

Configuring Arrays on HP Smart Array Controllers Reference Guide

Abstract

This document identifies, and provides instructions for, the array configuration tools available for HP ProLiant controller and server products. This document is for the person who installs, administers, and troubleshoots servers and storage systems. HP assumes you are qualified in the servicing of computer equipment and trained in recognizing hazards in products with hazardous energy levels.



© Copyright 2006, 2013 Hewlett-Packard Development Company, L.P.

The information contained herein is subject to change without notice. The only warranties for HP products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. HP shall not be liable for technical or editorial errors or omissions contained herein.

Confidential computer software. Valid license from HP required for possession, use or copying. Consistent with FAR 12.211 and 12.212, Commercial Computer Software, Computer Software Documentation, and Technical Data for Commercial Items are licensed to the U.S. Government under vendor's standard commercial license.

Microsoft®, Windows®, and Windows Server® are U.S. registered trademarks of Microsoft Corporation.

Contents

Overview of array configuration tools	5
Utilities available for configuring an array	5
Comparison of the utilities	5
Support for standard configuration tasks	6
Support for advanced configuration tasks	6
HP Smart Array Advanced Pack	8
About SAAP	8
Required hardware	8
Option ROM Configuration for Arrays	9
About ORCA	9
HP Array Configuration Utility	10
About ACU	10
Native support for 64-bit and 32-bit operating systems	10
Accessing ACU in the offline environment	10
Launching ACU with HP Intelligent Provisioning (Gen8 or later)	11
Launching ACU during POST (Gen8 or later)	11
Launching ACU from the SmartStart CD (G7 or earlier)	11
Launching ACU from an ISO image (all generations)	11
Accessing ACU in the online environment	15
Launching ACU on a local server	16
Launching ACU on a local server to configure a remote server	17
Launching ACU on a remote server to configure a local server	18
Using the ACU GUI	19
Navigating the GUI	19
Configuration tasks	28
Diagnostics tasks	39
Wizards	41
Using the ACU CLI	45
Opening the CLI in Console mode	46
Opening the CLI in Command mode	46
CLI syntax	46
Typical procedures	51
Using ACU scripting	68
Capturing a configuration	68
Using an Input script	68
Creating an ACU script file	69
Script file options	70
XML support	84
ACU scripting warning messages	87
ACU scripting error messages	87
HP Array Diagnostics and SmartSSD Wear Gauge Utility	92
About the utility	92
Reported information	92
Installing the utility	94

Setting up ADU Remote Service Mode.....	95
Launching the utility in CLI mode.....	95
Launching the utility in GUI mode.....	95
Diagnostic report procedures.....	96
Viewing the diagnostic report.....	96
Generating the diagnostic report.....	98
Identifying and viewing diagnostic report files.....	99
SmartSSD Wear Gauge report procedures.....	99
Viewing the SmartSSD Wear Gauge report.....	99
Generating the SmartSSD Wear Gauge report.....	100
Identifying and viewing SmartSSD Wear Gauge report files.....	101
Drive arrays and fault-tolerance methods.....	102
Drive arrays.....	102
Effects of a hard drive failure on logical drives.....	104
Fault-tolerance methods.....	105
RAID 0—No fault tolerance.....	105
RAID 1 and RAID 1+0 (RAID 10).....	106
RAID 1 (ADM) and RAID 10 (ADM).....	107
RAID 5—distributed data guarding.....	109
RAID 6 (ADG)—Advanced Data Guarding.....	109
RAID 50.....	111
RAID 60.....	112
Comparing the hardware-based RAID methods.....	112
Selecting a RAID method.....	113
Alternative fault-tolerance methods.....	113
Diagnosing array problems.....	115
Diagnostic tools.....	115
Troubleshooting resources.....	115
Acronyms and abbreviations.....	117
Documentation feedback.....	119
Index.....	120

Overview of array configuration tools

Utilities available for configuring an array

To configure an array on an HP Smart Array controller, three utilities are available:

- **HP Array Configuration Utility (ACU)**—An advanced utility that enables you to perform many complex configuration tasks
- **Option ROM Configuration for Arrays (ORCA)**—A simple utility used mainly to configure the first logical drive in a new server before the operating system is loaded
- **HP Online Array Configuration Utility for NetWare (CPQONLIN)**—A customized version of ACU for online configuration of servers that use Novell NetWare

Before you use a utility, confirm that the utility can support the required task. View a comparison of the utilities (on page 5).

Whichever utility you use, remember the following factors when you build an array:

- All drives grouped in a logical drive must be of the same type (for example, either all SAS or all SATA and either all hard drives or all solid state drives).
- For the most efficient use of drive space, all drives within an array should have approximately the same capacity. Each configuration utility treats every physical drive in an array as if it has the same capacity as the smallest drive in the array. Any excess capacity of a particular drive cannot be used in the array and is unavailable for data storage.
- The more physical drives configured in an array, the greater the probability that the array will experience a drive failure during any given period.
- To guard against the data loss that occurs when a drive fails, configure all logical drives in an array with a suitable fault-tolerance (RAID) method. For more information, see "Drive arrays and fault-tolerance methods (on page 102)."

Comparison of the utilities

Feature	ACU	CPQONLIN	ORCA
Interface	GUI, CLI, and Scripting	Menu-based	Menu-based or CLI
Languages	English, French, German, Italian, Japanese, Simplified Chinese, and Spanish	English	English
Source of executable file	Software CD, web, or embedded in the system	Software CD or web	Preinstalled in ROM of HP Smart Array controllers
When the utility can be used	All formats can be run in online and offline environments.	Any time the operating system is running	During POST, before the operating system is installed
Where the utility can be used	Any machine that uses a supported browser	Only on the local server	Only on the local server

ORCA supports only basic configuration tasks, whereas CPQONLIN and ACU provide full-range support for standard configuration tasks (on page 6). ACU also provides support for advanced configuration tasks (on page 6). Some of these advanced tasks are not available in all ACU interface formats (GUI, CLI, and Scripting).

Support for standard configuration tasks

A "+" indicates the feature or task is supported. A "-" indicates the feature or task is not supported.

Task	ACU	CPQONLIN	ORCA*
Create or delete arrays and logical drives	+	+	+
Assign a RAID level to a logical drive	+	+	+
Identify devices by causing their LEDs to illuminate	+	+	-
Assign a spare drive to an array	+	+	+
Share a spare drive among several arrays	+	+	-
Assign multiple spare drives to an array	+	+	-
Set the spare activation mode	+	-	-
Specify the size of the logical drive	+	+	-
Create multiple logical drives per array	+	+	-
Set the stripe size	+	+	-
Migrate the RAID level or stripe size	+	+	-
Expand an array	+	+	-
Set the expand priority, migrate priority, and accelerator ratio	+	+	-
Extend a logical drive	+	-	-
Set the boot controller	+	-	+
Configure HP SmartCache	+	-	-

*ORCA is not supported on the following controllers:

- HP Smart Array P430/2G
- HP Smart Array P431/2G
- HP Smart Array P731m/2G

Support for advanced configuration tasks

The following table uses these symbols:

+ —The ACU format supports this task.

- —The ACU format does not support this task.

+/- —Support for this task varies by controller. To support this task, some controllers must have SAAP activated by a registered license key.

To identify controller-specific feature support and SAAP requirements, see the controller user guide or the HP website (<http://www.hp.com/products/smartarray>).

For more information, see "About SAAP (on page 8)."

Procedure	ACU GUI	ACU CLI	ACU Scripting
Activate or delete license keys	+	+	+
Configure multiple systems identically	+ ¹	+ ¹	+
Configure a RAID 6 logical drive	+/-	+/-	+/-

Procedure	ACU GUI	ACU CLI	ACU Scripting
Configure a RAID 60 logical drive	+/-	+/-	+/-
Copy the configuration of one system to multiple systems	- ¹	- ¹	+
Disable a redundant controller	+ ²	+	-
Enable or disable a physical drive write cache	+	+	+
HP Drive Erase (replace the content of a physical drive or logical drive with zeros or random 0 and 1)	+/-	+/-	-
Identify devices by causing their LEDs to flash	+	+	-
Move an array (copy all array data to a new array and then delete the old array)	+/- ²	+/-	+/-
Optimize the controller performance for video	+/- ²	+/-	+/-
Re-enable a failed logical drive	+	+	-
Set the surface scan delay	+	+	+
Set the preferred controller for a logical drive (in systems that support redundant controllers)	+ ²	+	+
Shrink an array (restripe the data on an array to occupy fewer physical drives, then remove the excess drives from the array)	+/- ²	+/-	+/-
Split a RAID 1 array or recombine a split array (offline only)	+/-	-	-

¹Scripting is the most efficient method for this task.

²The task is supported only from the Configuration screen.

HP Smart Array Advanced Pack

About SAAP

SAAP is a collection of additional and advanced controller features embedded in the firmware of select Smart Array controllers.

To access SAAP features, activate the software with a registered license key.

SAAP 1.0 provides the following features for HP Smart Array G6 and G7 controllers:

- RAID 6 (ADG)
- RAID 60
- Advanced Capacity Expansion
- Mirror splitting and recombining in offline mode
- Physical Drive Erase
- Performance optimization for video on demand
- Dual domain

To access SAAP features for HP Smart Array G6 and G7 controllers, you must purchase a license key from HP. To obtain a license key, see the SAAP product page on the HP website (<http://www.hp.com/go/SAAP>).

To install the license key and activate SAAP, use one of the following methods:

- Installing a license key with ACU (on page 33)
- Installing a license key with ACU CLI ("Entering or deleting a license key" on page 54)
- Installing a license key with ACU Scripting ("LicenseKey, DeletelicenseKey" on page 74)

Some SAAP features are required for advanced configuration tasks in ACU. For a list of these tasks, see "Support for advanced configuration tasks (on page 6)."

Required hardware

For a list of Smart Array controllers that support SAAP, see the SAAP product page on the HP website (<http://www.hp.com/go/SAAP>).

To support some controller features, the controller may also require a hardware configuration that includes the following cache (array accelerator) options:

- A cache module that is 256 MiB or larger
- A compatible battery pack or capacitor pack

To obtain these options, contact an HP authorized reseller or see the HP website (<http://www.hp.com/products/smartarray>).

Option ROM Configuration for Arrays

About ORCA

ORCA is a ROM-resident array configuration utility that executes automatically during initialization of an HP Smart Array controller. This utility is designed to enable a logical drive to be configured on a new HP server before the operating system is installed:

- If the boot drive has not been formatted and the boot controller is connected to six or fewer physical drives, ORCA runs as part of the auto-configuration process when the new server is first powered up. During this auto-configuration process, ORCA uses all of the physical drives on the controller to set up the first logical drive. The RAID level used for the logical drive depends on the number of physical drives (one drive = RAID 0; two drives = RAID 1+0; three to six drives = RAID 5). If the drives have different capacities, ORCA locates the smallest drive and uses the capacity of that drive to determine how much space to use on each of the other drives.
- If the boot drive has been formatted or if there are more than six drives connected to the controller, you are prompted to run ORCA manually.

For more information about the auto-configuration process, see the *HP ROM-Based Setup Utility User Guide* on the Documentation CD that is provided with the server.

ORCA is available in two formats:

- Using the ORCA menu-driven interface
 - Using the ORCA CLI
- HP ProLiant 100 Series servers do not support the CLI format. For these servers, use the menu-driven interface.

Either format provides a quick and easy method for basic logical drive configuration. Both formats have limited support for standard configuration tasks (on page 6). However, these few tasks are adequate if your configuration needs are simple. For example, stripe size is predetermined by the RAID level that you choose, and the size of the logical drive is determined automatically by the size of the physical drives that you select.

HP Array Configuration Utility

About ACU

ACU is the main tool for configuring arrays on HP Smart Array controllers. It exists in three interface formats: the ACU GUI, the ACU CLI, and ACU Scripting. All formats provide support for standard configuration tasks (on page 6). ACU also provides support for advanced configuration tasks (on page 6). Some of the advanced tasks are available in only one format.

The diagnostic features in ACU are also available in the standalone software HP Array Diagnostics and SmartSSD Wear Gauge Utility (on page 92).

Starting with ACU version 9.0 and ProLiant Gen8 servers and server blades, ACU is accessible both offline and online:

- Accessing ACU in the offline environment (on page 10)
Using one of multiple methods, you can run ACU before launching the host operating system. In offline mode, users can configure or maintain detected and supported ProLiant devices, such as optional Smart Array controllers, integrated Smart Array controllers, and RAID Array controllers. Some ACU CLI features are only available in the offline environment, such as setting the boot controller and boot volume.
- Accessing ACU in the online environment (on page 15)
This method requires an administrator to download the ACU executables and install them. You can run ACU online after launching the host operating system.

Native support for 64-bit and 32-bit operating systems

ACU now offers a native 64-bit ACU application for supported 64-bit operating systems, eliminating the need for compatibility libraries. A 32-bit ACU application is also available. Users can choose to install the application that matches the OS installed on the server product.

The 64-bit ACU application is not a direct upgrade from 32-bit ACU. On 64-bit systems running 32-bit ACU, you must uninstall the 32-bit application, and then install the 64-bit application.

Future versions of ACU will be available as native 32-bit or native 64-bit applications, as long as 32-bit operating systems are supported.

Accessing ACU in the offline environment

To access and launch the ACU GUI in an offline environment, use one of the following methods:

- Launching ACU with HP Intelligent Provisioning (Gen8 or later) (on page 11)
- Launching ACU during POST (Gen8 or later) (on page 11)

- Launching ACU from the SmartStart CD (G7 or earlier) (on page 11)
 - Launching ACU from an ISO image (all generations) (on page 11)
- To access the ACU CLI or ACU Scripting in an offline environment, you must launch ACU from an ISO image.

When offline ACU launches, an **Execution Mode** screen does not appear, because ACU does not support Remote Service Mode in an offline environment. For this functionality, use ACU in an online environment ("[Accessing ACU in the online environment](#)" on page 15).

Launching ACU with HP Intelligent Provisioning (Gen8 or later)

1. Boot the server.
 2. Press **F10** to launch HP Intelligent Provisioning.
 3. At the main screen, select **Perform Maintenance**.
 4. At the **Maintenance** screen, select **Array Configuration Utility (ACU)**.
- The system launches the ACU GUI.

Launching ACU during POST (Gen8 or later)

1. Boot the server.
During POST, the system recognizes devices.
2. When the system recognizes a Smart Array controller, press **F5**.
The system launches the ACU GUI.

Launching ACU from the SmartStart CD (G7 or earlier)

1. Close all applications.
2. Insert the SmartStart CD into the CD-ROM drive.
3. Restart the server.
The server boots from the CD, and then loads the SmartStart executable and drivers.
4. When prompted, select the language and agree to the license restrictions.
5. Click **Maintain Server**.
6. Click **Array Configuration Utility**.
ACU opens, scans the local server, and detects controllers. This process can last up to 2 minutes. When controller detection is complete, the controllers are available on the **Controller/Device** menu.

Launching ACU from an ISO image (all generations)

To launch ACU, you can also boot from an ISO image. To prepare the image, use one of the following methods:

- Mounting the image on a local drive (on page 12)
- Mounting the image through iLO (on page 12)
- Burning the image to a CD or DVD (on page 12)
- Flashing the image to a USB memory key or SD card (on page 12)

- Installing the image on a PXE server (on page 13)

Booting from the ISO image on a drive, on a key, or through iLO provides the same GUI interface. The user can select to run Offline ACU GUI, ACU CLI, or ACU Scripting.

Mounting the image on a local drive

1. Download the HP ProLiant Offline Array Configuration Utility ISO image from the HP website (<http://h20000.www2.hp.com/bizsupport/TechSupport/SoftwareDescription.jsp?lang=en&cc=US&swItem=MTX-aad98a00c0d6469d8abf78e2f3&mode=4&idx=1>).
2. Using ISO mounting software, mount the Offline ACU ISO image to a drive.
3. Set the server to boot the image.
4. Reboot the server.

Mounting the image through iLO

This iLO functionality requires an iLO Advanced license.

