listing so that it only shows this host.	N	All hosts.

Name	Type	Description	Mandatory	Default
cluster	Object name	Filters the access control listing so that it only shows this cluster.	N	All clusters.

This command lists access control definitions. The list can be displayed for all access control definitions or it can be filtered for a specific user group, host/cluster or both.

Id	Name	Description	Default Position
type	Type		1
name	Name		2
user_group	User Group		3

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Allowed	
Read-only users	Allowed	
Technicians	Disallowed	

Completion Codes:

- HOST_BAD_NAME
  Host name does not exist
- CLUSTER_BAD_NAME
  Cluster name does not exist
- USER_GROUP_NAME_DOES_NOT_EXIST
  User group name does not exist

Configuring LDAP in the System

Configures general system parameters governing user authentication against LDAP servers

```
ldap_config_set
    [ base_dn=LdapDn ]
    [ xiv_group_attrib=LdapAttrib ]
    [ storage_admin_role=LdapAttribute ]
    [ read_only_role=LdapRole ]
    [ use_ssl=<yes|no> ]
    [ user_id_attrib=LdapAttrib ]
    [ session_cache_period=Minutes ]
    [ bind_time_limit=Seconds ]
    [ first_expiration_event=Days ]
    [ second_expiration_event=Days ]
    [ third_expiration_event=Days ]
    [ version=<2|3> ]
```

Parameters:

Name	Type	Description	Mandatory	Default
base_dn		Base_DN of the LDAP directory	N	none
xiv_group_attrib	String	LDAP attribute designated to hold XIV-mapped roles	N	none
storage_admin_role	String	LDAP value mapped to the XIV storage administrator role	N	none
read_only_role	String	LDAP value mapped to the XIV read only role	N	none
use_ssl	Boolean	Indicates if secure LDAP is mandated	N	no
user_id_attrib	String	LDAP attribute set to identify the user (in addition to user DN) when recording user operations in the XIV event log	N	objectSiD
session_cache_period	Positive integer	Duration user credentials are kept before attempt to re-login the user	N	20
bind_time_limit	Positive integer	Duration after which the next LDAP server on the ldap server list will be called	N	0 (with bind_time_limit=0 we approach the LDAP server for every command. the performance issues depend on its availability.)
first_expiration_event	Positive integer	Number of days before expiration of certificate to set first alert (severity: warning)	N	30/14/7 (third is smallest)
second_expiration_event	Positive integer	Number of days before expiration of certificate to set second alert (severity: warning)	N	30/14/7 (third is smallest)

Name	Type	Description	Mandatory	Default
third_expiration_event	Positive integer	Number of days before expiration of certificate to set third alert (severity: warning)	N	30/14/7 (third is smallest)
version	Positive integer	Version of LDAP used	N	Range 2, 3

This command configures general system parameters governing user authentication against LDAP servers.

LDAP access permissions which are not enforced for XIV predefined users. These XIV predefined users are authenticated by the IBM XIV Storage System and not by LDAP even if LDAP authentication is enabled.

Predefined user names include the following:

- admin
- smis_user
- technician
- xiv_development
- xiv_maintenance

Whenever an LDAP user - with a user name identical to any of the predefined names - tries to log into XIV (when LDAP authentication is enabled), that user will normally be denied access, since it is not authenticated against LDAP (but rather against XIV), and the user's (LDAP) password will likely not match the XIV password. However, logging into XIV using the password of the corresponding predefined user, the user will be granted the rights of the corresponding predefined user, regardless of its LDAP settings (e.g., association with an XIV application admin role), as LDAP is not consulted for predefined users.

Example:

```
xcli  -u -c XIV1 ldap_config_set
base_dn version xiv_group_attrib
storage_admin_role read_only_role
session_cache_period bind_time_limit
use_ssl user_id_attrib first_expiration_event
second_expiration_event
third_expiration_event
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- LDAP_IS_NOT_FULLY_CONFIGURED
  LDAP is not fully configured
  Troubleshooting: Check your settings.
- LDAP_CONFIG_CHANGE_IS_ILLEGAL_WHEN_AUTHENTICATION_IS_ACTIVE
  This LDAP configuration change is invalid when LDAP configuration is active
  Troubleshooting: Disable LDAP-based authentication and then change LDAP configuration.
- LDAP_ROLE_ALREADY_USED
  LDAP role is already in use in LDAP configuration or in a user group
- NO_LDAP_SERVERS_WITH_CERTIFICATE_ARE_DEFINED
  No LDAP servers with an LDAP certificate are defined in the system
- INVALID_EXPIRATION_EVENT_DATES
  Dates for expiration events must be in ascending order

Listing LDAP Configuration Parameters

Lists LDAP configuration parameters.

```
ldap_config_get
```

Parameters:

Name	Type	Description	Mandatory	Default

This command lists LDAP configuration parameters. A successful execution of this command depends on corresponding to a valid LDAP server. The output of the command does not list LDAP servers. Such data can be found by using the ldap_list_servers command.

Example:

```
xcli  -u -c XIV1 ldap_config_get
```

Output:

```
TBD
```

Id	Name	Description	Default Position
name	Name		1
value	Value		2

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Enabling or Disabling LDAP-Based Authentication Mode

Enables/disables LDAP-based authentication mode.

```
ldap_mode_set  mode=<active|inactive>
```

Parameters:

Name	Type	Description	Mandatory	Default
mode	Boolean	Required state of LDAP authentication.	Y	

This command enables, or disables, LDAP-based authentication.

Example:

```
xcli  -u -c XIV1 ldap_mode_set mode=active
```

Output:

```
Command executed successfully.
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Warnings:

- ARE_YOU_SURE_YOU_WANT_TO_ENABLE_LDAP_AUTHENTICATION
  Are you sure you want to enable LDAP authentication?
- ARE_YOU_SURE_YOU_WANT_TO_DISABLE_LDAP_AUTHENTICATION
  Are you sure you want to disable LDAP authentication?

Completion Codes:

- LDAP_IS_NOT_FULLY_CONFIGURED
  LDAP is not fully configured

Troubleshooting: Check your settings.

- NO_LDAP_SERVERS_WITH_CERTIFICATE_ARE_DEFINED
  No LDAP servers with an LDAP certificate are defined in the system
- NO_LDAP_SERVERS_ARE_DEFINED
  No LDAP servers are defined in the system

Listing LDAP-Based Authentication Mode

Lists LDAP-based authentication mode.

```
ldap_mode_get
```

Parameters:

Name	Type	Description	Mandatory	Default

This command returns the authentication mode.

The command succeeds regardless of whether the LDAP server is inaccessible.

Example:

```
xcli  -u -c XIV1 ldap_mode_get
```

Output:

```
TBD
```

Id	Name	Description	Default Position
mode	Mode		1

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Adding an LDAP Server Definition

Adds an LDAP server definition.

```
ldap_add_server  fqdn=Fqdn address=IPaddress
  type=<SUN DIRECTORY|MICROSOFT ACTIVE DIRECTORY>
  [ certificate=PemCertificate ]
```

Parameters:

Name	Type	Description	Mandatory	Default
fqdn		FQDN of the LDAP server.	Y	
address		IP address of the LDAP server.	Y	
type	Enumeration	Type of the LDAP server.	Y	
certificate		The content of a `.pem` file, with asterisks (`*`) instead of newlines.	N	no certificate

This command adds an LDAP server to the system.

Example:

```
xcli -u -c XIV1 ldap_add_server fqdn=ldap.example.com address=1.2.3.4 type=SUN DIECTORY
```

Output:

```
Command executed successfully.
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- BAD_SSL_CERTIFICATE
  Cannot decipher SSL certificate
- MAX_LDAP_SERVERS_REACHED
  Maximum number of LDAP servers already defined
- SSL_CERTIFICATE_HAS_EXPIRED
  SSL certificate has expired on *Expiration Date*.

Removing an LDAP Server Definition

Removes an LDAP server definition.