1. Download the HP ProLiant Offline Array Configuration Utility ISO image from the HP website (<http://h20000.www2.hp.com/bizsupport/TechSupport/SoftwareDescription.jsp?lang=en&cc=US&swItem=MTX-aad98a00c0d6469d8abf78e2f3&mode=4&idx=1>).
2. Browse to the server iLO page.
3. Start the remote console for the server.
4. On the remote console, use the iLO mount feature to browse to the location of the ISO image.
5. Select the ISO image to be mounted.
6. Reboot the server.

Burning the image to a CD or DVD

1. Download the HP ProLiant Offline Array Configuration Utility ISO image from the HP website (<http://h20000.www2.hp.com/bizsupport/TechSupport/SoftwareDescription.jsp?lang=en&cc=US&swItem=MTX-aad98a00c0d6469d8abf78e2f3&mode=4&idx=1>).
2. Use a third-party software to burn the ISO image to a CD or DVD.
3. Set the server to boot from the optical drive.
4. Insert the CD or DVD.
5. Reboot the server.

Flashing the image to a USB memory key or SD card

1. Download the HP ProLiant Offline Array Configuration Utility ISO image from the HP website (<http://h20000.www2.hp.com/bizsupport/TechSupport/SoftwareDescription.jsp?lang=en&cc=US&swItem=MTX-aad98a00c0d6469d8abf78e2f3&mode=4&idx=1>).
2. Download the HP USB Key Utility for Windows from the HP website (<http://h20000.www2.hp.com/bizsupport/TechSupport/SoftwareDescription.jsp?lang=en&cc=US&swItem=MTX-eee9b3632ced47478f9ef85d43&mode=5>).



CAUTION: Before creating a bootable USB key from the Offline ACU ISO image, back up any critical data stored on the key to a different location. The utility overwrites all data on the key.

3. Using ISO mounting software, mount the Offline ACU ISO image to a Windows drive.
For this example, use "E:."
4. Insert a USB key into a USB connector on the Windows system.
For this example, use "F:."
5. Run the HP USB Key Utility.
6. At the main screen, select **Next**.
7. If you agree with the end user license agreement, select the **Agree** radio button, and then click **Next**.
8. Select **Create a bootable USB key from CD/DVD**, and then click **Next**.
9. At the requirements screen, click **Next**.
10. The Select CD/DVD drive: option is selected by default.
11. From the pull-down menu, select the drive letter for the location of the mounted ISO image (E:).
If the location is not shown, press **Rescan Source**, and then select the location.
12. The USB Key Drive Letter field shows the drive letter for the USB key (F:).
If the drive letter is not populated, select **Rescan Target**, and then select the drive letter.
13. Click **Next**.
14. At the following screen, click **Next**.
The USB key is overwritten with the Offline ACU ISO image.
15. Select **Finish**.

Installing the image on a PXE server

To install the Offline ACU ISO image on a PXE server and boot from the image over a network, use the following procedures:

1. Review the prerequisites (on page 13).
2. Set up PXELinux (on page 14).
3. Configure PXELinux (on page 14).
4. Specify the ISO image path (on page 14).

Depending on the network configuration, boot times might vary.

Prerequisites

Before proceeding with the configuration, you must have all the prerequisites:

- A good, working knowledge of PXE and TFTP
- A network with a DHCP server on it
- A TFTP server configured on the same network as the DHCP server
- A network file server that hosts the ISO images and is accessible by a PXE booted system
- PXELinux (<http://syslinux.zytor.com/wiki/index.php/PXELINUX>)

These instructions presume that you are using a Linux TFTP server and the TFTP package (<http://www.kernel.org/pub/software/network/tftp>). Other TFTP servers should work similarly.

Set up PXELinux

Before proceeding with the configuration, ensure that your TFTP server and PXELinux configuration is set up and configured properly. To set up PXELinux:

1. Download the HP ProLiant Offline Array Configuration Utility ISO image from the HP website (<http://h20000.www2.hp.com/bizsupport/TechSupport/SoftwareDescription.jsp?lang=en&cc=US&swItem=MTX-aad98a00c0d6469d8abf78e2f3&mode=4&idx=1>).
2. Copy the ISO image to the network file system, and note the location. NFS and Windows® file shares are supported.

For this example, use the following NFS and path to the ISO image:

```
192.168.0.99:/path/to/acucd/image/hpacuoffline-8.75-12.0.iso
```

3. Before proceeding, test your network file system to ensure that it is accessible.
4. Access the `/system` directory of the CD in one of the following ways:
 - o Burn and mount the ISO image.
 - o Extract the ISO image using a third-party tool.
5. Copy all the files from the `/system` directory of the CD to your TFTP server so that it is accessible by the TFTP software.

Configure PXELinux

1. Using the `isolinux.cfg` file from the `/system/` directory of the CD as a guide, copy the labeled targets to your PXELinux configuration file. You do not need to include the entire file:

```
label sos
    MENU LABEL HP ProLiant Offline ACU Image
    kernel hpboot_v.c32
    append vmlinuz initrd=initrd.img media=cdrom rw root=/dev/ram0
ramdisk_size=257144 init=/bin/init loglevel=3 ide=nodma ide=noraid
pnpbios=off vga=791 splash=silent showopts TYPE=AUTOMATIC
label vsos
    MENU LABEL HP ProLiant Offline ACU Image
    kernel hpboot_v.c32
    append vmlinuz initrd=initrd.img media=cdrom rw root=/dev/ram0
ramdisk_size=257144 init=/bin/init loglevel=3 ide=nodma ide=noraid
pnpbios=off vga=791 splash=silent showopts TYPE=MANUAL
```

2. Replace the lines `kernel hpboot_v.c32` with `kernel vmlinuz`.
3. Remove `vmlinuz` from the `append` line.

The paths to the files on the TFTP server are `vmlinuz` and `initrd.img`. You must modify them to include any directories or naming conventions you may have on your TFTP server.

Specify the ISO image path

For the PXE-booted server to find the ISO image, you must add the ISO image path to the `append` line in the PXELinux configuration file.

Add the following arguments:

```
isol=nfs://192.168.0.99/path/to/acucd/image/hpacuoffline-8.75-12.0.iso
isolmnt=/mnt/bootdevice
```

The `iso1` parameter helps the PXE-booted ACU Offline CD locate the ISO image. The `iso1 mnt` parameter tells the PXE-booted ACUCD where the `iso1` image must be mounted.

Your final configuration must be similar to the following example:

```
label sos
  MENU LABEL HP ProLiant Offline ACU Image
  kernel vmlinuz
  append initrd=initrd.img media=cdrom rw root=/dev/ram0
  ramdisk_size=257144 init=/bin/init loglevel=3 ide=nodma ide=noraid
  pnpbios=off vga=791 splash=silent showopts TYPE=AUTOMATIC
  iso1=nfs://192.168.0.99/path/to/acucd/image/hpacuoffline-8.75-12.0.iso
  iso1mnt=/mnt/bootdevice

label vsos
  MENU LABEL HP ProLiant Offline ACU Image
  kernel vmlinuz
  append initrd=initrd.img media=cdrom rw root=/dev/ram0
  ramdisk_size=257144 init=/bin/init loglevel=3 ide=nodma ide=noraid
  pnpbios=off vga=791 splash=silent showopts TYPE=MANUAL
  iso1=nfs://192.168.0.99/path/to/acucd/image/hpacuoffline-8.75-12.0.iso
  iso1mnt=/mnt/bootdevice
```

You can add additional ISO images by specifying the additional `iso#` and `iso#mnt` arguments, for example, `iso2=/path/to/iso2.iso iso2mnt=/mnt/iso2`.

Supported network file systems

The following network file systems are supported for use with PXE booting:

- NFS:

```
iso1=nfs://192.168.0.99/path/to/acucd/image/hpacuoffline-8.75-12.0.iso
iso1mnt=/mnt/bootdevice
```

NFS volumes are mounted with the following options:

 - `-o ro`
 - `nolock`
- Windows® operating systems:

```
iso1=smbfs://192.168.0.99/share/path/to/acucd/image/hpacuoffline-8.75-12.0.iso iso1mnt=/mnt/bootdevice
```
- Windows® operating systems with login credentials:

```
iso1=smbfs://user:password@192.168.0.99/share/path/to/acucd/image/hpacuoffline-8.75-12.0.iso iso1mnt=/mnt/bootdevice
```

Accessing ACU in the online environment

To access, install, and launch ACU in the online environment, you must download the ACU executables. All three formats have separate executables.

Starting with version 8.28.13.0, ACU Scripting is now a standalone application that is distributed with the ACU CLI application. In ACU versions prior to 8.28.13.0, the scripting executable was provided with the ACU GUI component.

Users familiar with the previous versions of ACU Scripting must now install the ACU CLI application to obtain the scripting executable. The new ACU scripting executable (hpacuscripting) replaces the former executable (cpqacuxe) in all scripts.

For information about the minimum monitor settings and the version numbers of supported operating systems and browsers, see the README.txt file provided with the executable.

To use ACU in the online environment:

1. Obtain the executable files from one of the following locations:
 - o The HP website (<http://www.hp.com/support>)
When prompted for product information, enter the appropriate server or server blade model name.
 - o The software CD that is provided with the controller
2. Follow the installation instructions provided with the executable.
3. After the executables are installed, launch each executable in the following manner:
 - o GUI—Click **Start**, and then select **Programs>HP System Tools>HP Array Configuration Utility>Setup HP Array Configuration Utility**.
Depending on your configuration scenario, choose one of the following options:
 - Launching ACU on a local server (on page 16)
 - Launching ACU on a local server to configure a remote server (on page 17)
 - Launching ACU on a remote server to configure a local server (on page 18)
 - o CLI—Click **Start**, and then select **Programs>HP System Tools>HP Array Configuration Utility>Setup HP Array Configuration Utility CLI**.
 - o Scripting—Run `hpacuscripting.exe`.

Launching ACU on a local server

Microsoft OS

1. Click **Start**, and then select **Programs>HP System Tools>HP Array Configuration Utility>Setup HP Array Configuration Utility**.
The **Execution Mode** screen appears.
 - o If **Local Application Mode** is selected, continue with step 2.
 - o If **Remote Service Mode** is selected, select **Local Application Mode**, reboot the server, and then continue with step 2.
2. Click **Start**, and then select **Programs>HP System Tools>HP Array Configuration Utility**.
ACU launches in either a browser (older versions) or application window (v8.70 and later) ACU then scans the system and detects controllers. This process can last up to 2 minutes. When controller detection is complete, the controllers are available on the **Controller/Device** menu.
3. Configure a controller ("[Configuring a controller](#)" on page 30).
When configuration is complete, continue with the next step.
4. If you changed to Local Application mode in step 1, and you are finished configuring arrays on this server, do the following:
 - a. Click **Start**, and then select **Programs>HP System Tools>HP Array Configuration Utility>Setup HP Array Configuration Utility**.
 - b. When the **Execution Mode** screen appears, select **Remote Service Mode**.

- c. Reboot the server.
5. (Optional) To make newly created logical drives available for data storage, use the operating system disk management tools to create partitions and format the drives.

Linux OS

1. From any command prompt, enter one of the following:
 - o For local mode, enter: `cpqgauxe-nosmh`
 - o For remote mode, enter: `cpqacuxe-R`ACU launches in a browser (Mozilla Firefox).
2. For a list of options, enter the following:
`cpqacuxe-h`

Launching ACU on a local server to configure a remote server

1. On the local server (host), click **Start**, and then select **Programs>HP System Tools>HP Array Configuration Utility>Setup HP Array Configuration Utility**.
The **Execution Mode** screen appears.
 - o If **Remote Service Mode** is selected, continue with step 2.
 - o If **Local Application Mode** is selected, select **Remote Service Mode**, reboot the server, and then continue with step 2.
2. On the remote server, open the browser.
3. Enter the following text into the address field of the remote browser (where *servername* is the name or IP address of the host):
`http://servername:2301`
The login screen for the System Management Homepage opens.
4. Enter your login credentials:
 - o If you are using version 2.0.0 or later of the System Management Homepage, use your operating system user name and password.
 - o If you are using an earlier version of the System Management Homepage, use your WBEM user name and password.The System Management Homepage opens.
For more information about the System Management Homepage, see the following:
 - o The HP System Management Homepage web page (<http://h18013.www1.hp.com/products/servers/management/agents/index.html>)
 - o The *HP System Management Homepage Installation Guide* on the HP website (<http://www.hp.com>)
5. Click **Array Configuration Utility** on the left side of the screen.
ACU opens, scans the remote server, and detects controllers. This process can last up to 2 minutes. When controller detection is complete, the controllers are available on the **Controller/Device** menu.
6. Configure a controller ("[Configuring a controller](#)" on page 30).
When configuration is complete, continue with the next step.
7. To operate ACU on this server in Local Application mode, do the following:

- a. Click **Start**, and then select **Programs>HP System Tools>HP Array Configuration Utility>Setup HP Array Configuration Utility**.
 - b. When the **Execution Mode** screen appears, select **Local Application Mode**.
 - c. Reboot the server.
8. (Optional) To make newly created logical drives available for data storage, use the operating system disk management tools to create partitions and format the drives.

Launching ACU on a remote server to configure a local server

1. On the server where ACU is installed, click **Start**, and then select **Programs>HP System Tools>HP Array Configuration Utility>Setup HP Array Configuration Utility**.
The **Execution Mode** screen appears.
 - o If **Remote Service Mode** is selected, continue with step 2.
 - o If **Local Application Mode** is selected, select **Remote Service Mode**, reboot the server, and then continue with step 2.
2. On the server that you want to configure, connect to the Systems Insight Manager server (port: 280), and then log in.
3. Select **Device Queries**.
4. Under **Device by Type**, select **All Servers**.
5. Connect to the server that is running ACU.
6. Under **Device Links**, select **System Management Homepage**.
The login screen for the System Management Homepage opens.
7. Log in using your credentials:
 - o If you are using version 2.0.0 or later of the System Management Homepage, use your operating system user name and password.
 - o If you are using an earlier version of the System Management Homepage, use your WBEM user name and password.

The System Management Homepage opens.

For more information about the System Management Homepage, see the following:

 - o The HP System Management Homepage web page (<http://h18013.www1.hp.com/products/servers/management/agents/index.html>)
 - o The *HP System Management Homepage Installation Guide* on the HP website (<http://www.hp.com>)
8. Click **Array Configuration Utility** on the left side of the screen.
ACU opens, scans the remote server, and detects controllers. This process can last up to 2 minutes. When controller detection is complete, the controllers are available on the **Controller/Device** menu.
9. Configure a controller ("[Configuring a controller](#)" on page 30).
When configuration is complete, continue with the next step.
10. To operate ACU on the remote server in Local Application Mode, do the following:
 - a. Click **Start**, and then select **Programs>HP System Tools>HP Array Configuration Utility>Setup HP Array Configuration Utility**.
 - b. When the **Execution Mode** screen appears, select **Local Application Mode**.

- c. Reboot the server.
11. (Optional) To make newly created logical drives available for data storage, in a Windows OS, use the operating system disk management tools to create partitions and format the drives.

Using the ACU GUI

Access ACU with one of the many methods available:

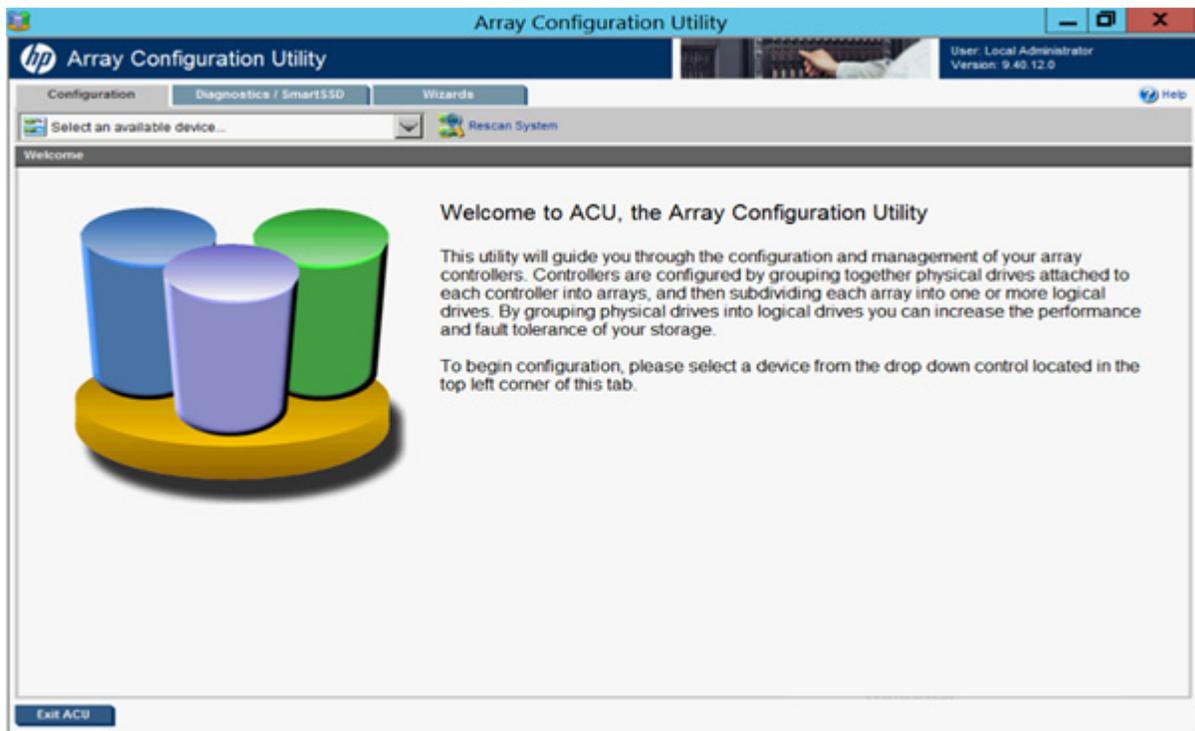
- Accessing ACU in the offline environment (on page 10)
- Accessing ACU in the online environment (on page 15)

When you launch the ACU GUI, the application opens and ACU scans the system and detects controllers. This process can last up to 2 minutes. When controller detection is complete, the controllers are available on the **Controller/Device** menu.

After the GUI is open, tasks are distributed among categories. For more information, see "Navigating the GUI (on page 19)."

Navigating the GUI

When you open ACU, the **Welcome** screen appears.



The following elements are visible:

- Three tabs appear near the top left of the screen. The latest version of the ACU GUI uses tab navigation. Clicking a tab displays the screen and tasks for the following categories:
 - **Configuration**—This screen displays available controller and array tasks that the user can select and complete manually.

In previous versions of ACU, this process was called the **Standard Configuration** mode. For more information, see "Configuration screen (on page 20)."

- **Diagnostics/SmartSSD**—This screen displays a list of controllers and options for generating, viewing, and saving diagnostic reports for those controllers. In previous versions, this tab and screen were called **Diagnostics**. For more information, see "Diagnostics/SmartSSD screen (on page 22)."
- **Wizards**—This screen displays available array and controller tasks that ACU can complete automatically or with minimal user input, such as express configuration.

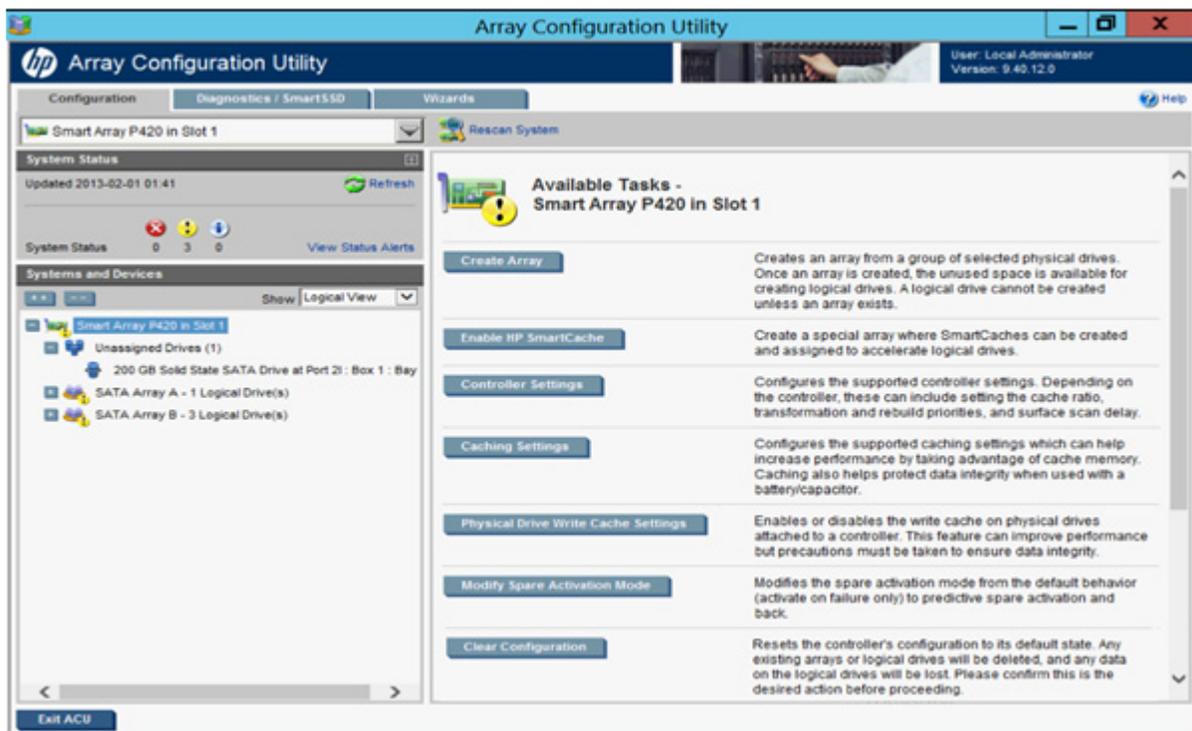
In previous versions of ACU, this functionality was called the Configuration Wizards mode and Express mode. For more information, see "Wizards screen (on page 25)."

- The **Controller/Device** menu is below the tabs.
To select a device, click on the menu, and then select a device. Use the scroll bar to view all the devices, as needed.
- The **Rescan System** button is to the right of the menu.
After adding or removing devices, click **Rescan System** to update the list of available devices.
- The **Help** button is near the top right of the screen.
To access help topics, press the **H** key or click **Help**. For more information, see "ACU Help (on page 28)."
- The **Exit ACU** button is near the bottom left of the screen.

Configuration screen

To access this screen, click the **Configuration** tab.

The **Configuration** screen displays the GUI elements from the **Welcome** screen and provides status, more detailed information, and available tasks or options for the selected device.

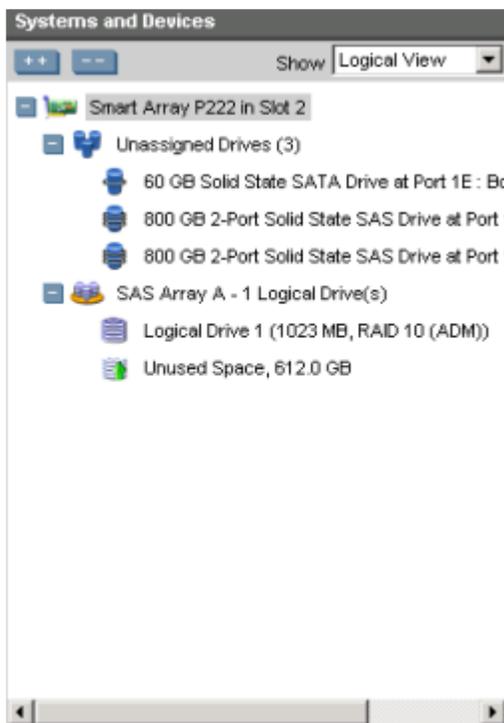


When a device is selected, the following elements appear:

- **System Status**—This panel, at left, provides the following information and functionality:
 - Date and time stamps for the status
 - A Refresh button to refresh the status
 - Status icons (critical, warning, and informational) with the number of individual alerts for each category
 - A View Status Alert link that displays device-specific alerts on the right side of the screen

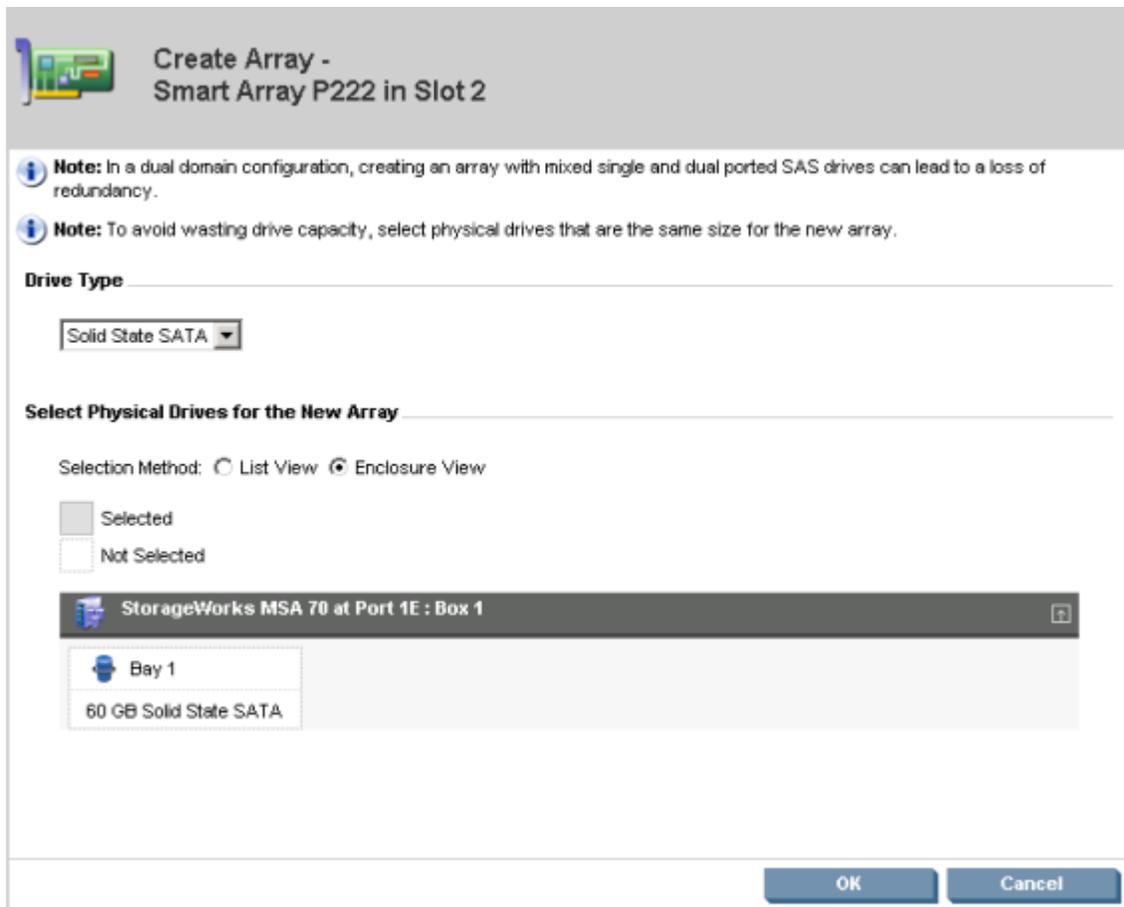


- **Systems And Devices**—This panel, at left, provides the following information and functionality:
 - A tree detailing systems, controllers, arrays, physical drives, and logical drives
 - Expand all and collapse all buttons
 - A **Show** menu that toggles between **Logical View** and **Physical View**



- **Available Tasks**—This panel, at right, provides the following information and functionality:
 - Tasks that are available for the selected device based on its current status and configuration

- Options and information pertinent to the task, after a task is selected

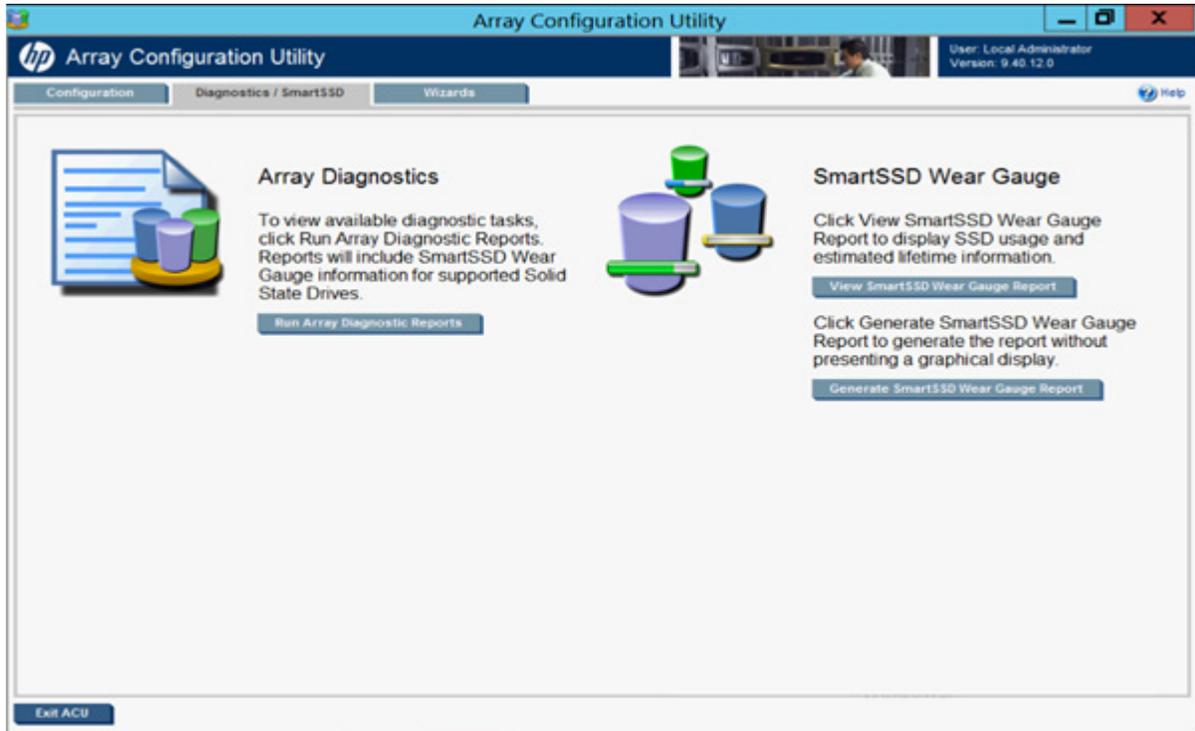


For a list of possible tasks that are available on the **Configuration** screen, see "Configuration tasks (on page 28)."

Diagnostics/SmartSSD screen

To access this screen, click the **Diagnostics/SmartSSD** tab.

The **Diagnostics/SmartSSD** screen provides a list of controllers and options related to generating and viewing diagnostic reports.



Tasks are selectable from this screen. Additional tasks are available upon clicking **Run Array Diagnostics Reports**.