```
ldap_remove_server  fqdn=Fqdn
```

Parameters:

Name	Type	Description	Mandatory	Default
fqdn		FQDN of the server to remove.	Y	

This command removes an LDAP server to the system.

Example:

```
xcli  -u -c XIV1 ldap_remove_server fqdn
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Warnings:

- ARE_YOU_SURE_YOU_WANT_TO_REMOVE_LDAP_SERVER
  Are you sure you want to remove LDAP server?

Completion Codes:

- LDAP_SERVER_NOT_FOUND
  LDAP server with specified FQDN is not defined in the system
- LDAP_IS_ACTIVE_BUT_THIS_IS_THE_LAST_SERVER
  Deleting the last LDAP server is illegal when LDAP authentication is active
- LDAP_USES_SSL_BUT_THIS_IS_THE_LAST_SERVER_WITH_CERTIFICATE
  Deleting the last LDAP server which has a valid SSL certificate is illegal when LDAP authentication is active and uses SSL

Listing LDAP Servers Defined in the System

Lists LDAP servers defined in the system.

```
ldap_list_servers  [ fqdn=Fqdn ]
```

Parameters:

Name	Type	Description	Mandatory	Default
fqdn		FQDN of a specific server to list.	N	All servers.

This command lists the LDAP servers defined in the system, along with their type description and whether they are mandatory.

Example:

```
xcli  -u -c XIV1 ldap_list_servers fqdn
```

Output:

```
FQDN               Address   Type            Has Certificate   Expiration Date
ldap.example.com   1.2.3.4   Sun Directory   no
```

Id	Name	Description	Default Position
fqdn	FQDN		1
address	Address		2
type	Type		3
has_certificate	Has Certificate		4
expiration_date	Expiration Date		5
valid_certificate	Valid Certificate		
accessible	Accessible		

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Testing User Credentials in LDAP Authentication

Tests user credentials in LDAP authentication.

```
ldap_user_test  [ fqdn=Fqdn ]
```

Parameters:

Name	Type	Description	Mandatory	Default
fqdn		FQDN of a specific LDAP server.	N	Current active LDAP server

This command authenticates the system's users via LDAP.

Example:

```
xcli  -u -c XIV1 ldap_user_test
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Allowed	
Read-only users	Allowed	
Technicians	Disallowed	

Completion Codes:

- LDAP_AUTHENTICATION_IS_NOT_ACTIVE
  LDAP authentication is not active
- LDAP_SERVER_NOT_DEFINED
  LDAP server *FQDN* is not defined in the system.
- USER_IS_PREDEFINED_IN_THE_SYSTEM
  User is predefined in the system

Adding a User to the SMIS Agent in the System

Adds a user to the SMIS agent in the system.

```
smis_add_user  user=UserName password=Password password_verify=Password
    [ current_password=Password ]
```

Parameters:

Name	Type	Description	Mandatory	Default
user	Object name	Name of the user.	Y	
password		Password for the user.	Y	
password_verify		Verification of the password. Its value must be identical to the value of `password`.	Y	
current_password		Current password of the user. Must be supplied if and only if the user is already defined in the SMIS agent.	N	None

Adds a user to the SMIS agent in the system. These users are distinct from the system users and do not affect them in any way. If the user already exists, this command can be used to change its password.

Example:

```
xcli  -u -c XIV1 smis_add_user user=newuser password=secret password_verify=secret
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- CURRENT_PASSWORD_NOT_SPECIFIED_FOR_SMIS_USER
  Current password must be specified if the SMIS user already defined.
- SMIS_USER_DOES_NOT_EXIST
  SMIS user does not exist and hence its current password cannot be specified.
- SMIS_CHANGE_FAILED
  SMIS configuration changed failed on *Num Failures* of *Num Modules*.

Listing Users Defined in the SMIS Agent in the System

Lists users defined in the SMIS agent in the system.

```
smis_list_users
```

Parameters:

Name	Type	Description	Mandatory	Default

This command lists users defined on the SMIS agent in the system. These users are distinct from the system users and do not affect them in any way.

Example:

```
xcli  -u -c XIV1 smis_list_users
```

Output:

```
Name
-----------
superuser
test
```

Id	Name	Description	Default Position
name	Name		1

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Removing a User from the SMIS Agent in the System

Removes a user from the SMIS agent in the system.

```
smis_remove_user  user=UserName
```

Parameters:

Name	Type	Description	Mandatory	Default
user	Object name	Name of the user to delete.	Y	

This command removes a user from the SMIS agent in the system. These users are distinct from the system users and do not affect them in any way. Note that the user `superuser` cannot be deleted.

Example:

```
xcli  -u -c XIV1 smis_remove_user user=olduser
```

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- SMIS_CHANGE_FAILED

  SMIS configuration changed failed on *Num Failures* of *Num Modules*.

Chapter 16. Fibre Channel and iSCSI Configuration and Status

The following sections describe the XIV Command Line Interface (XCLI) for fibre channel port configuration.

The sections are listed as follows:

- fc_port_list(Lists the status and configuration of the system's FC ports.)
- fc_port_config(Configures FC ports.)
- fc_port_reset(Resets an FC port.)
- host_connectivity_list(Lists FC and iSCSI-level connectivity to a pre-defined host.)
- fc_connectivity_list(Discovers FC hosts and targets on the FC network.)

Listing FC Ports

Lists the status and configuration of the system's FC ports.

```
fc_port_list [ module=ComponentId  | fcport=ComponentId ]
```

Parameters:

Name	Type	Description	Mandatory	Default
module		Limits the listing to a specific module.	N	All ports in all modules.
fcport		Lists only a specific port.	N	All ports in all modules.

This command lists all or some FC ports on the system. When no parameters are specified, all ports are listed. If a module is specified without a port, all ports on that module are listed. If a port is specified, a single port is listed.

The following information is provided for each port:

- Component ID of the module Port number (internal to module) 1-N
- WWPN
- Port ID
- Role (Initiator, Target, Dual)
- User-enabled (Yes/No)
- Maximum support rate: 1 GB, 2 GB, 4 GB, 10 GB; constant - function of the HBA's capability
- Configured rate: 1 GB, 2 GB, 4 GB, 10 GB, auto-negotiation; cannot be greater than the maximum supported rate
- Current active rate: 1 GB, 2 GB, 4 GB, 10 GB; equal to the configured rate, unless the configured rate is auto-negotiation
- Port state: Online, Offline, Loopback, Link Down (physical connection is on, but no logical connection)
- Error counts

- Link type: Fabric Direct Attach, Private Loop, Point-to-Point, Public Loop, Unknown

Id	Name	Description	Default Position
component_id	Component ID		1
status	Status		2
currently_functioning	Currently Functioning		3
port_num	Port Number		
wwpn	WWPN		4
port_id	Port ID		5
role	Role		6
user_enabled	User Enabled		7
max_supported_rate	Maximum Supported Rate (GBaud)		
configured_rate	Configured Rate (GBaud)		
current_rate	Current Rate (GBaud)		8
port_state	Port State		9
link_type	Link Type		10
error_count	Error Count		11
module	Module		
serial	Serial		
original_serial	Original Serial		
model	Model		
original_model	Original Model		

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Allowed	
Read-only users	Allowed	
Technicians	Allowed	

Changing FC Port Configuration

Configures FC ports.

```
fc_port_config  fc_port=ComponentId  [ enabled=<yes|no> ]
                                     [ role=<target|initiator> ]
                                     [ rate=<1|2|4|auto> ]
```

Parameters:

Name	Type	Description	Mandatory	Default
fc_port		Port identifier.	Y	

Name	Type	Description	Mandatory	Default
enabled	Boolean	User enabling/ disabling of the port.	N	yes
role	Enumeration	Port role: target, initiator or both.	N	Leaves the role unchanged.
rate	Enumeration	Line rate or auto for auto-negotiated rate.	N	Leaves the rate unchanged.