On the diagnostic report screen, when a device is selected, the following elements appear:

- **Report Contents**—This panel, at left, provides the following information and functionality:
 - A list of all connected controllers and devices

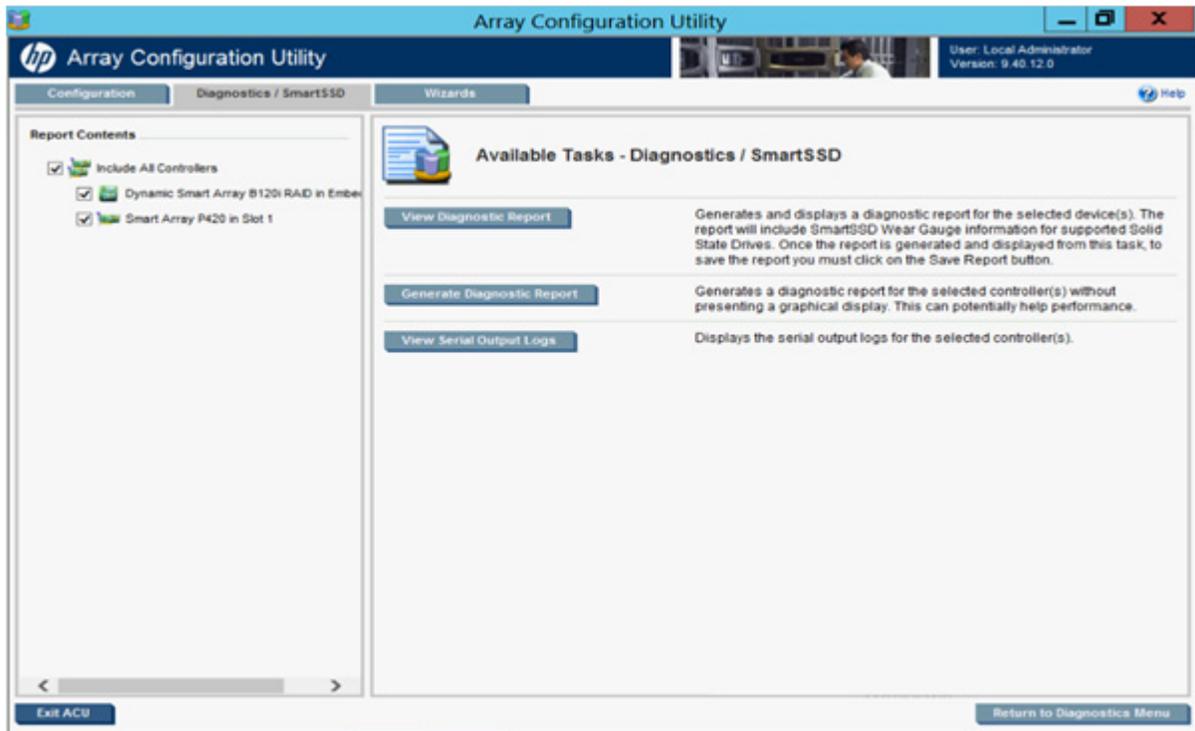
- Check boxes for individual controllers or all controllers

Report Contents _____

-  Include All Controllers
 -  Smart Array P411 in Slot 6
 -  Smart Array P800 in Slot 8
 -  Smart Array P400 in Slot 10
 -  MSA1000 in MSA1001
 -  MSA1000 in MSA1002
 -  MSA1500 CS in MSA1502

- **Available Tasks**—This panel, at right, provides the following information and functionality:
 - Tasks that are available for the selected device based on its current status and configuration

- Options and information pertinent to the task, after a task is selected

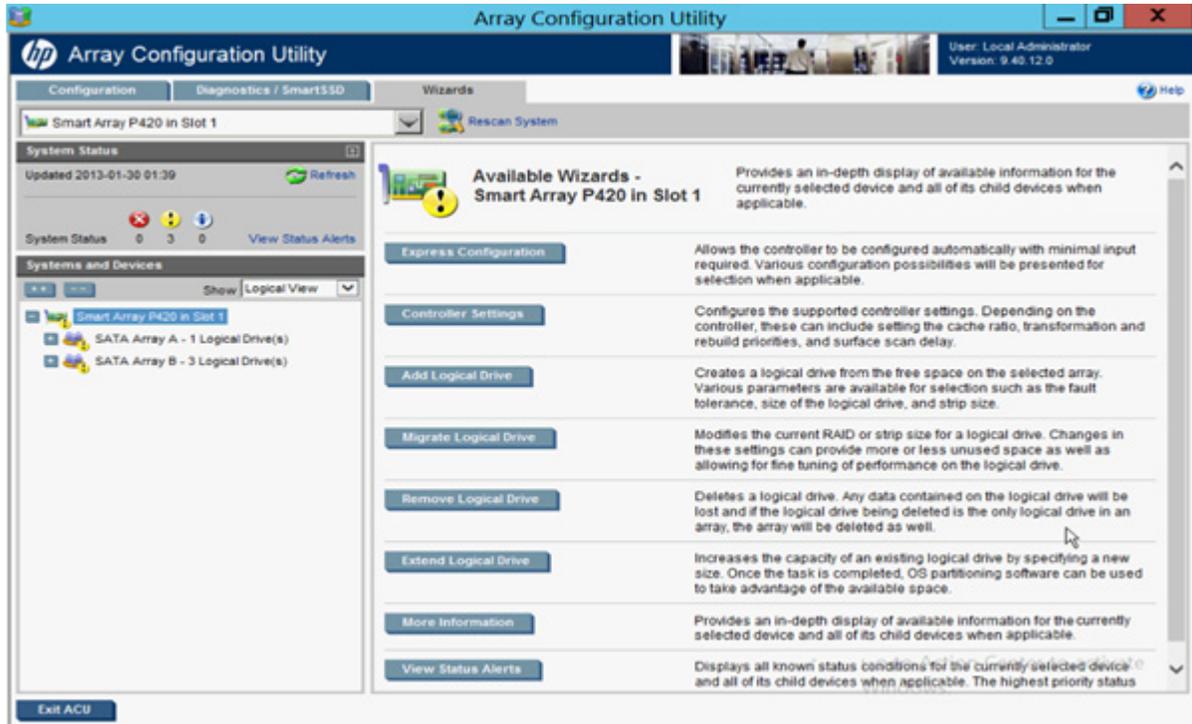


For a list of possible tasks that are available on the **Diagnostics/SmartSSD** screen, see "Diagnostics tasks (on page 39)."

Wizards screen

To access this screen, click the **Wizards** tab.

The **Wizards** screen displays the GUI elements from the **Welcome** screen and provides status, more detailed information, and available wizards or options for the selected device.



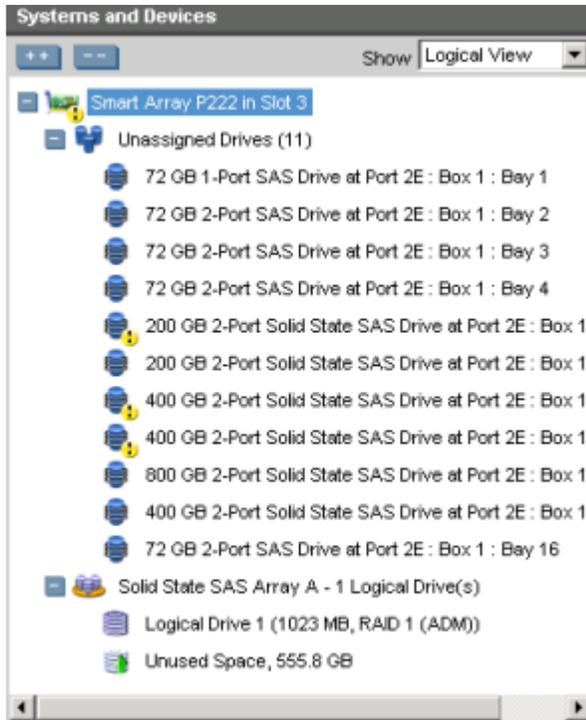
When a device is selected, the following elements appear:

- **System Status**—This panel, at left, provides the following information and functionality:
 - Date and time stamps for the status
 - A Refresh button to refresh the status
 - Status icons (critical, warning, and informational) with the number of individual alerts for each category
 - A View Status Alert link that displays device-specific alerts on the right side of the screen



- **Systems And Devices**—This panel, at left, provides the following information and functionality:
 - A tree detailing systems, controllers, arrays, physical drives, and logical drives
 - Expand all and collapse all buttons

- A **Show** menu that toggles between **Logical View** and **Physical View**



In this example, the Systems And Devices information continues past the edge of the panel. To view all of the information, use the horizontal scroll bar or use the mouse to widen the panel.

- **Available Wizards**—This panel, at right, provides the following information and functionality:
 - Wizards that are available for the selected device based on its current status and configuration

- Options and information pertinent to the wizard, after a wizard is selected



For a list of possible wizards that are available on the **Wizards** screen, see "Wizards (on page 41)."

ACU help

The Help button, at upper right, opens the embedded ACU help file. In addition to providing information about the main screens and tabs, Help also provides several useful topics for new users, including the following:

- **Image Legend**—A visual reference list defining the icons and graphical buttons used in ACU
- **Keyboard Controls**—An explanation and list of keyboard functions for navigating the GUI
- **Keyboard Shortcuts**—A list of keys and operations they perform within the GUI

To view these help topics and others, press the **H** key or click **Help**. When the Help window opens, expand the topic "Getting Started with ACU."

The glossary in ACU help defines industry standard and HP terms as they relate to the ACU application.

Configuration tasks

From the **Configuration** screen, you can perform tasks related to controllers, arrays, physical drives, and logical drives.

For certain tasks, the controller must have SAAP activated by a registered license key. For more information, see "About SAAP (on page 8)."

When a controller or device is selected, the tasks that appear are a subset of the total number of possible tasks for the selected item. ACU lists or omits tasks based on the controller model and configuration. For example, if the selected controller has no unassigned physical drives, Create Array is not an available task.

The following table lists all the possible tasks for every type of item.

Item	Tasks
Controller	Advanced Controller Settings* ** Array Accelerator Settings Clear Configuration Controller Settings Create Array Disable Standby Controller Enable HP SmartCache † Manage License Keys* More Information Physical Drive Write Cache Settings Redundancy Settings* View Status Alerts
Array	Create Array Create Logical Drive Create Split Mirror Backup Delete Expand Array Heal Array** Manage Split Mirror Backup More Information Move Array** Re-Mirror Array** Replace Array** Shrink Array** Spare Management Split Mirrored Array** View Status Alerts
Logical drive	Create Logical Drive Delete Erase Drive* ** Extend Logical Drive Migrate RAID/Stripe Size Move Logical Drive* ** More Information Re-enable Failed Logical Drive View Status Alerts
Unused space	Create Logical Drive More Information
Physical drive	Erase Drive** View Status Alerts
Unassigned drives	Create Array More Information

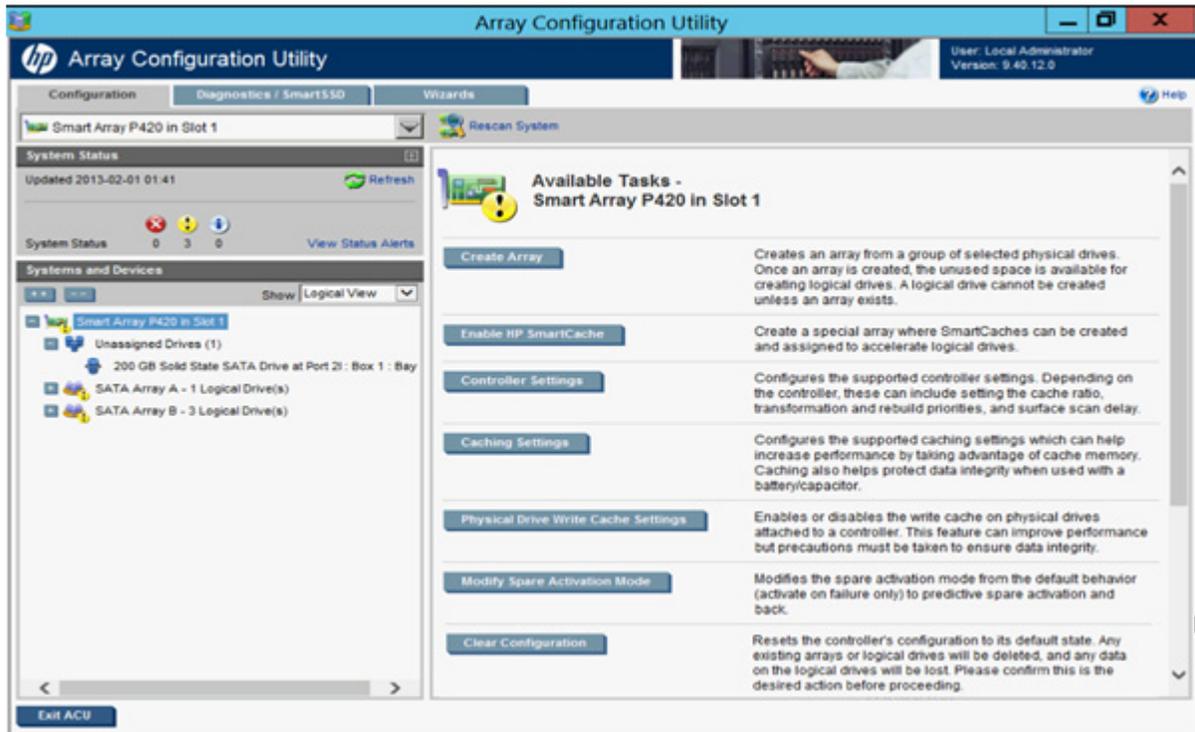
*This task is not available on all controller models.

**If performed with an HP Smart Array G6 or G7 controller, this task requires a registered SAAP license key or a controller where SAAP functionality is standard. See "About SAAP (on page 8)."

If performed with a Gen8 controller, this task requires a registered HP SmartCache license key.

Configuring a controller

1. Open ACU.
For more information, see "Using the ACU GUI (on page 19)."
2. Select a controller from the **Controller/Device** menu.
The **Configuration** screen appears.

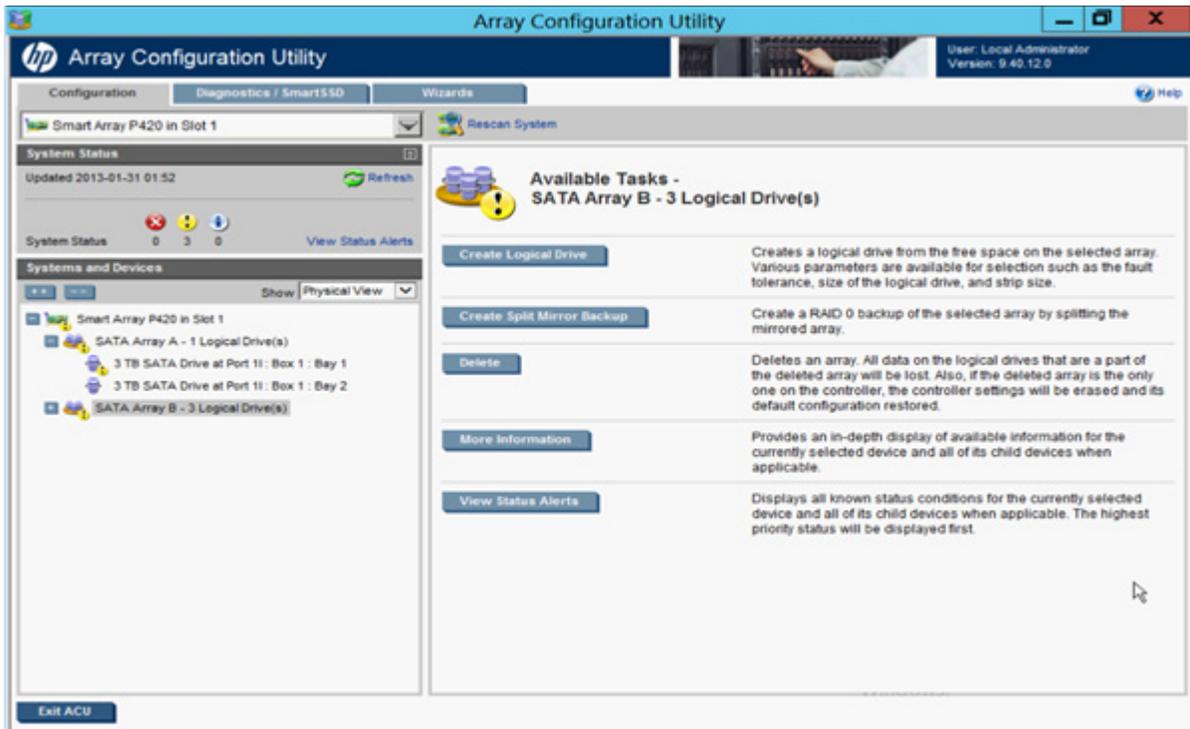


3. Configure the controller:
 - o To configure manually, see "Performing a Configuration task (on page 30)."
 - o To configure with a wizard, see "Using Wizards (on page 42)" or "Using Express Configuration (on page 44)."
4. When prompted, save the configuration.
5. Do one of the following:
 - o Configure an additional controller. Repeat steps 3 through 5.
 - o Click **Exit ACU**.

Performing a Configuration task

1. Open ACU.
For more information, see "Using the ACU GUI (on page 19)."
If ACU is already open, click the **Configuration** tab.
2. Select a device from the Controller/Device menu.

The System Status, Systems And Devices, and Available Tasks panels appear. The listed tasks are available for this device in its current configuration. For more information, see "Configuration tasks (on page 28)."