This command changes the configuration of a port.

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- COMPONENT_IS_NOT_AN_FC_PORT
  Component must specify an FC port
- FC_PORT_DOES_NOT_EXIST
  Specified FC port does not exist

Resetting FC Ports

Resets an FC port.

```
fc_port_reset  fc_port=ComponentId
```

```
fc_port_reset  fc_port=ComponentId  [ firmware_reset=<yes|no> ]
```

Parameters:

Name	Type	Description	Mandatory	Default
fc_port		FC port identifier.	Y	
firmware_reset	Boolean	Performs a firmware reset (resets all ports on the HBA).	N	

This command resets an FC port.

Access Control:

User Category	Permission	Condition
Storage administrator	Disallowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Allowed	

Completion Codes:

- COMPONENT_IS_NOT_AN_FC_PORT
  Component must specify an FC port
- FC_PORT_DOES_NOT_EXIST
  Specified FC port does not exist

Listing Connectivity to Hosts

Lists FC and iSCSI-level connectivity to a pre-defined host.

```
host_connectivity_list [host=HostName | fc_host_port=WWPN]
[module=ComponentId | fcport=ComponentId]
```

Parameters:

Name	Type	Description	Mandatory	Default
host	Object name	Limits viewing to the ports of a specific host.	N	All hosts.
fc_host_port		Limits viewing to this specific port.	N	All ports
module		Limits output to only enable connectivity to this module.	N	All modules
fcport		Limits output to this specific XIV port.	N	All ports

This command shows the connectivity status between a XIV port and a defined host. The output can be limited to a specific port, module or XIV port. Hosts can attach to the FC and iSCSI either directly (point-to-point), via an FC fabric or via a Gigabit Ethernet switch. Connectivity refers to both physical connectivity and SCSI log in. Each output line contains the following information:

- Host (name)
- Host port (WWPN)
- XIV module (rack_id:module_id)
- XIV port number (within the module)
- Boolean-connected Y/N (login)

Id	Name	Description	Default Position
host	Host		1
host_port	Host Port		2
module	Module		3
local_fc_port	Local FC port		4
type	Type		5

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Allowed	
Read-only users	Allowed	
Technicians	Allowed	

Discovering FC Hosts

Discovers FC hosts and targets on the FC network.

```
fc_connectivity_list  [ role=<dual|initiator|target> ] [ wwpn=WWPN ]
                        [ module=ComponentId | fc_port=ComponentId ]
                        [ logged_in=<yes|no> ]
```

Parameters:

Name	Type	Description	Mandatory	Default
role	Enumeration	Specifies whether the discovery is for initiators or targets.	N	list all - targets and/or initiators.
wwpn		Limits the output only to this specific address.	N	All addresses
module		Limits the output to only enable connectivity to this module.	N	All modules
fc_port		Limits the output to this specific XIV port.	N	All ports
logged_in	Boolean	Includes FC hosts that are not logged in.	N	All

This command lists FC hosts on the network.

type=initiator detects initiators on the network. When type=initiator, the non-logged-in option can only be used to debug hosts that are on the network, but that did not log in.

type=target detects targets. When type=target, the non-logged-in option can only be used to debug targets that rejected the XIV login. This command creates an error for a target-only port to list targets, or for an initiator-only port to list initiators. Each output line contains the following information:

- Component ID (of the module)
- XIV port number (within the module)
- WWPN
- Port ID (can be correlated with the switch database)
- Role: Initiator, Target, Dual
- Initiator/target (is the same for all lines of the same command)
- Login status (Yes/No)

Id	Name	Description	Default Position
component_id	Component ID		1
wwpn	WWPN		2
port_id	Port ID		3
role	Role		4
logged_in	Logged In		5

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Allowed	
Read-only users	Allowed	
Technicians	Allowed	

Chapter 17. Hardware Maintenance

The following sections describe the XIV Command Line Interface (XCLI) for maintaining hardware components.

The sections are listed as follows:

- component_list(Lists system components and their status.)
- component_phaseout(Phases out a component.)
- component_phasein(Phases in a hardware component.)
- component_test(Tests a hardware component.)
- component_equip(Marks a hardware component as one that was installed.)
- component_connectivity_list(Lists the status of all internal cables.)
- monitor_redist(Monitors the status of a rebuild or redistribute process.)
- disk_list(Lists special disk statuses.)
- module_list(Lists the configuration of all or of the specified modules.)
- ethernet_cable_list(Lists ethernet cables in the system.)
- module_probe(Probes to determine whether a module failed due to a hardware or software problem.)
- ups_list(Lists the status of UPS components.)
- service_list(Lists all the service specific statuses.)
- switch_list(Lists the special statuses of the internal switches.)
- psu_list(Lists the PSUs in the XIV system.)
- traces_stop(Stops system traces.)
- traces_resume(Resumes system traces.)
- traces_status_list(Lists status of system traces.)
- system_logs_send_file(Sends a system logs file as an email attachment.Sends a system logs file as an email attachment.)
- technician_work_in_progress(Notifies the system of a technician at work.)

Listing System Components

Lists system components and their status.

```
component_list [ component=ComponentId ] [ filter=<ALL|FAILED|NOTOK> ]
```

Parameters:

Name	Type	Description	Mandatory	Default
component		Lists only this component.	N	All components.
filter	Enumeration	Filters the list to show only failed or only non-OK components.	N	ALL

Lists system components. The list can be filtered to show only a specific component, all failed components or all components in a non-OK state.

For status and configuration of specific component types, refer to the List commands for specific components, such as: module_list, disk_list, ups_list, switch_list. The output is a list of components, with the following information for each component:

- Component identification
- Component general status
- Indication about whether the component is currently functioning

Id	Name	Description	Default Position
component_id	Component ID		1
status	Status		2
currently_functioning	Currently Functioning		3

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Allowed	
Read-only users	Allowed	
Technicians	Allowed	

Phasing Out a Component

Phases out a component.

```
component_phaseout component=ComponentId [ markasfailed=<yes|no> ]
```

Parameters:

Name	Type	Description	Mandatory	Default
component		Component identification.	Y	
markasfailed	Boolean	Marks the component as failed after phase-out.	N	yes

This command instructs the system to stop using the component, where the component can be either a disk, module, switch or UPS.

For disks, the system starts a process for copying the disk's data, so that even without this disk, the system is redundant. The state of the disk after the command is *Phasing-out*.

The same process applies for data modules. The system starts a process for copying all the data in the module, so that the system is redundant even without this module. A data module phase-out causes a phase-out for all the disks in that module.

For UPSs and switches, the system configures itself to work without the component. There is no phase-out for power supplies, SFPs or batteries.

Phasing out a module or a disk, if it results in the system becoming non-redundant, is not permitted. Components must be in either OK or a Phase In status.

Once the phase-out process is completed, the component's state is either Fail or Ready, depending on the argument *markasfailed*. If true, the phased-out component is marked as a failed component (in order to replace the component). If false, the phased-out component is in the Ready state.

Access Control:

User Category	Permission	Condition
Storage administrator	Disallowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Allowed	

Completion Codes:

- PHASEOUT_NOT_SUPPORTED_FOR_THIS_COMPONENT_TYPE
  This component type cannot be phased-out
- PHASEOUT_WILL_MAKE_SYSTEM_NON_REDUNDANT
  Cannot phase out the component because it will cause data to be unprotected
- PHASEOUT_NOT_ALLOWED_IN_CURRENT_STATUS
  Component cannot be phased-out in its current status
- COMPONENT_DOES_NOT_EXIST
  Component does not exist
- COMPONENT_TYPE_CANNOT_BE_PHASED_OUT_AS_FAILED
  Components of this type cannot be phased-out and marked as failed
- MODULE_CANNOT_BE_PHASED_OUT_DUE_TO_MANAGEMENT_REQUIREMENT
  Module cannot be phased out due to management requirement
  Troubleshooting: Contact support
- CAN_NOT_PHASE_OUT_DISK_WITH_MARKASFAILED_NO
  Disks cannot be phased-out with markasfailed=no
- COMPONENT_TYPE_MUST_BE_PHASED_OUT_AS_FAILED
  Components of this type must be phased-out as failed
- USE_SERVICE_PHASEOUT_COMMAND
  Command component_phaseout does not support services. Please use service_phaseout.
- CONTAINING_COMPONENT_IN_WRONG_STATUS
  Operation not allowed in current status of containing component.

- SUBCOMPONENT_IN_WRONG_STATUS
  Operation not allowed in current status of a subcomponent.
- CONTROLLING_SERVICES_NOT_ALLOWED_FOR_USER_CATEGORY
  Controlling services not allowed for user category

Phasing In a Component

Phases in a hardware component.

```
component_phasein component=ComponentId
```

Parameters:

Name	Type	Description	Mandatory	Default
component		Component to be phased in.	Y	

This command instructs the system to phase in a component. Components are used by the system immediately. For disk and data modules, a process for copying data to the components (redistribution) begins. Components must be in Ready or Phasing Out states. There is no phase-in for power supplies, SFPs or batteries.

Access Control:

User Category	Permission	Condition
Storage administrator	Disallowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Allowed	

Warnings:

- WARNING_COMPONENT_IS_PHASING_OUT
  Component is being phased out. Are you sure you want to phase it in?

Completion Codes:

- PHASEIN_NOT_SUPPORTED_FOR_THIS_COMPONENT_TYPE
  This component type cannot be phased-in
- PHASEIN_NOT_ALLOWED_IN_CURRENT_STATUS
  Component cannot be phased-in in its current status
- COMPONENT_DOES_NOT_EXIST
  Component does not exist
- USE_SERVICE_PHASEIN_COMMAND
  Command component_phasein does not support services. Please use service_phasein.
- CONTAINING_COMPONENT_IN_WRONG_STATUS
  Operation not allowed in current status of containing component.
- SUBCOMPONENT_IN_WRONG_STATUS
  Operation not allowed in current status of a subcomponent.

- CONTROLLING_SERVICES_NOT_ALLOWED_FOR_USER_CATEGORY
  Controlling services not allowed for user category
- SERVICE_CANNOT_BE_PHASED_IN
  Service cannot be phased in because its interface services cannot be activated.
  Troubleshooting: Check system requirements for activating interface services.

Testing a Component

Tests a hardware component.

```
component_test component=ComponentId
```

Parameters:

Name	Type	Description	Mandatory	Default
component		Component ID.	Y	

This command instructs the XIV system to test the component. The command is used after a failed component is replaced. Components must be in a Failed status. Upon a successful test, the component changes to Ready status. Upon a failed test, the component remains in a Failed state.

Access Control:

User Category	Permission	Condition
Storage administrator	Disallowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Allowed	

Completion Codes:

- COMPONENT_DOES_NOT_EXIST
  Component does not exist
- TEST_NOT_SUPPORTED_FOR_THIS_COMPONENT_TYPE
  This component type cannot be tested
- DISK_IS_TOO_SMALL
  Disk capacity is smaller than the current system minimum.
  Troubleshooting: Replace disk
- TEST_NOT_ALLOWED_IN_CURRENT_STATUS
  Component cannot be tested in its current status
- CONTAINING_COMPONENT_IN_WRONG_STATUS
  Operation not allowed in current status of containing component.

Setting a Component as Equipped

Marks a hardware component as one that was installed.

```
component_equip component=ComponentId
```

Parameters:

Name	Type	Description	Mandatory	Default
component		Component to be equipped.	Y	

This command configures the system to start using the component, assuming that it was assembled. The component is tested by the system. After completing the test, the component is marked as either Ready or Failed.

Access Control:

User Category	Permission	Condition
Storage administrator	Disallowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Allowed	

Completion Codes:

- EQUIP_NOT_SUPPORTED_FOR_THIS_COMPONENT_TYPE
  This component type cannot be equipped
- **Target event not found. href = COMPONENT_ALREADY_EQUIPPED**
- EQUIP_NOT_ALLOWED_IN_CURRENT_STATUS
  Component already equipped
- COMPONENT_DOES_NOT_EXIST
  Component does not exist

Listing Internal Cabling Status

Lists the status of all internal cables.

```
component_connectivity_list [ filter=<all|failed> ]
```

Parameters:

Name	Type	Description	Mandatory	Default
filter	Enumeration	Sets the filtering of information.	N	all

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Monitoring Rebuild or Redistribution Processes

Monitors the status of a rebuild or redistribute process.

```
monitor_redist
```

This command outputs the current rebuild or redistribution process. This command may show that no such process exists.

If such a process exists, the following information is shown:

- Type (adding new capacity, replacing failed component, phase-out, rebuild after failure)
- Initial capacity to copy
- Time started
- Capacity remaining to copy
- Time elapsed
- Percent completed
- Estimated time to completion

Id	Name	Description	Default Position
type	Type		1
initial_capacity_to_copy	Initial Capacity to Copy (GB)		2
capacity_remaining_to_copy	Capacity Remaining to Copy (GB)		3
percent_done	%done		4
time_started	Time Started		5
estimated_time_to_finish	Estimated Time to Finish		6
time_elapsed	Time Elapsed		7

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Allowed	
Read-only users	Allowed	
Technicians	Allowed	

Listing Disk Status

Lists special disk statuses.

```
disk_list [ disk=ComponentId ]
```

Parameters:

Name	Type	Description	Mandatory	Default
disk		Disk for which special statuses are to be listed.	N	All disks

This command lists the statuses of the disk, including the following:

- Component generic status
- Disk capacity

Id	Name	Description	Default Position
component_id	Component ID		1
status	Status		2
currently_functioning	Currently Functioning		3
capacity	Capacity (GB)		4
target_status	Target Status		5
model	Model		6
size	Size		7
serial	Serial		8
original_model	Original Model		
original_serial	Original Serial		

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Allowed	
Read-only users	Allowed	
Technicians	Allowed	

Listing Module Configuration

Lists the configuration of all or of the specified modules.

```
module_list [ module=ComponentId ]
```

Parameters:

Name	Type	Description	Mandatory	Default
module		Lists the configuration of the specified module.	N	All modules

This command lists the following information for each module:

- Generic component status
- Module type
- Number of disks
- Number of FC ports
- Number of Ethernet ports for iSCSI

Additional information is available through running `module_list -t all` :

- Serial
- Original Serial
- Part Number
- Original Part Number

Id	Name	Description	Default Position
component_id	Component ID		1
status	Status		2
currently_functioning	Currently Functioning		3
target_status	Target Status		4
type	Type		5
disk_bay_count	Data Disks		6
fc_port_count	FC Ports		7
ethernet_port_count	iSCSI Ports		8
serial	Serial		
original_serial	Original Serial		
part_number	Part Number		
original_part_number	Original Part Number		

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Allowed	
Read-only users	Allowed	
Technicians	Allowed	

Listing Ethernet Cables in the System

Lists ethernet cables in the system.

```
ethernet_cable_list  [ ethernet_cable=ComponentId ]
```

Parameters:

Name	Type	Description	Mandatory	Default
ethernet_cable		Lists the status only for the specified ethernet cable.	N	All ethernet cables

Lists ethernet cables in the system.