3. Click a task button.

A list of all possible options for that task appears on the right side of the screen, replacing the task list.

Controller Settings -
Smart Array P400 in Slot 10

Rebuild Priority

High
 Medium
 Low

Surface Scan Delay

15 Valid Range: 1 - 30 seconds

Note: Clicking Save will commit changes to the controller

Save Cancel

4. Select the settings or configuration options for the device.
5. Use the Next and Back buttons to navigate multiple screens of options.
6. Click **Save** or **OK**.

Rapid Parity Initialization

When you create a logical drive, you must initialize the parity using Rapid Parity Initialization.

RAID levels that use parity (RAID 5, RAID 6 (ADG), RAID 50, and RAID 60) require that the parity blocks be initialized to valid values. Valid parity data is required to enable enhanced data protection through background surface scan analysis and higher performance write operations. Two initialization methods are available:

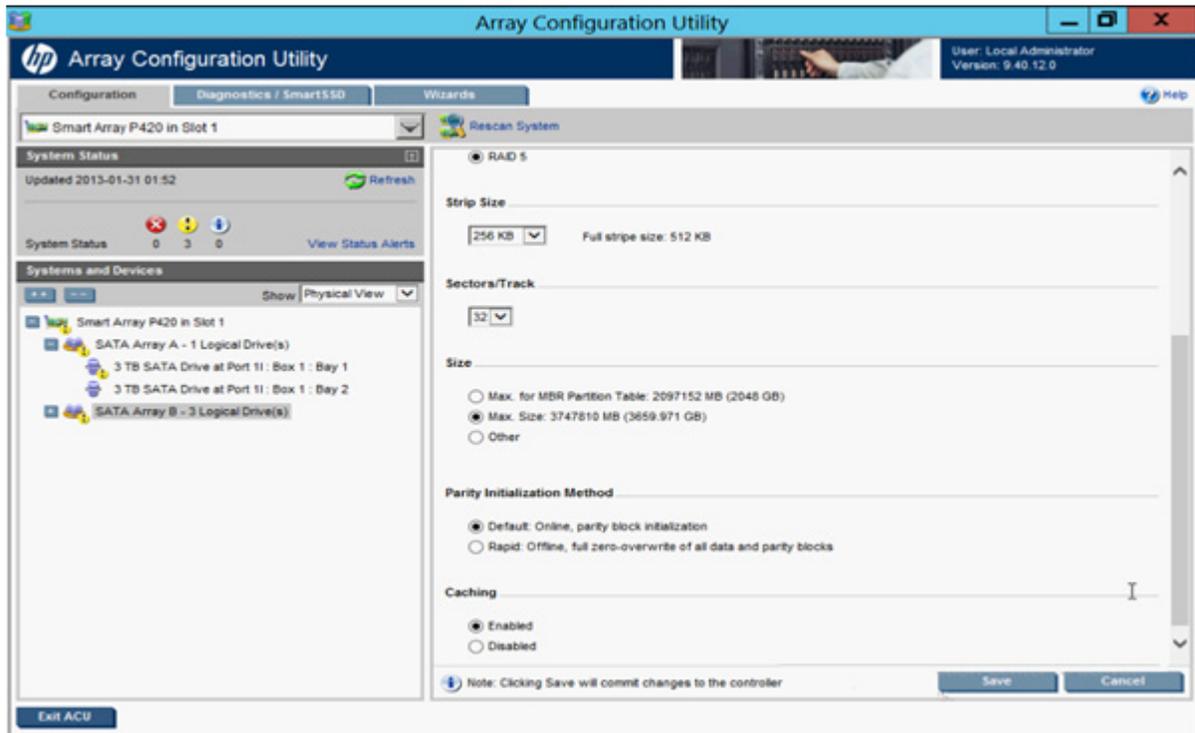
- **Default** – Initializes parity blocks in the background while the logical drive is available for access by the operating system. A lower RAID level results in faster parity initialization.
- **Rapid** – Overwrites both the data and parity blocks in the foreground. The logical drive remains invisible and unavailable to the operating system until the parity initialization process completes. All parity groups are initialized in parallel, but initialization is faster for single parity groups (RAID 5 and RAID 6). RAID level does not affect system performance during rapid initialization.

Rapid Parity Initialization is available only for supported controllers and in arrays composed of supported physical drives.

To select the method for parity initialization:

1. Open ACU.
For more information, see "Using the ACU GUI (on page 19)."

- If ACU is already open, click the **Configuration** tab.
2. Select a controller from the Controller/Device menu.
The System Status, Systems And Devices, and Available Tasks panels appear.
 3. Select **Create a logical drive**.
 4. Make selections for **Fault Tolerance**, **Strip Size**, **Sectors/Track**, **Size**, and **Caching**.
 5. Under **Parity Initialization Method**, select either **Default** or **Rapid**.



6. Click **Save**.

Installing a license key with ACU

You can use HP SSA to install the license key and activate SAAP and HP SmartCache functionality. For more information, see "About SAAP (on page 8)."

To install a license key:

1. Open ACU.
For more information, see "Using the ACU GUI (on page 19)."
If ACU is already open, click the **Configuration** tab.
2. Select a controller from the Controller/Device menu.
The System Status, Systems And Devices, and Available Tasks panels appear.
3. In the Available Tasks panel, click **Manage License Keys**.
Specific license key tasks appear. A complete list of existing license keys also appears.
4. Click **Add License Key**.
5. Enter the license key number.
6. Click **Save**.

Changing the Spare Activation Mode

The spare activation mode feature enables the controller firmware to activate a spare drive under the following conditions:

- When a data drive reports a predictive failure (SMART) status
- When a data drive fails; this mode is the default.

In normal operations, and for older controllers, the firmware starts rebuilding a spare drive only when a data drive fails. With the predictive failure activation mode, rebuilding can begin before the drive fails, reducing the likelihood of data loss that could occur if an additional drive fails.

To change the Spare Activation Mode:

1. Open ACU.
For more information, see "Using the ACU GUI (on page 19)."
If ACU is already open, click the **Configuration** tab.
2. Select a controller from the **Controller/Device** menu.
The **System Status**, **Systems And Devices**, and **Available Tasks** panels appear.
3. In the **Available Tasks** panel, click **Modify Spare Activation Mode**.
4. From the menu, select one of the following modes:
 - Failure Spare Activation
 - Predictive Spare Activation
5. Click **Save**.

Changing the Spare Management mode

The Spare Management feature provides multiple methods for handling spare behavior. You can choose from the following options:

- **Dedicated** – The failed data drive, when replaced, must be rebuilt from the data on the spare drive. In Dedicated mode, one spare can be dedicated to multiple arrays.
- **Auto-Replace Drives** – The spare for the failed data drive automatically becomes the replacement data drive. When the spare is replaced, the data drive does not need to be rebuilt. In Auto-replace mode, spare drives cannot be shared between arrays.

If assigning **Auto-Replace Drives** mode to an array with a RAID 0 drive, **Spare Activation Mode** must be set to **Predictive Spare Activation** mode.

To change the Spare Management mode:

1. Open ACU.
For more information, see "Using the ACU GUI (on page 19)."
If ACU is already open, click the **Configuration** tab.
2. Select a controller from the **Controller/Device** menu.
The **System Status**, **Systems And Devices**, and **Available Tasks** panels appear.
3. In the **Available Tasks** panel, click **Spare Management**.
4. From the menu, select one of the following modes:
 - Dedicated
 - Auto-Replace Drives

5. Click **Save**.

About HP SmartCache

HP SmartCache enables solid state drives to be used as caching devices for hard drive media. Data can be accessed from the solid state drive instead of hard drives. HP SmartCache provides the following features:

- Accelerates application performance
- Provides lower latency for transactions in applications
- Supports all operating systems, without the need for changes

To support HP SmartCache, the Smart Array Controller firmware must be 3.42 or later. ACU must be version 9.40.12.0 or later.

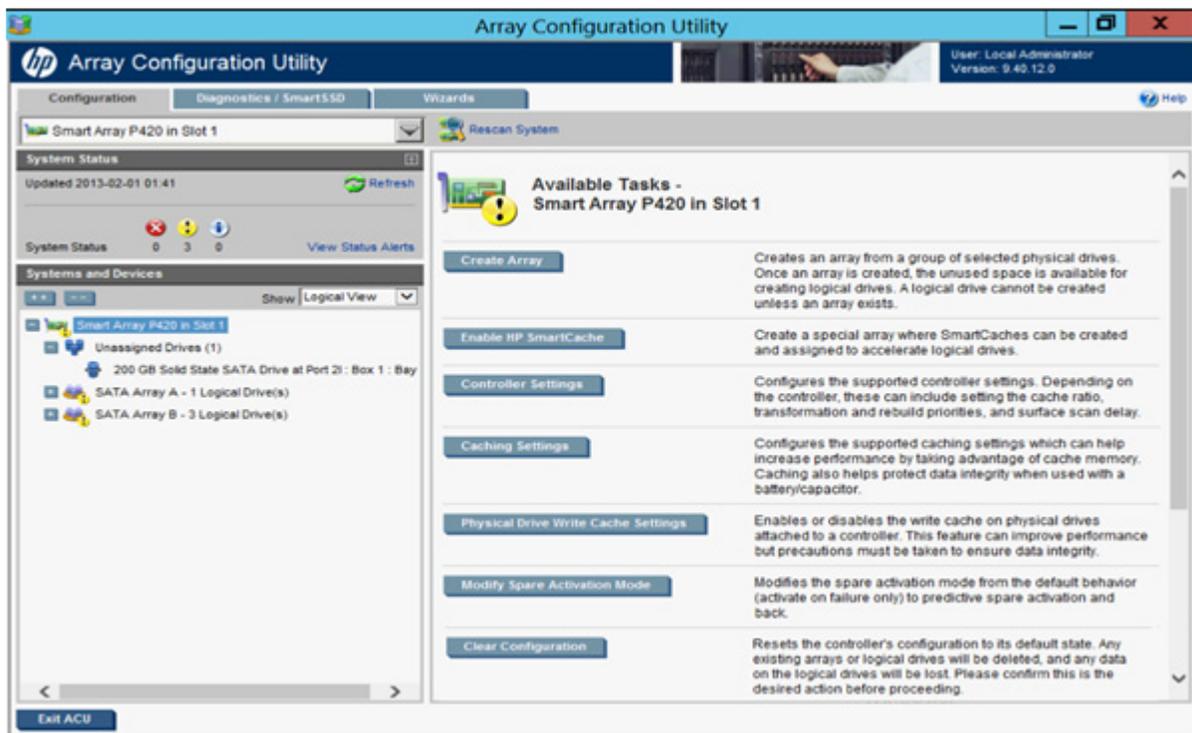
HP SmartCache requires a Smart Array Advanced Pack license (<http://www.hp.com/go/SAAP>).

Enable HP SmartCache

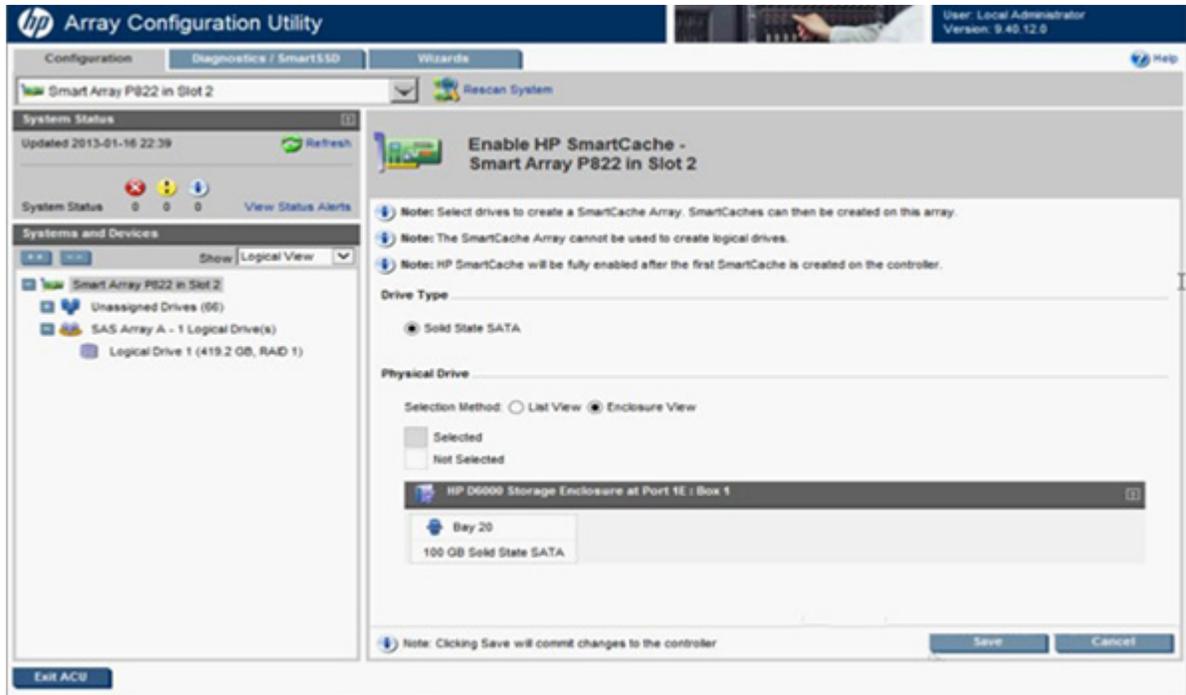
Enabling HP SmartCache creates an array to create SmartCaches and assign them to accelerate logical drives. At least one logical drive must be created on the controller prior to enabling HP SmartCache.

To enable HP SmartCache:

1. Open ACU.
For more information, see "Using the ACU GUI (on page 19)."
2. Select a controller from the **Controller/Device** menu.
The **Configuration** screen appears.



3. Click **Enable HP SmartCache**.



4. Select a solid state drive type.
5. Select one or more physical drives from the list of available drives.
6. Click **Save**.
Some features may not be available with HP SmartCache enabled.
7. With the SmartCache selected in the left panel, click **Create SmartCache for Logical Drive**.
8. Select the logical drive to cache.
9. Select the size of the cache. HP recommends 10% of the drive size, and it must be a minimum of 1 GiB.
10. Click **Save**.

The SmartCache is created for the Logical Drive.

Working with mirrored arrays

Among the advanced tasks possible with the ACU GUI, you can split a mirrored array and then recombine it. This process entails breaking a RAID 1 or RAID 1+0 mirror into two identical new arrays consisting of RAID 0 logical drives.

Support for these procedures requires the following:

- The ACU GUI must be run in offline mode ("[Accessing ACU in the offline environment](#)" on page 10).
- HP Smart Array G6 and G7 controller models must have a valid SAAP license ("[About SAAP](#)" on page 8).
- Mirrored arrays being split can have RAID 1, RAID 1+0, RAID 1 (ADM), or RAID 10 (ADM) configurations. Arrays with other RAID configurations cannot be split.

Several reasons exist for splitting and recombining a mirrored array. For more information, see the "[RAID 1\(+0\): breaking mirrors and rebuilding drives](#)" how-to white paper on the HP website (<http://h20000.www2.hp.com/bc/docs/support/SupportManual/c00378986/c00378986.pdf>).

Splitting a mirrored array

1. Run the ACU GUI in offline mode. See "Accessing ACU in an offline environment ("Accessing ACU in the offline environment" on page 10)."
2. At the **Configuration** screen, select the appropriate controller from the **Controller/Device** menu.
3. From the **Systems and Devices** tree, select the appropriate array.
4. In the **Available Tasks** panel, click **Split Mirrored Array**.
5. Click **OK**.
6. When ACU finishes splitting the array, two logical drives appear in the **Systems and Devices** tree:
 - o When a RAID 1 or RAID 1+0 array splits, two RAID 0 logical drives are created.
 - o When an array that contains a RAID 1 (ADM) logical drive splits, a RAID 1 logical drive and a RAID 0 logical drive are created.
 - o When an array that contains a RAID 10 (ADM) logical drive splits, a RAID 1+0 logical drive and a RAID 0 logical drive are created.
7. Shut down the OS.
8. Power down the server.
9. With power off, remove the physical drives that constitute one of the new arrays.

If you do not remove the physical drives for one of the arrays, the OS will be unable to distinguish between the two arrays when the server is restarted, because the arrays are identical.
10. Power up the server.
11. Restart the OS.