Example:

```
xcli  -u -c XIV1 ethernet_cable_list
```

Output:

```
Component ID            Status   Currently Functioning   Connected to
----------------------- -------- ----------------------- ---------------
1:Ethernet_Cable:1:1    OK       yes                     1:Switch:1:1
1:Ethernet_Cable:1:2    OK       yes                     1:Switch:2:1
1:Ethernet_Cable:2:1    OK       yes                     1:Switch:1:2
1:Ethernet_Cable:2:2    OK       yes                     1:Switch:2:2
1:Ethernet_Cable:4:1    OK       yes                     1:Switch:1:4

Should be connected to   Link status   Interface Role
--------- ------------- ----------------
1:Switch:1:1             UP            Internal
1:Switch:2:1             UP            Internal
1:Switch:1:2             UP            Internal
1:Switch:2:2             UP            Internal
1:Switch:1:4             UP            Internal
```

Id	Name	Description	Default Position
component_id	Component ID		1
status	Status		2
currently_functioning	Currently Functioning		3
link_status	Link status		4
interface_role	Interface Role		5

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Allowed	
Read-only users	Allowed	
Technicians	Allowed	

Probing a Failed Module

Probes to determine whether a module failed due to a hardware or software problem.

```
module_probe module=ComponentId
```

Parameters:

Name	Type	Description	Mandatory	Default
module		Module to be probed.	Y	

This command probes a failed module. The XIV system's analysis determines whether the module failed because of a hardware failure or a software bug. This command can only be used for failed modules.

Access Control:

User Category	Permission	Condition
Storage administrator	Disallowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Allowed	

Completion Codes:

- MODULE_IS_NOT_IN_FAILED_STATE
  The module is not in failed state and therefore was not probed
- COMPONENT_DOES_NOT_EXIST
  Component does not exist

Listing UPS Component Statuses

Lists the status of UPS components.

```
ups_list [ ups=ComponentId ]
```

Parameters:

Name	Type	Description	Mandatory	Default
ups		Lists the status only for the specified UPS.	N	All UPS systems.

This command lists the status of the components of UPSs. The following information is provided:

- Generic status

- Input power on: Y/N
- Battery charge level
- Last date of self-test
- Result of last self-test
- Is monitoring enabled
- Last calibration date
- result of last calibration
- Status of UPS
- Date of next self test
- Serial Number
- Load level percent
- Apparent load level percent

Additional information, available through running `ups_list -t all`, includes:

- Last Calibration Date
- Last Calibration Result
- Next Self Test
- Serial
- Original Serial
- Load % Watts
- Apparent Load % VA
- Minutes Left
- Temperature
- AOS Version
- Self-test Status
- Component Test Status
- Battery Date
- UPS Manufacture Date

Id	Name	Description	Default Position
component_id	Component ID		1
status	Status		2
currently_functioning	Currently Functioning		3
input_power_on	Input Power On		4
battery_charge_level	Battery Charge Level		5
last_self_test_date	Last Self Test Date		6
last_self_test_result	Last Self Test Result		7
is_enabled	Monitoring Enabled		8
ups_status	UPS Status		9
last_calibration_date	Last Calibration Date		
last_calibration_result	Last Calibration Result		
next_scheduled_self_test	Next Self Test		
serial	Serial		

Id	Name	Description	Default Position
original_serial	Original Serial		
load_power_percent	Load % Watts		
apparent_load_power_percent	Apparent Load % VA		
runtime_remaining	Minutes left		
internal_temperature	Temperature		
aos_version	AOS Version		
self_test_status	Self-Test Status		
component_test_status	Component Test Status		
battery_date	Battery Date		
manufacture_date	UPS Manufacture Date		

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Allowed	
Read-only users	Allowed	
Technicians	Allowed	

Listing Service Status

Lists all the service specific statuses.

```
service_list [ service=ComponentId ]
```

Parameters:

Name	Type	Description	Mandatory	Default
service		Service to be listed.	N	All services

This command lists the statuses that apply to services. The list includes the following information:

- Component generic status
- Service on/failed
- Comment (optional)

Id	Name	Description	Default Position
component_id	Component ID		1
status	Status		2
currently_functioning	Currently Functioning		3
target_status	Target Status		4

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Allowed	
Read-only users	Allowed	
Technicians	Allowed	

Listing Switch Statuses

Lists the special statuses of the internal switches.

```
switch_list [ switch=ComponentId ]
```

Parameters:

Name	Type	Description	Mandatory	Default
switch		Shows the status for the specified switch only.	N	All switches

This command shows the status of the switch, including the following information as default:

- Generic component status
- AC power status
- DC power status
- Status of interconnect
- Number of failed fans
- Serial number
- Temperature

Additional information, available through running `switch_list -t all` includes:

- Serial
- Original Serial
- Temperature
- Temperature Status
- Current Active Version
- Next Active Version

Id	Name	Description	Default Position
component_id	Component ID		1
status	Status		2
currently_functioning	Currently Functioning		3
ac_power_state	AC Power State		4
dc_power_state	DC Power State		5

Id	Name	Description	Default Position
interconnect	Interconnect		6
failed_fans	Failed Fans		7
serial	Serial		
original_serial	Original Serial		
temperature	Temperature		
temperature_status	Temperature Status		
current_active_version	Current Active version		
next_active_version	Next Active version		

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Allowed	
Read-only users	Allowed	
Technicians	Allowed	

Listing PSUs in the System

Lists the PSUs in the XIV system.

```
psu_list
```

This command lists the PSUs in the XIV system.

Example:

```
xcli  -u  -c Nextra1 psu_list
```

Output:

```
Component ID   Status   Currently Functioning   Hardware Status
-------------- -------- ----------------------- -----------------
1:PSU:13:1     Failed   no                      Cable Failure
1:PSU:13:2     OK       yes                     OK
1:PSU:14:1     Failed   no                      Cable Failure
1:PSU:14:2     OK       yes                     OK
1:PSU:15:1     Failed   no                      Cable Failure
1:PSU:15:2     OK       yes                     OK
1:PSU:6:1      Failed   no                      Cable Failure
1:PSU:6:2      OK       yes                     OK
1:PSU:9:1      Failed   no                      Cable Failure
1:PSU:9:2      OK       yes                     OK
```

Id	Name	Description	Default Position
component_id	Component ID		1
status	Status		2

Id	Name	Description	Default Position
currently_functioning	Currently Functioning		3
hardware_status	Hardware Status		4

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Allowed	
Read-only users	Allowed	
Technicians	Allowed	

Stopping System Traces

Stops system traces.

```
traces_stop
```

Stops system traces.

Id	Name	Description	Default Position
module	Module		1
status	Status		2

Example:

```
xcli  -u -c Nextra1 traces_stop
```

Output:

```
Module        Status
------------- ---------
1:Module:1    Stopped
1:Module:2    Stopped
1:Module:3    Stopped
1:Module:4    Stopped
1:Module:5    Stopped
1:Module:6    Stopped
1:Module:7    Stopped
1:Module:8    Stopped
1:Module:9    Stopped
1:Module:10   Stopped
1:Module:11   Stopped
1:Module:12   Stopped
1:Module:13   Stopped
1:Module:14   Stopped
1:Module:15   Stopped
```

Access Control:

User Category	Permission	Condition
Storage administrator	Disallowed	

User Category	Permission	Condition
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Allowed	

Resuming System Traces

Resumes system traces.

```
traces_resume
```

Resumes system traces.

Id	Name	Description	Default Position
module	Module		1
status	Status		2

Example:

```
xcli  -u -c Nextra1 traces_resume
```

Output:

```
Module        Status
------------- ---------
1:Module:1    Running
1:Module:2    Running
1:Module:3    Running
1:Module:4    Running
1:Module:5    Running
1:Module:6    Running
1:Module:7    Running
1:Module:8    Running
1:Module:9    Running
1:Module:10   Running
1:Module:11   Running
1:Module:12   Running
1:Module:13   Running
1:Module:14   Running
1:Module:15   Running
```

Access Control:

User Category	Permission	Condition
Storage administrator	Disallowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Allowed	

Listing Status of System Traces

Lists status of system traces.

```
traces_status_list
```

Lists status of system traces.

Id	Name	Description	Default Position
module	Module		1
status	Status		2

Example:

```
xcli  -u -c Nextra1 traces_status_list
```

Output:

```
Module        Status
------------- ---------
1:Module:1    Running
1:Module:2    Running
1:Module:3    Stopped
1:Module:4    Running
1:Module:5    Running
1:Module:6    Running
1:Module:7    Running
1:Module:8    Running
1:Module:9    Running
1:Module:10   Running
1:Module:11   Running
1:Module:12   Running
1:Module:13   Running
1:Module:14   Running
1:Module:15   Running
```

Access Control:

User Category	Permission	Condition
Storage administrator	Disallowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Allowed	

Sending a System Logs File as an Email Attachment

Sends a system logs file as an email attachment.

```
system_logs_send_file
file=FileName
email_address=email
smtpgw=Address
[port=PortNumber] [sender_address=<SenderEmailAddress|DEFAULT>] [subject=SubjectFormat]
```

Parameters:

Name	Type	Description	Mandatory	Default
smtpgw		SMTP gateway address (IP or DNS name).	Y	
sender_address		Email address of the sender of the email. If not defined, the value email_sender_address is used (see Printing Configuration Parameters). If neither is defined, the command fails.	N	DEFAULT
file	String	Name of the file.	Y	
email_address		Email address to send the email to.	Y	
port	Integer	TCP port used in the gateway instead of the default port 25.	N	25
subject	Token String	Controls the formatting of the Email subject line. The tags {system_name} {machine_model} {machine_serial_number} {file} and {sender_address} can be used.	N	System Logs file {file} from {machine_model}-{machine_serial_number}

Sends a system logs file as an email attachment.

Example:

```
xcli  -u -c XIV1 system_logs_send_file file
email_address smtpgw port sender_address subject
```

Access Control:

User Category	Permission	Condition
Storage administrator	Disallowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Allowed	

Completion Codes:

- CANNOT_GET_SYSTEM_LOGS_COLLECTION_STATUS
  Cannot get status of collection of system logs
  Troubleshooting: Contact support
- CANNOT_READ_FROM_FILE
  Cannot read from file *'Filename'*
  Troubleshooting: Contact support
- SENDER_ADDRESS_NOT_DEFINED
  Sender address is not given and the default From Address is defined

Notifying the System of a Technician at Work

Notifies the system of a technician at work.

```
technician_work_in_progress
[ mode=<start|end|get> ] [ timeout=<[hh:]mm> ] [ comment=Comment ]
```

Parameters:

Name	Type	Description	Mandatory	Default
mode	Enumeration	Mode of the command. `start` resets the timeout. `end` notifies the system that the technician has finished their work. `get` only returns the time elapsed since the last resetting of the timeout (or 0 if a technician is not currently at work).	N	get
timeout		Timeout in either hh:mm format or a number of minutes. The timeout cannot exceed 23 hours and 59 minutes. Must be specified if `mode` is `start` and cannot be specified otherwise.	N	

Name	Type	Description	Mandatory	Default
comment	String	Comment to add to the events that pertain to the command. Must be specified if `mode` is `start` and cannot be specified otherwise.	N	none

Notifies the system of a technician at work.

Example:

```
xcli  -u -c XIV1 technician_work_in_progress mode=start timeout=1:30
```

Output:

```
Command executed successfully.
```

Access Control:

User Category	Permission	Condition
Storage administrator	Disallowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Allowed	

Completion Codes:

- TECHNICIAN_WORK_TIMEOUT_CANNOT_BE_SPECIFIED
  Timeout can only be specified when technician work starts.
- TECHNICIAN_WORK_COMMENT_CANNOT_BE_SPECIFIED
  Comment can only be specified when technician work starts.
- TECHNICIAN_WORK_TIMEOUT_NOT_SPECIFIED
  Timeout must be specified when technician work starts.
- TECHNICIAN_WORK_COMMENT_NOT_SPECIFIED
  Comment must be specified when technician work starts.

Chapter 18. Statistics

The following sections describe the XIV Command Line Interface (XCLI) for getting system statistics.

The sections are listed as follows:

- statistics_get(Retrieves performance statistics from the XIV system.)
- usage_get(Shows the usage history of a volume or a Storage Pool.)

Getting Performance Statistics

Retrieves performance statistics from the XIV system.

```
statistics_get [host=H |
                host_iscsi_name=initiatorName | host_fc_port=WWPN |
                target=RemoteTarget | remote_fc_port=WWPN |
      remote_ipaddress=IPAdress | vol=VolName |
       ipinterface=IPInterfaceName | local_fc_port=ComponentId ]
                < start=TimeStamp | end=TimeStamp >
      [module=ComponentId]
                  count=N
                  interval=IntervalSize
                  resolution_unit=<minute|hour|day|week|month>
```

Parameters:

Name	Type	Description	Mandatory	Default
host	Object name	Limits statistics to the specific host only.	N	All hosts
host_fc_port		FC address of the host port.	N	All ports.
target	Object name	Limits statistics to I/O generated by the specified remote target only (due to remote mirroring).	N	All targets.
remote_fc_port		Limits statistics to the specified host/remote FC port only.	N	All ports.
remote_ipaddress		IP address of the remote target port.	N	All ports.
host_iscsi_name	iSCSI initiator name	Limits statistics to the specified iSCSI initiator only.	N	All ports.

Name	Type	Description	Mandatory	Default
ipinterface	Object name	Limits statistics to the specified IP interface (relevant for iSCSI only).	N	All interfaces.
module		Limits statistics to the specified module only.	N	All modules
local_fc_port		Limits statistics to I/O performed on the specified FC port only.	N	All ports
vol	Object name	Limits statistics to the specified volume only.	N	All volumes
start		Starting point for the statistics report.	N	
end		Ending point for the statistics report.	N	
count	Positive integer	Number of time points reported.	Y	
interval	Positive integer	The length of time in each statistic's time point. The resolution of this number is set in *resolution_unit*.	Y	
resolution_unit	Enumeration	Sets the unit of measurement for the length of each bin.	Y	

This command lists I/O statistics. The *count* parameter sets the number of lines in the statistics report. Together, the *interval* and *resolution_unit* set the length of time for each statistics line. Either start or end timestamps must be provided. These timestamps set the time for the statistics report. Other parameters restrict statistics to a specific host, host port, volume, interface port and so on.

For each line of statistics, 48 numbers are reported, which represent all the combinations of reads/writes, hits/misses and I/O size reporting for each of the 16 options for bandwidth, IOPS and latency.

The syntax for the `start` and `end` fields is as follows: `Y-M-D[.[h[:m[:s]]]]`, where the ranges are as follows:

- Y - year (four digit)
- M - month (1-12)
- D - day (1-31)
- h - hour (0-23, with 0 as default)

- m - minute (0-59, with 0 as default)
- s - second (0-59, with 0 as default)

Note:

The year, month and day are separated by dashes, and the optional hours, minutes and seconds are separated by colons.