Recombining a split mirrored array

1. Run the ACU GUI in offline mode. See "Accessing ACU in the offline environment (on page 10)."
2. At the **Configuration** screen, select the appropriate controller from the **Controller/Device** menu.
3. From the **Systems and Devices** tree, select the array to use as the source array.
4. In the **Available Tasks** panel, click **Re-Mirror Array**.
5. Select the array to be mirrored to the source array.

This array is usually the array that was split from the original mirrored array. However, it can be any other array of the correct size.
6. Click **OK**.
7. When ACU finishes re-mirroring the array, restart the OS.

The controller uses the rebuild process to synchronize the mirrored drives. The drive online LED flashes during the rebuild process. Depending on the hard drive size and the server load, this process can take up to 2 hours. You can boot the OS during this time, but the logical drive is not fault-tolerant until the rebuild is complete.

Creating a split mirror backup

This task splits an array that consists of one or more RAID 1, RAID 1+0, RAID 1 (ADM), or RAID 10 (ADM) logical drives, and then creates two arrays: a primary array and a backup array.

To create a split mirror backup:

1. Run the ACU GUI in offline mode. See "Accessing ACU in the offline environment (on page 10)."

2. At the **Configuration** screen, select the appropriate controller from the **Controller/Device** menu.
3. From the **Systems and Devices** tree, select the appropriate array.
4. In the **Available Tasks** panel, click **Create Split Mirror Backup**.
A verification and message dialog box appears.
5. Click **OK**.
ACU creates the array according to the following rules:
 - If the original array contained RAID 1 or RAID 1+0 drives, then the primary array will contain RAID 0 drives.
 - If the original array contained RAID 1 (ADM) drives, the primary array will contain RAID 1 drives.
 - If the original array contained RAID 10 (ADM) drives, the primary array will contain RAID 1+0 drives.
 - The backup array always contains RAID 0 logical drives.
 - The primary array continues to be fully accessible to the operating system while the backup array is hidden from the operating system.
6. When ACU finishes creating the split mirror backup, the new backup array appears in the **Systems and Devices** tree:
The array includes the designation "Backup" at the beginning of the array name.

Re-mirroring, rolling back, or re-activating a split mirror backup

1. Run the ACU GUI in offline mode. See "Accessing ACU in the offline environment (on page 10)."
2. At the **Configuration** screen, select the appropriate controller from the **Controller/Device** menu.
3. From the **Systems and Devices** tree, select the primary array.
4. In the **Available Tasks** panel, click **Manage Split Mirror Backup**.
5. Select one of the following actions:
 - Re-mirror the array and preserve the existing data. Discard the backup array.
This option re-creates the original mirrored array with the current contents of the primary array.
 - Re-mirror the array and roll back to the contents of the backup array. Discard existing data.
This option re-creates the mirrored array but restores its original contents, which are in the backup array. HP recommends that you do not perform this option under the following circumstances:
 - In an online environment
 - If the logical drive to be rolled back is mounted
 - If the logical drive to be rolled back is in use by the operating system
 - Activate the backup array.
This option makes the backup array fully accessible to the operating system. ACU removes the designation "backup" from the name of the array.

Healing an array

The Heal Array operation enables you to enter a command to replace failed physical drives in the array with healthy physical drives. After replacement, the original array and logical drive numbering is unaffected.

To use Heal Array, you must meet the following conditions:

- The array has at least one failed drive.
- The array is not transforming (for example, rebuilding to a spare).
- The array has a working cache, making it capable of transformation.
- The replacement physical drives and the original drives must be of the same interface type, such as SAS, SATA, and so on.
- A sufficient number of unassigned physical drives of the correct size are available to replace each failed physical drive in the array.
The correct size is defined as a drive as large as the smallest drive on the array, but no larger than the smallest spare.
- If using an HP ProLiant Smart Array G6 or G7 controller, an SAAP 1.0 license is activated.

When you select Heal Array, and all conditions are met, one of the following actions occurs to the volume in the array:

- If a volume has failed, ACU recreates the volume. This action occurs with a RAID 0 volume.
- If a volume is degraded, ACU rebuilds the volume.

Replacing an array

Some controllers may not support this option or might require a license key to enable the feature.

ACU enables you to transfer the contents of an array to an existing empty array or a new array. During this operation, all logical drives transfer from the original array to the destination array. The original array is deleted, and the drives that were being used are freed and listed as unassigned drives.

Replacing an array is a time-consuming process for two reasons: all data in each logical drive is copied to the destination array, and the controller performs all data transformations while servicing IO requests to other logical drives.

To perform the Replace Array operation, you must meet the following conditions:

- The destination array must have the same number of physical drives as the source or original array.
- Both the source and destination arrays must be in OK state.
- All existing logical drives in the source array must be in OK state.
- The destination array must have sufficient capacity to hold all logical drives present in the source array.

Diagnostics tasks

The ACU Diagnostics feature replaced the Array Diagnostic Utility supported by SmartStart v8.20 and earlier. Beginning with ACU v8.70.9, the diagnostics functionality within ACU has expanded and includes the SmartSSD Wear Gauge.

ACU generates the following reports and logs:

- Array diagnostic report

This report contains information about all devices, such as array controllers, storage enclosures, drive cages, as well as logical, physical, and tape drives. For supported solid state drives, this report also contains SmartSSD Wear Gauge information.

- SmartSSD Wear Gauge report

This report contains information about the current usage level and remaining expected lifetime of solid state drives attached to the system.

- Serial output logs

This log details the serial output for the selected controller.

For each controller, or for all of them, you can select the following tasks:

- View Diagnostic Report
- Generate Diagnostic Report
- View Serial Output Logs
- View SmartSSD Wear Gauge Report
- Generate SmartSSD Wear Gauge Report

For the view tasks, ACU generates and displays the report or log. For the generate tasks, ACU generates a report without the graphical display.

For either task, you can save the report. In the SmartStart environment (G7 or earlier servers), you can save the report to a formatted diskette or USB device. In online and offline environments, ACU saves the diagnostic report to a compressed folder, which contains an XML report, a plain text report, and a viewer file so you can display and navigate the report through a web browser.

Each ACU Diagnostics report contains a consolidated view of any error or warning conditions encountered. It also provides detailed information for every storage device, including the following:

- Device status
- Configuration flags
- Firmware version numbers
- Physical drive error logs

ACU Diagnostics never collects information about the data content of logical drives. The diagnostic report does not collect or include the following:

- File system types, contents, or status
- Partition types, sizes, or layout
- Software RAID information
- Operating system device names or mount points

Performing a Diagnostics task

1. Open ACU.

For more information, see "Using the ACU GUI (on page 19)."

If ACU is already open, click the **Diagnostics/SmartSSD** tab.

The **Diagnostics/SmartSSD** screen appears with the following selections:

- **View SmartSSD Wear Gauge Report**

- **Generate SmartSSD Wear Gauge Report**
- **Run Array Diagnostic Reports**

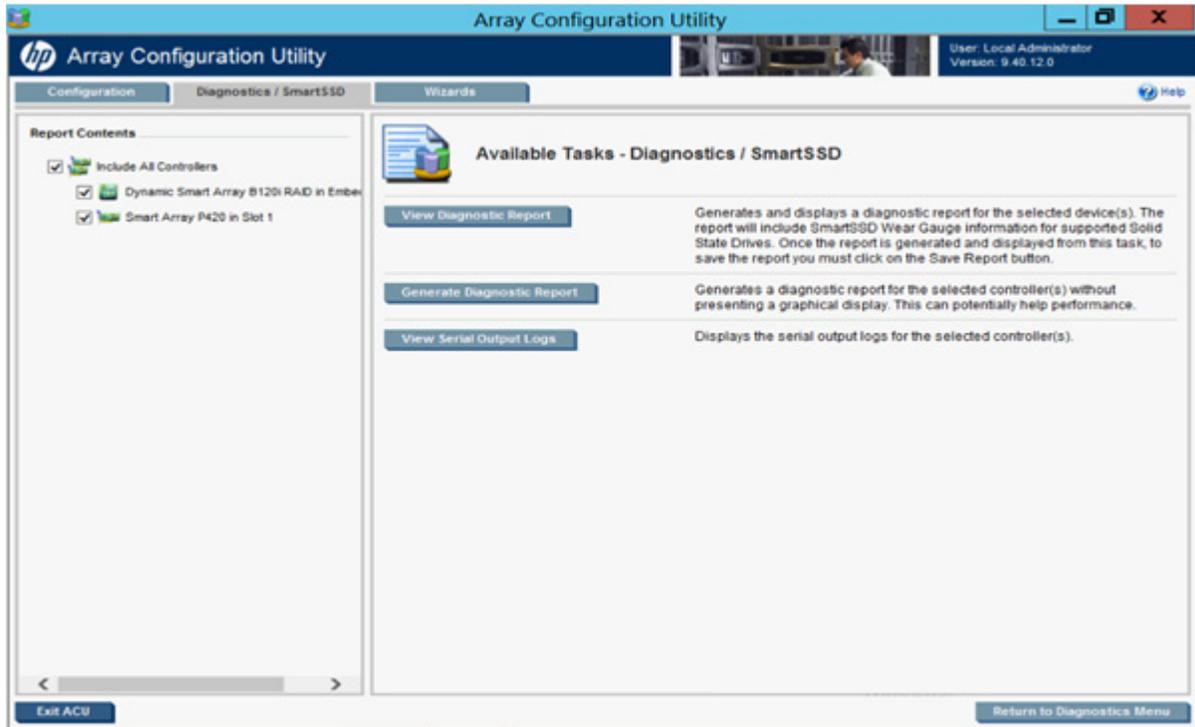
For this example, use the diagnostic reports selection.

2. Select **Run Array Diagnostic Reports**.

The array diagnostic report screen appears.

3. Select a device or devices from the **Report Contents** panel.

The **Available Tasks** panel appears. For more information, see "Diagnostics tasks (on page 39)."



4. Click one of the task buttons:

- If you select **Generate Diagnostic Report**, wait for the report to generate, and then click **Close Report** or **Save Report**.
- If you select **View Diagnostic Report**, the report appears. When you are finished viewing the current report, click **Close Report**, **Refresh Report**, or **Save Report**.
- If you select **View Serial Output Logs**, the log appears. When you are finished viewing the log, click **Close Logs**.

For more information about reports and viewing them in a browser, see the following sections:

- "Reported information (on page 92)"
- "Identifying and viewing diagnostic report files (on page 99)"
- "Identifying and viewing SmartSSD Wear Gauge report files (on page 101)"

Wizards

From the Wizards screen, you can perform tasks related to controllers, arrays, physical drives, and logical drives.

For certain tasks, the controller must have SAAP activated by a registered license key. For more information, see "About SAAP (on page 8)."

When a controller is selected, the wizards that appear are a subset of the total number of wizards that are possible for the selected controller. ACU lists or omits wizards based on the controller model and configuration. For example, if the selected controller has no unassigned physical drives, Expand Array is not an available wizard.

Wizards are available by selecting the root controller in the Systems and Devices tree. The following list includes all possible wizards:

- Add Logical Drive
- Controller Settings
- Expand Array
- Express Configuration
- Manage Spare Drive
- Migrate Logical Drive
- Move Array
- Remove Logical Drive

For more information, see "Using Wizards (on page 42)."

The Express Configuration wizard operates differently. See "Using Express Configuration (on page 44)."

Using Wizards

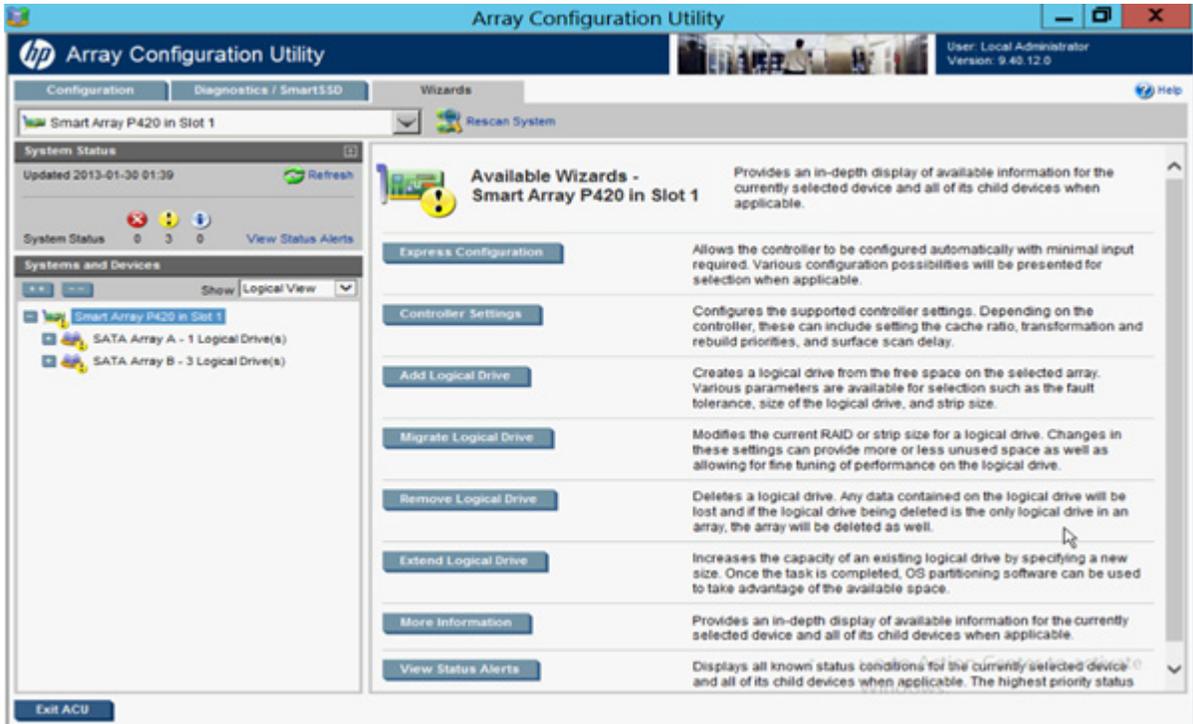
1. Open ACU.

For more information, see "Using the ACU GUI (on page 19)."

If ACU is already open, click the **Wizards** tab.

2. Select a controller from the **Controller/Device** menu.

The **System Status**, **Systems And Devices**, and **Available Wizards** panels appear. The listed wizards are available for this device in its current configuration. For more information, see "Wizards (on page 41)."



3. Click a wizard button.

A list of all possible options for that wizard appears on the right side of the screen, replacing the wizard list.



4. Select the settings or configuration options for the device.
5. Use the **Next** and **Back** buttons to navigate multiple screens of options.
6. Click **Save** or **OK**.