Id	Name	Description	Default Position
time	Time		1
failures	Failures		
aborts	Aborts		
read_hit_very_large_iops	Read Hit Very large - IOps		2
read_hit_very_large_latency	Read Hit Very large - Latency		3
read_hit_very_large_throughput	Read Hit Very large - Throughput		4
read_hit_large_iops	Read Hit Large - IOps		5
read_hit_large_latency	Read Hit Large - Latency		6
read_hit_large_throughput	Read Hit Large - Throughput		7
read_hit_medium_iops	Read Hit Medium - IOps		8
read_hit_medium_latency	Read Hit Medium - Latency		9
read_hit_medium_throughput	Read Hit Medium - Throughput		10
read_hit_small_iops	Read Hit Small - IOps		11
read_hit_small_latency	Read Hit Small - Latency		12
read_hit_small_throughput	Read Hit Small - Throughput		13
read_miss_very_large_iops	Read Miss Very large - IOps		14
read_miss_very_large_latency	Read Miss Very large - Latency		15
read_miss_very_large_throughput	Read Miss Very large - Throughput		16
read_miss_large_iops	Read Miss Large - IOps		17
read_miss_large_latency	Read Miss Large - Latency		18
read_miss_large_throughput	Read Miss Large - Throughput		19
read_miss_medium_iops	Read Miss Medium - IOps		20

Id	Name	Description	Default Position
read_miss_medium_latency	Read Miss Medium - Latency		21
read_miss_medium_throughput	Read Miss Medium - Throughput		22
read_miss_small_iops	Read Miss Small - IOps		23
read_miss_small_latency	Read Miss Small - Latency		24
read_miss_small_throughput	Read Miss Small - Throughput		25
write_hit_very_large_iops	Write Hit Very large - IOps		26
write_hit_very_large_latency	Write Hit Very large - Latency		27
write_hit_very_large_throughput	Write Hit Very large - Throughput		28
write_hit_large_iops	Write Hit Large - IOps		29
write_hit_large_latency	Write Hit Large - Latency		30
write_hit_large_throughput	Write Hit Large - Throughput		31
write_hit_medium_iops	Write Hit Medium - IOps		32
write_hit_medium_latency	Write Hit Medium - Latency		33
write_hit_medium_throughput	Write Hit Medium - Throughput		34
write_hit_small_iops	Write Hit Small - IOps		35
write_hit_small_latency	Write Hit Small - Latency		36
write_hit_small_throughput	Write Hit Small - Throughput		37
write_miss_very_large_iops	Write Miss Very large - IOps		38
write_miss_very_large_latency	Write Miss Very large - Latency		39
write_miss_very_large_throughput	Write Miss Very large - Throughput		40
write_miss_large_iops	Write Miss Large - IOps		41
write_miss_large_latency	Write Miss Large - Latency		42
write_miss_large_throughput	Write Miss Large - Throughput		43
write_miss_medium_iops	Write Miss Medium - IOps		44

Id	Name	Description	Default Position
write_miss_medium_latency	Write Miss Medium - Latency		45
write_miss_medium_throughput	Write Miss Medium - Throughput		46
write_miss_small_iops	Write Miss Small - IOps		47
write_miss_small_latency	Write Miss Small - Latency		48
write_miss_small_throughput	Write Miss Small - Throughput		49
time_in_seconds	Time (s)		50

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Allowed	
Read-only users	Allowed	
Technicians	Disallowed	

Completion Codes:

- BAD_TIME_FORMAT
  Bad time format. Should be YYYY-MM-DD[.HH[:MM[:SS]]]
- TARGET_PORT_BAD_ADDRESS
  Remote port address is illegal or does not belong to the remote target
- VOLUME_BAD_NAME
  Volume name does not exist
- STATS_TOO_MANY_SAMPLES
  Requested number of statistics samples is too high
- TARGET_BAD_NAME
  Target name does not exist
- COMPONENT_DOES_NOT_EXIST
  Component does not exist
- HOST_BAD_NAME
  Host name does not exist
- HOST_PORT_DOES_NOT_EXIST
  Port ID is not defined
- IPINTERFACE_DOES_NOT_EXIST
  IP Interface name does not exist

Retrieving History Usage

Shows the usage history of a volume or a Storage Pool.

```
usage_get < vol=VolName | pool=PoolName    > [ start=StartTime | start_in_seconds=StartTime ]
          [ end=EndTime ] [ max=MaxEntries ]
```

Parameters:

Name	Type	Description	Mandatory	Default
vol	Object name	Volume for which usage statistics are retrieved.	N	
pool	Object name	Storage Pool for which usage statistics are retrieved.	N	
start		Starting time for usage history retrieval.	N	Creation time of the object.
end		Ending time for usage history retrieval.	N	Current time.
max	Integer	Maximum number of entries to retrieve.	N	No limit.
start_in_seconds	Integer	Starting time for usage history retrieval, in seconds since 12:00:00 AM, 1 January 1970.	N	Creation time of the object.

This command retrieves the usage history of a Storage Pool or volume, in units of one megabyte.

Id	Name	Description	Default Position
time	Time		1
volume_usage	Volume Usage (MB)		2
snapshot_usage	Snapshot Usage (MB)		3

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- VOLUME_BAD_NAME
  Volume name does not exist
- POOL_DOES_NOT_EXIST
  Storage Pool does not exist
- BAD_TIME_FORMAT
  Bad time format. Should be YYYY-MM-DD[.HH[:MM[:SS]]]
- END_BEFORE_START
  End Time should be later than Start Time
- VOLUME_IS_SNAPSHOT
  Operation is not permitted on snapshots
  Operation is not permitted on snapshots

Chapter 19. Meta-data

The following sections describe the XIV Command Line Interface (XCLI) for meta-data handling.

The sections are listed as follows:

- metadata_set(Sets a meta-data of an object.)
- metadata_delete(Deletes a meta-data of an object.)
- metadata_list(Lists a meta-data of an object.)

Setting Meta Data

Sets a meta-data of an object.

```
metadata_set
object_type=<cg|cluster|dest|destgroup|host|pool|rule|schedule|smsgw|smtpgw|target|user|user_group|vol>
name=Name key=Key value=Value
```

Parameters:

Name	Type	Description	Mandatory	Default
object_type	Enumeration	Type of object.	Y	
name	Object name	Name of object	Y	
key	String	Metadata key.	Y	
value	String	Metadata value	Y	

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Conditionally Allowed	Metadata can be set for only volumes, snapshots, snapshot groups, clusters or hosts, and only for objects associated with the application administrator executing the command. Hosts or clusters should be associated with the user. Volumes should be mapped to a host or a cluster associated with the user. Snapshots or snapshot groups should be ones created by application administrator.
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- **Target event not found. href = OBJECT_BAD_NAME**
- **Target event not found. href = MAX_METADATA_OBJECTS_REACHED**

Deleting Meta Data

Deletes a meta-data of an object.

```
metadata_delete object_type=<cg|cluster|dest|destgroup|host|pool|rule|schedule|smsgw|smtpgw|target|user|user_gro
                        name=Name key=Key
```

Parameters:

Name	Type	Description	Mandatory	Default
object_type	Enumeration	Type of object.	Y	
name	Object name	Name of object	Y	
key	String	Metadata key.	Y	

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Conditionally Allowed	Metadata can be set for only volumes, snapshots, snapshot groups, clusters or hosts, and only for objects associated with the application administrator executing the command. Hosts or clusters should be associated with the user. Volumes should be mapped to a host or a cluster associated with the user. Snapshots or snapshot groups should be ones created by application administrator.
Read-only users	Disallowed	
Technicians	Disallowed	

Completion Codes:

- **Target event not found. href = OBJECT_BAD_NAME**
- **Target event not found. href = METADATA_OBJECT_KEY_NOT_FOUND**

Listing Meta Data

Lists a meta-data of an object.

```
metadata_list [ object_type=<cg|cluster|dest|destgroup|host|pool|rule|schedule|smsgw|smtpgw|target|user|user_gro
              [name=Name][key=Key]
```

Parameters:

Name	Type	Description	Mandatory	Default
object_type	Enumeration	Type of object.	N	All object types
name	Object name	Name of object	N	All objects
key	String	Metadata key.	N	List all keys and values.

Id	Name	Description	Default Position
object_type	Object Type		1
name	Name		2
key	Key		3
value	Value		4

Access Control:

User Category	Permission	Condition
Storage administrator	Allowed	
Application administrator	Disallowed	
Read-only users	Disallowed	
Technicians	Disallowed	

Chapter 20. Events

Chapter 21. Return Codes

The following are the return codes.

Return Code	Error Description
0	Success.
1	Command execution failed.
2	No connection to the system.
3	Password is required.
4	Password does not match system password.
7	Command not allowed from this client.
8	Bad XCLI option.
9	Internal XCLI error.