Using Express Configuration

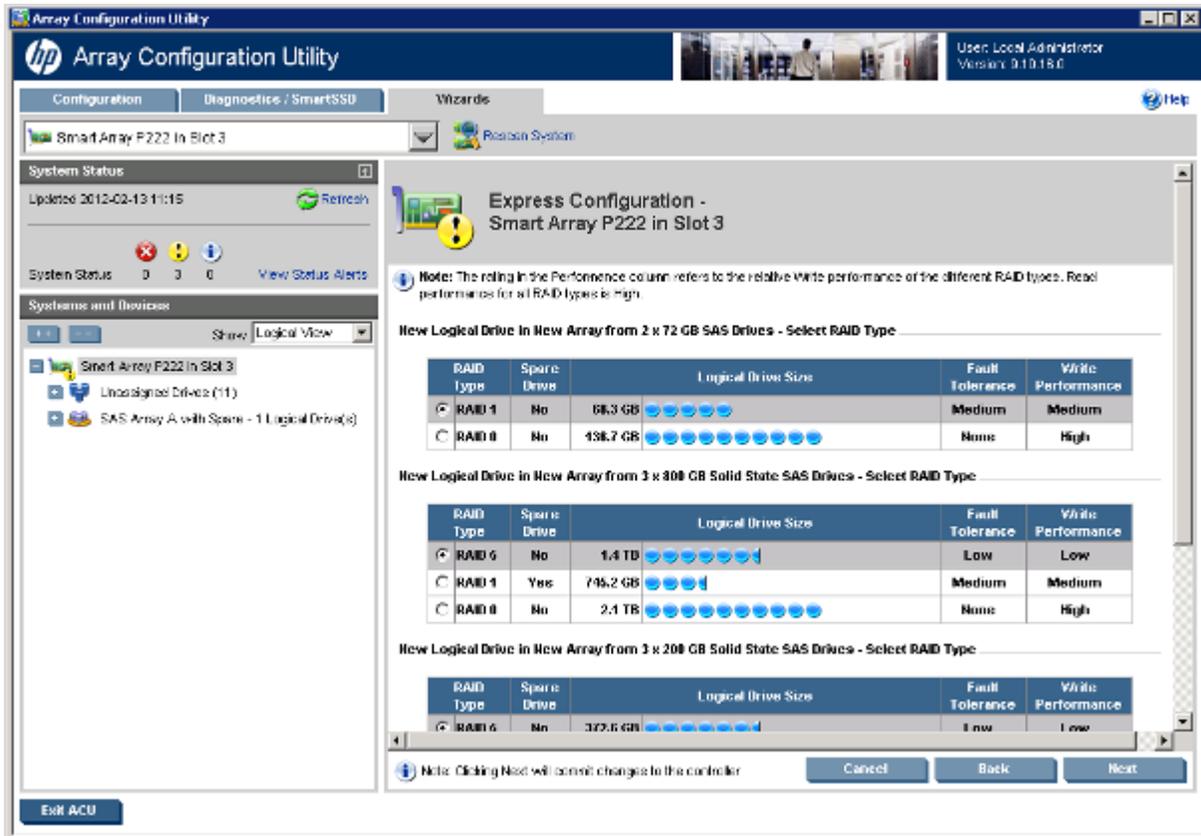
Express Configuration is a wizard that creates the optimum number of arrays and logical drives from all of the physical drives that are attached to the controller.

This wizard is available only under two conditions: when an array on the selected controller contains unused drive space, or when physical drives are connected to the controller, but they are not assigned to an array.

To use Express Configuration:

1. Open ACU.
For more information, see "Using the ACU GUI (on page 19)."
If ACU is already open, click the **Wizards** tab.
2. Select a controller from the **Controller/Device** menu.
The **System Status**, **Systems And Devices**, and **Available Wizards** panels appear. The listed wizards are available for this device in its current configuration. For more information, see "Wizards (on page 41)."
3. Click **Express Configuration**.

ACU displays possible logical drive configurations based on groupings of drives with the same capacity.



4. For each logical drive, select a RAID Type.
ACU identifies possible RAID configurations and lists spare drive capability, the size of the logical drive, fault tolerance, and write performance.
5. Click **Next**.
The wizard configures the arrays and logical drives.
6. When the configuration is complete, click **Finish**.

Using the ACU CLI

Access ACU with one of the many methods available:

- Accessing ACU in the offline environment (on page 10)
- Accessing ACU in the online environment (on page 15)

Some ACU CLI features are only available in the offline environment, such as setting the boot controller and boot volume.

ACU CLI has two operating modes:

- Console mode ("Opening the CLI in Console mode" on page 46)
You can adjust several configuration parameters on several devices without having to restart ACU each time.
- Command mode ("Opening the CLI in Command mode" on page 46)

You can make an isolated change of just one configuration parameter on one device.

For VMware ESXi 5.0, ACU CLI works similarly to the Command mode mentioned. However, slight differences exist, requiring you to run ACU CLI through the VMware vSphere esxcli command-line utility. For more information about using HP ACU CLI for ESXi 5.0, see ACU CLI release notes or the HP website (<ftp://ftp.hp.com/pub/softlib2/software1/pubsw-linux/p414707558/v68034/hpacucli-9.0-24.0.noarch.txt>).

Opening the CLI in Console mode

The syntax of the command required to open the ACU CLI in Console mode depends on the operating system that you are using.

- For Microsoft® Windows®, enter the following text:
`C:\Program Files\Compaq\Hpacucli\Bin\hpacucli.exe`
Alternatively, click **Start**, and select **Programs>HP System Tools>HP Array Configuration Utility CLI>HP Array Configuration Utility CLI**.
- For Linux, enter the following text:
`[root@localhost root]# hpacucli`

After you have entered Console mode in either operating system, the screen displays the following message and console prompt:

```
HP Array Configuration Utility CLI 7.15.17.0
Detecting Controllers...Done.
Type "help" for a list of supported commands.
Type "exit" to close the console.
=>
```

The remaining examples in the ACU CLI section of this guide are described as if entered in Console mode.

Opening the CLI in Command mode

To use Command mode, identify the appropriate ACU CLI command ("The `<command> variable`" on page 47). Then, add the command to the end of the text line that is used for opening the CLI in Console mode (on page 46).

The following examples use `help` as the command:

- Using Microsoft® Windows®:
`C:\Program Files\Compaq\Hpacucli\Bin\hpacucli.exe help`
- Using Linux:
`[root@localhost root]# hpacucli help`

The remaining examples in the ACU CLI section of this guide are described as if entered in Console mode.

CLI syntax

Whether entered in Command mode or Console mode, a typical ACU CLI command line consists of three parts: a target device, a command, and a parameter with values if necessary. Using angle brackets to denote a required variable and plain brackets to denote an optional variable, the structure of a typical ACU CLI command line is as follows:

```
<target> <command> [parameter=value]
```

The <target> variable

This variable provides the path to the device that you want to configure. The device can be a controller, an array, a logical drive, or a physical drive. The syntax used is as follows:

```
controller all | slot=# | wwn=# | chassisname="AAA" | serialnumber=# |
chassisserialnumber=# | [array=all|<id>] [logicaldrive all|#]
[physicaldrive all|allunassigned|[#:]:#:#,[:]:#:#...|[#:]:#:#-[#:]:#:#]
```

For example:

```
controller slot=3
controller wwn=500805F3000BAC11
controller slot=2 array A
controller chassisname="A" array B logicaldrive 2
controller chassisname="A" physicaldrive 1:0
controller all
controller slot=2 array all
controller slot=3 physicaldrive 1:2-1:5
```

The <command> variable

The <command> variable can be any of the following words or phrases, corresponding to typical configuration tasks:

```
add
create
delete
modify
remove
set target
```

Nonconfiguration commands are also available:

```
diag ("Generating a diagnostic report" on page 53)
help ("The help command" on page 51)
rescan ("Rescanning the system" on page 54)
shorthand ("Keyword abbreviations" on page 48)
show ("The show command" on page 49)
version ("Displaying the current versions of the application layers" on page 51)
```

A command often requires a parameter, and some parameters require a value, but the specific parameters and values that are valid for a given command depend on the target to which you apply the command.

To determine the valid parameters and values for a particular combination of target and command variables, you can query the device ("[Querying a device](#)" on page 47). You can also inspect the example commands in other sections of this guide.

Querying a device

If you do not know what values a parameter can have, you can sometimes query the device to find out by entering a ? as the value of the parameter.

Example command:

```
=> ctrl ch="Lab4" ld 1 modify raid=0 ss=?
```

A typical screen response in this case could be:

```
Available options are:
```

8
 16 (current value)
 32
 64
 128 (default)
 256

To determine which parameters can be queried, use the help feature of the CLI ("The help command" on page 51).

Hiding warning prompts

When you enter a command for an operation that can potentially destroy user data, the CLI displays a warning and prompts you for input (a `y` or an `n`) before continuing the operation. This situation is undesirable when running batch file scripts. To prevent warning prompts from being displayed, use the term `forced` as a parameter.

Example command:

```
ctrl ch="Lab4" ld 1 delete forced
```

Under VMware ESXi 5.0, the `forced` flag is required for any command that has the option for a forced flag. If the `forced` parameter is not used, the system generates an error message notifying the user that the command could not be completed without this flag.

Keyword abbreviations

Several commonly used keywords in the ACU CLI have acceptable abbreviations, as shown in the following table. For a complete list of abbreviations, enter `help shorthand` in the ACU CLI.

Keyword	Abbreviation in ACU CLI	Keyword	Abbreviation in ACU CLI
adapterid	ai	nobatterywritecache	nbwc
allunassigned	all	numberparitygroups	npg
arrayaccelerator	aa	parallelscsi	ps
acheratio	cr	physicaldrive	pd
chassisname*	ch*	postprompttimeout	ppto
chassisserialnumber	csn	preferredpathmode	ppm
chassisslot	chs	queuedepth	qd
configurationmode	cm	raid1writebuffering	r1wb
connectionname	cn	rebuildpriority	rp
controller	ctrl	redundantcontroller	rc
ctrlpath	cp	serialnumber	sn
degradedperformancemode	dpo	spareactivationmode	sam
drivetype	dt	ssdphysicaldrive	ssdpd
drivewritecache	dwc	stripesize	ss
elevatorsort	es	surfaceanalysiseventnotify	saen
enclosure	enc	surfacescandelay	ssd
exitonerror	eo	surfacescanmode	ssm
expandpriority	ep	surfacescanpriority	sp
inconsistencyrepairpolicy	irp	surfacescanschedule	sss

Keyword	Abbreviation in ACU CLI	Keyword	Abbreviation in ACU CLI
licensekey	lk	tapedrive	td
logicaldrive	ld	waitforcacheroom	wfcr
mnpdelay	mnpd	—	—

*The CLI also uses this keyword and abbreviation for the terms **box name** and **RAID array ID**.

The show command

The `show` command enables you to obtain information about a device.

Syntax:

```
<target> show [detail][[status]
```

When you specify a target that consists of several devices, the information in the output is normally less comprehensive than when you specify only one device as the target. You can use the `[detail]` parameter in this case to retain all the information usually given for individual devices.

An extra parameter is available for controller targets: `config`. This parameter is used as follows:

```
<target controller> show config [detail]
```

If you use the `config` parameter, the output includes information about each device connected to the controller.

Example 1

```
=> ctrl slot=9 show
```

A typical output would be:

```
Smart Array P421 in Slot 9
  Bus Interface: PCI
  Slot: 9
  Serial Number: PBKTV0XTAZZ005
  RAID 6 (ADG) Status: Enabled
  Controller Status: OK
  Hardware Revision: A
  Firmware Version: 0.02-106
  Rebuild Priority: Medium
  Expand Priority: Medium
  Surface Scan Delay: 3 secs
  Surface Scan Mode: Idle
  Queue Depth: Automatic
  Monitor and Performance Delay: 60 min
  Elevator Sort: Enabled
  Degraded Performance Optimization: Disabled
  Inconsistency Repair Policy: Disabled
  Wait for Cache Room: Disabled
  Surface Analysis Inconsistency Notification: Disabled
  Post Prompt Timeout: 15 secs
  Cache Board Present: True
  Cache Status: OK
```

```
Accelerator Ratio: 10% Read / 90% Write
Drive Write Cache: Disabled
Total Cache Size: 1024 MB
Total Cache Memory Available: 816 MB
No-Battery Write Cache: Disabled
Cache Backup Power Source: Capacitors
Battery/Capacitor Count: 1
Battery/Capacitor Status: OK
SATA NCQ Supported: True
Spare Activation Mode: Activate on drive failure
Controller Temperature (C): 40
Cache Module Temperature (C): 0
Capacitor Temperature (C): 0
Boot Controller: True
Primary Boot Volume: logicaldrive 1
Secondary Boot Volume: logicaldrive 1
```

Example 2

```
=> ctrl all show
```

Because this target consists of several devices, the output will be brief. A typical output would be:

```
MSA1000 at dog (sn: P56350D9IP903J, csn: (9J3CJN71XDCH, wwn:
500805F3000BAC11)
Smart Array 5312 in Slot 3 (sn: P4AB5X9BFMLNTJ)
Smart Array 532 in Slot 2 (sn: P44940LDAORS4F)
```

Example 3

```
=> ctrl ch="lab4" show config
```

The output in this case will have detailed information because the target consists of only one device. A typical output would be:

```
MSA1000 at dog (sn: P56350D9IP903J, csn: (9J3CJN71XDCH, wwn:
500805F3000BAC11)

array A (Parallel SCSI, Unused Space: 20091 MB)
  logicaldrive 1 (219 MB, RAID 6(ADG), OK)
    physicaldrive 1:3 (box 1:bay 3, Parallel SCSI, 4.3 GB, OK)
    physicaldrive 1:4 (box 1:bay 4, Parallel SCSI, 9.1 GB, OK)
    physicaldrive 1:5 (box 1:bay 5, Parallel SCSI, 9.1 GB, OK)
    physicaldrive 1:6 (box 1:bay 6, Parallel SCSI, 9.1 GB, OK)
    physicaldrive 1:7 (box 1:bay 7, Parallel SCSI, 9.1 GB, OK)
    physicaldrive 1:9 (box 1:bay 9, Parallel SCSI, ??? GB, failed, spare)

unassigned
  drive 1:1 (box 1:bay 1, Parallel SCSI, 36 GB, OK)
  physicaldrive 1:2 (box 1:bay 2, Parallel SCSI, 36 GB, OK)
  physicaldrive 1:8 (box 1:bay 8, Parallel SCSI, 9.1 GB, OK)
  physicaldrive 1:10 (box 1:bay 10, Parallel SCSI, 9.1 GB, OK)
```

```
physical
physicaldrive 1:11 (box 1:bay 11, Parallel SCSI, 9.1 GB, OK)
```

Displaying the current versions of the application layers

The `version` command displays the current versions of the application and other software layers.

Syntax:

```
version
```

Example command:

```
version
```

A typical output would be:

```
ACU CLI Version: 9.10.16.0
SoulAPI Version: 6.0.15.0
InfoManager Version: 6.1.14.0
```

The help command

To get help with the CLI, enter `help` at the CLI prompt, and then enter one or more help items, as follows:

```
=> help <item1> [item2] [item3]
```

A help item can be any of the following:

- A CLI command ("[The <command> variable](#)" on page 47)
- An ACU CLI keyword or keyword abbreviation ("[Keyword abbreviations](#)" on page 48)
- A CLI parameter
- A term commonly used in ACU, such as `migrate`, `extend`, or `cache`
- The word `shorthand` (gives a list of abbreviations for keywords in the CLI)

The help feature of the ACU CLI behaves like a browser search engine in that each item that you add to the help input string reduces the amount of help output text. For example, `help controller` produces extensive information, while `help controller remove` restricts the help output to information about how the `remove` command applies to controllers.

Typical procedures

The following sections describe some common ACU CLI procedures.

Setting the boot controller

This procedure is only available in the offline environment.

The boot controller is the first controller that the system examines (after power up) to find a bootable logical drive or volume. To work properly, the boot controller must have a designated boot volume. See "Setting the boot volume (on page 52)."

Syntax:

```
<target> modify [bootcontroller=enable|disable]
```

where `<target>` is a controller with a volume that can be set as the boot volume.

Example command:

```
controller slot=1 modify bootcontroller=enable
```

Setting the boot volume

This procedure is only available in the offline environment.

The boot volume is the volume that contains the OS and its support files. To work properly, the boot volume must be accessible by the designated boot controller. See "Setting the boot controller (on page 51)."

Syntax:

```
<target> modify [bootvolume=primary|secondary|none]
```

where <target> is a controller and a logical drive.

Example commands:

```
controller slot=1 ld 1 modify bootvolume=primary
controller slot=1 ld 2 modify bootvolume=secondary
controller slot=1 ld 1 modify bootvolume=none
```

Setting the target

If you must perform several operations on a given target device, you can simplify the required commands by setting the device as the default <target> for the CLI operations.

After you have set the target, any command that you enter in the CLI without a specified <target> is automatically applied to the set target. If you must also perform operations on other devices, you can still do so at any time by specifying the <target> for each of those operations as usual. You can also change the set target or clear it completely. The set target is automatically cleared when you close the CLI.



IMPORTANT: You cannot use the `set target` command in batch file scripts.

Syntax:

```
set target <target>
```

where <target> is a controller, array, or logical drive.

Example commands:

```
=> set target ctrl slot=3
=> clear target
```

Typical scenario

First, set a target as follows:

```
=> set target ctrl ch="Lab 4"
=> show target
```

```
controller chassisname="Lab 4"
```

As an example of how the `set target` command works, check the status of array A on this controller:

```
=> array A show
MSA1000 at Lab 4
  array A
    Interface Type: Parallel SCSI
    Unused Space: 7949 MB
    Status: OK
```

Note that the controller does not need to be specified because it is currently the set target.