Chapter 22. Glossary

The following terms and abbreviations are used throughout this document. This list is in alphabetical order.

Alerting Event.

An event that triggers recurring event notifications until it is cleared.

API.

See *Application Program Interface (API)*

Application Program Interface (API).

The interface through which the application accesses the operating system and the other services.

Authorization Level.

The Authorization Level determines the permitted access level to the various functions of the GUI:

- Read only: Only viewing is allowed.
- Full: Enables access to all the configuration and control functions, including shutdown of the system. This level requires a password.

Auto Delete Priority.

As the storage capacity reaches its limits, snapshots are automatically deleted to make more space. The deletion takes place according to the value set for each snapshot, as follows:

- ″1″ = last to be deleted
- ...
- ″4″ = first to be deleted

Each snapshot is given a default Auto Delete Priority of ″1″ at creation.

Clearing Events.

The process of stopping the recurring event notification of *Alerting Events*.

CLI.

See *Command Line Interface (CLI)*

Command Line Interface (CLI).

The non-graphical user interface used to interact with the system via set commands and functions.

Completion Code.

The returned message sent as a result of running CLI commands.

Consistency Group.

A cluster of specific volumes that can all be cloned simultaneously as a group, thus creating a synchronized snapshot. The volumes in a Consistency Group are grouped into a single Volume Set. The volume set can be cloned into multiple Snapshot Sets under the specific Consistency Group.

See also *Snapshot Set*, *Volume Set*

Coupling.

A master volume and a slave volume connected together via mirroring definitions.

Data Module.

The part of the system rack containing the storage disks.

Destination.

See *Event Destination*.

Escalation.

A process in which event notifications are sent to a wider list of event destinations because the event was not cleared within a certain time.

Event.

A user or system activity that is logged (with an appropriate message).

Event Destination.

An address for sending *Event Notifications*.

Event Notification.

The process of notifying a user about an event.

Event Notification Rule.

A rule that determines which users are to be notified, for which events and by what means.

Fabric.

The hardware that connects workstations and servers to storage devices in a SAN. The SAN fabric enables any-server-to-any-storage device connectivity through the use of Fibre Channel switching technology.

FC.

See *Fibre Channel.*

FC-AL.

Also known as Arbitrated Lmetadata. A Fibre Channel topology that requires no Fibre Channel switches. Devices are connected in a one-way Lmetadata fashion.

FC-HBA.

Fibre Channel Host Bus Adapter.

Fibre Channel.

Serial data transfer architecture developed by a consortium of computer and mass storage device manufacturers and now being standardized by ANSI.

Functional Area.

One of the high level groupings of icons (Functional modules) of the left-hand pane of the GUI screen. For example, Monitor, Configuration or Volume Management.

See Functional Module.

Functional Module.

One of the icons of a Functional area, on the left-hand pane of the GUI screen. For example, System (under Monitor) or Hosts and LUNs (under Configuration).

See *Functional Area*.

Graphical User Interface (GUI).

On-screen user interface supported by a mouse and a keyboard.

GUI.

See *Graphical User Interface (GUI)*.

H/W.

Hardware.

HBA.

Host Bus Adapter.

Host.

A host is a port name of a host that can connect to the system. The system supports Fibre Channel and iSCSI hosts.

Host Interface Module.

The part of the system rack containing the host interface HBAs and manager software.

I/O.

Input/output.

Image Snapshot.

A snapshot that has never been unlocked. It is the exact image of the master volume from which it was copied at the time of its creation.

See also *Snapshot*

Internet Protocol.

Specifies the format of packets (also called datagrams), and their addressing schemes.

See also *Transmission Control Protocol (TCP)*

IOPs.

Input/output (I/O) per second.

IP.

See *Internet Protocol*

iSCSI.

Internet SCSI. An IP-based standard for linking data storage devices over a network and transferring data by carrying SCSI commands over IP networks.

Latency.

Amount of time delay between the moment an operation is initiated and the moment it begins to take effect.

Load Balancing.

Even distribution of load across all components of the system.

Locking.

Setting a volume (or snapshot) as unwritable (read-only).

LUN.

Logical Unit Number. Exports a system's volume into a registered host.

LUN Map.

A table showing the mappings of the volumes to the LUNs.

Master Volume.

A volume that has snapshots (meaning that it is mirrored for backup on a remote storage system) is called the master volume of its snapshots.

MIB.

Management Information Base. A database of objects that can be monitored by a network management system. SNMP Managers use standardized MIB formats to monitor SNMP agents.

Mirror Volume.

A volume which contains a backup copy of the original volume.

Mirroring.

See *Remote Mirroring*.

Modified State.

A snapshot state. A snapshot in modified state may never be used for restoring its master volume.

Multipathing.

Enables host interface modules direct access to any volume.

Pool.

See *Storage Pool*

SAN.

Storage Area Network.

SCSI.

Small Computer System Interface.

Slave Volume.

A volume which serves as a backup of a *Master Volume.*

SNMP.

Simple Network Monitor Protocol. A protocol for monitoring network devices.

See *MIB, SNMP Agent, SNMP Manager, Trap*.

SNMP Agent.

A device that reports information through the SNMP protocol to *SNMP Managers*.

SNMP Manager.

A host that collects information from *SNMP Agents* through the SNMP protocol.

SNMP Trap.

An SNMP message sent from the *SNMP Agent* to the *SNMP Manager*, where the sending is initiated by the SNMP Agent and not as a response to a message sent from the SNMP Manager.

Snooze.

The process of sending recurring event notifications until the events are cleared.

SMS Gateway.

An external server which is used to send SMSs.

SMTP Gateway.

An external host which is used to relay email messages via the SMTP protocol.

Snapshot.

A snapshot or copy of a volume. If a snapshot remains locked (unmodified), it may be used to restore its master volume (the volume of which it is a duplicate). Once unlocked, it is marked as modified and can no longer be used to restore its master volume.

See also *Image Snapshot*

Snapshot Set.

The resulting set of synchronized snapshots of a volume set in a Consistency Group.

See also *Consistency Group*, *Volume Set*

Storage Pool.

A reserved area of virtual disk space serving the storage requirements of the volumes. Specific volumes may be moved to (clustered into) user-defined Storage Pools.

Synchronization.

The process of making the master volume and slave volume identical after a communication down time or upon the initialization of the mirroring.

Target.

See *Remote Target*.

TCP/IP.

See *Transmission Control Protocol*, *Internet Protocol*

Transmission Control Protocol.

Transmission Control Protocol (TCP) on top of the Internet Protocol (IP). Establishes a virtual connection between a destination and a source over which streams of data can be exchanged.

See also *IP*

Trap.

See *SNMP Trap*.

Unassociated Volume.

A volume that is not associated with a Consistency Group.

See *Consistency Group*.

UPS.

Uninterruptible Power Supply. Provides battery backup power for a determined period of time, particularly to enable the system to power down in a controlled manner upon the occurrence of a lengthy power outage.

Volume.

A volume is a logical address space whose data content is stored on the system's disk drives. A volume may be virtually any size as long as the total allocated storage space of all the volumes does not exceed the net capacity of the system. A volume can be exported to an attached host via a LUN. A volume can be exported to multiple hosts simultaneously.

See also *Storage Pool, Unassociated Volume*.

Volume Snapshotting.

Creating a snapshot from a volume.

Volume Set.

A cluster of specific volumes in a Consistency Group, which can all be cloned simultaneously, thus creating a synchronized snapshot of all of them. The volume set can be cloned into multiple Snapshot Sets of the specific Consistency Group.

See also *Snapshot Set*, *Volume Set*

WWPN.

World Wide Port Name.

XCLI.

XIV CLI command set.

See *Command Line Interface (CLI)*

XDRP.

XIV's Disaster Recovery Program – XIV's *Remote Mirror* features.

Printed in USA

GC27-2213-02