Now clear the target, reset it, and enter a few commands for the new set target:

```
=> clear target
```

```

=> set target ctrl slot=3
=> array A add drives=1:7,1:8,1:9
=> array B add spares=1:10,1:11
=> ctrl slot=4 ld 3 modify ss=64
=> modify rp=high

```

This sequence includes a command for a different target (the controller in slot 4) as a demonstration. Note that the next command in the sequence (the one for modifying the rebuild priority) applies to the controller in slot 3, not the one in slot 4. This is because the command does not specify a <target> for the rebuild priority, so the default set target is used instead.

Identifying devices

You can enter a command that causes the LEDs on target devices to flash, enabling you to identify the devices. After an hour, the LEDs stop flashing. You can also stop LED flashing by entering the `off` command.

Syntax:

```
<target> modify led=on|off
```

Example commands:

```

=> ctrl ch="Lab 4" modify led=on
=> ctrl ch="Lab 4" array A modify led=off

```

Deleting target devices

Syntax:

```
<target> delete [forced]
```

where <target> can be a controller, array, or logical drive. Except in the case of controllers, you can delete several devices simultaneously if they are of similar type by using the `all` keyword.

Because deleting a target device can result in data loss, the screen displays a warning prompt unless you include the `forced` parameter.

Example commands:

```

=> ctrl ch="Lab 4" delete forced
=> ctrl slot=3 ld all delete

```

Generating a diagnostic report

HP provides several ways for administrators to generate diagnostic information, including all three ACU formats and the standalone HP Array Diagnostics and SmartSSD Wear Gauge Utility.

In ACU CLI, the `diag` command outputs diagnostic information about a specified controller or all controllers on the system. The `ssdrpt` option generates the SmartSSD Wear Gauge report.

Syntax:

```

<target> diag <file=filename> [ris=on|off] [ssdrpt=on|off] [xml=on|off]
[zip=on|off]

```

where the following is designated:

- <target> is a controller or all controllers.
- <file=filename> designates the target file in which the diagnostic information is saved.
- [ris=on|off] determines whether RIS information is or is not included. The `off` value is provided for backward compatibility and is ignored.

- [ssdrpt=on|off] specifies to generate or not generate the Smart SSD Wear Gauge report. The default is off.
- [xml=on|off] outputs diagnostic information in formatted XML. The off value is provided for backward compatibility and is ignored.
- [zip=on|off] compresses the output to a zipped file. Default behavior is uncompressed. The off value is provided for backward compatibility and is ignored.

Example commands:

```
ctrl all diag file=c:\allcontrollers.zip
ctrl slot=4 diag file=c:\ctrl_slot4.zip
ctrl ch="mybox" diag file=mybox.zip ssdrpt=on
```

Erasing a physical or logical drive

Syntax:

```
<target> modify [erase
erasepattern=zero|random_zero|random_random_zero] [deleteaftererase=yes|no]
```

where the target can be any valid physical drive or logical drive. The option to delete the target after erasing it is no longer supported. Although [deleteaftererase=yes|no] can still be entered, it is valid only if the target is a logical drive, and the logical drive will always be erased.

To stop an erase process at any time, enter the stoperase command.

Example commands:

```
=> ctrl slot=3 ld 2 modify erase erasepattern=zero
=> ctrl slot=4 ld all modify erase erasepattern=random_zero
=> ctrl slot=3 ld 2 modify stoperase
```

Rescanning the system

A rescan detects devices that have been added to the system since the previous rescan or since the ACU CLI was started, whichever is more recent.

Syntax:

Use the word rescan directly at the ACU CLI prompt, without any target device or parameters.

Example command:

```
=> rescan
```

Entering or deleting a license key

Some advanced configuration tasks (available only on certain controller models) can be performed only after software is installed on the controller and a license key is registered to activate the software.

Syntax:

```
<target> add [lk=xxxxx-xxxxx-xxxxx-xxxxx-xxxxx]
```

where the target is any valid controller. The hyphens are optional.

To delete a license key, use a standard delete command but use the license key (not the controller) as the target:

```
<target> delete
```

Example commands:

```
=> ctrl slot=5 lk=12345-65432-78787-43434-24680 delete
```

```
=> ctrl slot=4 add lk=9876543210222224444466666
```

Optimizing controller performance for video

On some controller models, you can optimize the controller performance for video.

To use this features on an HP Smart Array G6 or G7 controller, a registered SAAP ("About SAAP" on page 8) license key is required.

Syntax:

```
<target> modify dpo=enable elevatorsort=disable irp=enable  
queuedepth=automatic mnpd=60
```

where the target is any valid controller.

The `queuedepth` parameter can also have a numerical value from 2 to 32, and the `mnpd` parameter can have any value from 0 (disabled) to 60.

To disable the video performance optimization feature, reverse the disable and enable values noted above, set the queue depth to automatic, and set `mnpd` to 0, as in the second example command.

Example commands:

```
=> ctrl slot=5 modify dpo=enable elevatorsort=disable irp=enable  
queuedepth=16 mnpd=25
```

```
=> ctrl slot=3 modify dpo=disable elevatorsort=enable irp=disable  
queuedepth=automatic mnpd=0
```

Creating a logical drive

Syntax:

```
<target> create type=ld [parameter=value]
```

`<target>` is usually a controller, but it can be an array if you are creating an additional logical drive on an existing array.

If you want to create a logical drive on a group of physical drives that are not yet assigned to an array, you do not have to build the array first. In the CLI, unlike in the GUI, the array is created automatically at the same time as the logical drive.

The standard parameters used when creating a logical drive are described in the following table. If you do not specify a particular parameter, the CLI uses the appropriate default value.

Parameter	Acceptable values	Comments
drives	[#:]#:#, [#:]#:#, ... [#:]#:#-[#:]#:# all allunassigned	The default setting is all.
raid	0 1 1adm 1+0 1+0adm 5 50 6 60 ?	The default setting is the highest RAID level that the controller model and number of drives can support, except 50 or 60 (which must be specified deliberately).
numberparitygroups	2 #	The default value is 2.
ss	8 16 32 64 128 256 512 1024 default ?	Units are KB.* The default setting depends on the RAID level.

Parameter	Acceptable values	Comments
size	# min max maxmbr ?	This parameter determines the desired size of the logical drive. Units are MB.* The default setting is max.
sectors	32 63 default ?	The default setting depends on the operating system.
aa	enable disable ?	The default setting is enable.
drivetype	sas satalogical sata saslogical parallel scsi ss_sas ss_sata ?	—

*Use only these units. Do not enter any extra text in the command to specify the units.

If you specify an array as the target, you can omit the `drives` parameter, because the drives are already implicitly defined by the array ID. This feature is useful if you are entering the command directly into the CLI console because you do not need to remember which drives belong to the array. When you write a batch file, however, it is often easier to specify every drive in the array than to parse out the array ID.

When you use the `drives` parameter you can list each drive individually, you can specify a drive range, or you can specify both a range and some individual drives. A drive range can span ports, boxes, and bays. If you specify individual drives, they do not have to form a continuous sequence. If you specify a range, the CLI automatically excludes from the target any drive in the range that is unavailable (for example, a drive is excluded if it already belongs to an array, is a spare, has too small a capacity, or has failed).

If you want to specify an existing array by its drives rather than by array ID, then all of the drives that you specify must belong to the same array, and none of the drives in the array can be omitted.

Example commands:

```
ctrl slot=5 create type=ld drives=1:0,1:1,1:3 raid=adg
ctrl slot=5 create type=ld drives=1:1-1:3 raid=adg
ctrl slot=5 create type=ld drives=1:7,1:10-2:5,2:8-2:12 raid=adg
ctrl slot=5 array A create type=ld size=330 raid=adg
```

The following pair of commands demonstrates how the `drives` parameter can be used in a batch file to create two logical drives on the same array, one of 330 MB and the other of 450 MB:

```
ctrl slot=2 create type=ld drives=1:1-1:6 size=330 raid=adg
ctrl slot=2 create type=ld drives=1:1-1:6 size=450 raid=5
```

Sample scenario

Consider a situation in which you want to create two arrays. One of these arrays needs two logical drives, while the other needs only one.

First, determine what physical drives are available and what their properties are:

```
=> ctrl ch="Lab 4" pd all show
```

For this sample scenario, the screen response is:

```
MSA1000 at Lab 4
unassigned
    physicaldrive 1:12 (box 1:bay12, Parallel SCSI, 36.4 GB, OK)
    physicaldrive 1:13 (box 1:bay13, Parallel SCSI, 9.1 GB, OK)
    physicaldrive 1:14 (box 1:bay14, Parallel SCSI, 9.1 GB, OK)
```

Knowing this information, you can now create the first array with one logical drive:

```
=> ctrl ch="Lab 4" create type=ld drives=1:12
```

Now, verify that the array has been created:

```
=> ctrl ch="Lab 4" pd all show
```

In this case, the screen response is:

```
MSA1000 at Lab 4
array A
  physicaldrive 1:12 (box 1:bay12, Parallel SCSI, 36.4 GB, OK)
unassigned
  physicaldrive 1:13 (box 1:bay13, Parallel SCSI, 9.1 GB, OK)
  physicaldrive 1:14 (box 1:bay14, Parallel SCSI, 9.1 GB, OK)
```

The second array is to be created on the two remaining physical drives. Before creating this array, determine what RAID options are available for these drives:

```
=> ctrl ch="Lab 4" create type=ld drives=1:13,1:14 size=300 raid=?
```

The response in this case is:

```
Available options are:
0
1+0 (default value)
```

Now, create the new array:

```
=> ctrl ch="Lab 4" create type=ld drives=1:13,1:14 size=300 raid=1+0
```

It is not strictly necessary to specify the RAID level in this example because it is the highest possible level for this scenario and will be used by default. However, it is included in the command as an example.

Now, verify that the array has been formed:

```
=> ctrl ch="Lab 4" pd all show
```

The screen response is:

```
MSA1000 at Lab 4
array A
  physicaldrive 1:12 (box 1:bay12, Parallel SCSI, 36.4 GB, OK)
array B
  physicaldrive 1:13 (box 1:bay13, Parallel SCSI, 9.1 GB, OK)
  physicaldrive 1:14 (box 1:bay14, Parallel SCSI, 9.1 GB, OK)
```

To create a second logical drive on array B, you can specify the array (method A) or you can specify every physical drive in the array (method B).

```
=> ctrl ch="Lab 4" array B create type=ld size=900 (method A)
=> ctrl ch="Lab 4" create type=ld drives=1:13,1:14 size=900 (method B)
```

Finally, verify that the logical drives have all been created correctly:

```
=> ctrl ch="Lab 4" ld all show
MSA1000 at Lab 4
array A
  logicaldrive 1 (33.9 GB, RAID 0, OK)
array B
  logicaldrive 2 (298 MB, RAID 1+0, OK)
  logicaldrive 3 (896 MB, RAID 1+0, OK)
```

Moving a logical drive

ACU now enables you to move a single logical drive from one array to another array. This feature is available on Gen8 Smart Array controllers.

Viewing SSD physical drives

The `ssdphysicaldrive` keyword is a valid target for the `show` command, and it outputs the solid state physical drives on the specified controller.

Syntax:

```
<target> ssdpd all show [detail]
```

where the target is any valid controller.

Example commands:

```
controller slot=5 ssdpd all show
controller slot=5 ssdpd all show detail
```

Viewing SSD information

The `ssdinfo` keyword provides a summary for solid state drives for the specified controller.

Syntax:

```
<target> show ssdinfo [detail | summary]
```

Command examples:

```
controller slot=1 show ssdinfo
controller all show ssdinfo
controller all show ssdinfo detail
controller all show ssdinfo summary
```

Smart Caching in HPACUCLI

HP Smart Caching allows a caching logical drive to be created on a solid state physical drive, to be associated with an existing data logical drive to provide a performance increase.

Command Examples:

To create a new Smart Caching array for existing data logical drive 1 using the solid state drive 1e:1:10:

```
=> ctrl slot=1 create type=ldcache drives=1e:1:10 dataId=1
```

The command above creates a new Smart Caching array, for example, "array B." Only one Smart Caching array may exist on a controller. This example creates an additional caching logical drive using the existing Smart Caching array for existing data logical drive 2:

```
=> ctrl slot=1 array B create type=ldcache dataId=2
```

Rapid Parity Initialization methods

RAID levels that use parity (RAID 5, RAID 6 (ADG), RAID 50, and RAID 60) require that the parity blocks be initialized to valid values. Valid parity data is required to enable enhanced data protection through background surface scan analysis and higher performance write operations. Two initialization methods are available:

- **Default** – Initializes parity blocks in the background while the logical drive is available for access by the operating system. A lower RAID level results in faster parity initialization.
- **Rapid** – Overwrites both the data and parity blocks in the foreground. The logical drive remains invisible and unavailable to the operating system until the parity initialization process completes. All parity groups are initialized in parallel, but initialization is faster for single parity groups (RAID 5 and RAID 6). RAID level does not affect system performance during rapid initialization.

Rapid Parity Initialization is available only for supported controllers and in arrays composed of supported physical drives.

Example: of creating a logical drive with Rapid Parity Initialization:

```
=> ctrl slot=3 create type=ld drives=1e:1:1-1e:1:4  
parityinitializationmethod=rapid
```

Assigning a chassis name to the controller

If a controller is configured with at least one logical drive, you can give the controller a simplified name (the chassis name) to make it easier to identify and enter the correct controller in a command.

Syntax:

```
<target> modify ch="new chassis name"
```

where <target> is a controller. If you change the chassis name of a controller that you have set as the default target ("[Setting the target](#)" on page 52), you must reset the target.

Example commands:

```
=> ctrl sn=P56350D9IP903J modify ch="Lab 6"  
=> ctrl ch="Lab 4" modify ch="Lab 6"
```

Managing spare drives

Assigning one or more online spare drives to an array enables you to postpone replacement of faulty drives. However, it does not increase the fault-tolerance level of any logical drives in the array. For example, a logical drive in a RAID 5 configuration suffers irretrievable data loss if two of its physical drives are simultaneously in a failed state, regardless of the number of spare drives assigned to the array.

Any drive that you want to use as a spare must meet the following criteria:

- It must be an unassigned drive or a spare for another array.
- It must be of the same type as existing drives in the array (for example, SATA or SAS).
- It must have a capacity no less than that of the smallest drive in the array.

Syntax:

```
<target> add spares=[#:]#:#, [#:]#:#, [#:]#:#-[#:]#:#, ... |allunassigned  
[forced]
```

```
<target> remove spares=[#:]#:#, [#:]#:#, [#:]#:#-[#:]#:#, ... |all
```

where <target> is an array (or logical drive, if the array contains only one logical drive). The `forced` parameter represses any warning message prompts. If you specify a drive range, any drives in the range that do not meet the previous criteria are not used.

Example commands:

```
=> ctrl slot=3 array B add spares=1:6  
=> ctrl slot=4 array all add spares=1:5,1:7  
=> ctrl slot=5 array A add spares=1:1-1:5  
=> ctrl slot=5 array A remove spares=1:1-1:5
```

Setting the spare activation mode

The spare activation mode feature enables the controller firmware to activate a spare drive under the following conditions:

- When a data drive reports a predictive failure (SMART) status

- When a data drive fails; this mode is the default.

In normal operations, and for older controllers, the firmware starts rebuilding a spare drive only when a data drive fails. With the predictive failure activation mode, rebuilding can begin before the drive fails, reducing the likelihood of data loss that could occur if an additional drive fails.

Use the `spareactivationmode` keyword to toggle the spare activation mode for the controller between drive failure and drive predictive failure.

Syntax:

```
<target> modify spareactivationmode=[ failure | predictive ]
```

Example commands:

```
controller slot=1 modify spareactivationmode=predictive
controller slot=1 modify spareactivationmode=failure
```

Spare Management Mode in HPACUCLI

The `sparetype` keyword allows you to specify the spare type as "dedicated" (default) or "autoreplace." A dedicated spare temporarily takes over for a failed drive and can be shared between arrays. An autoreplace spare replaces a failed drive and cannot be shared between arrays.

Command example:

To add a spare with spare type of autoreplace:

```
=> ctrl slot=1 array A add spares=1e:1:5 sparetype=autoreplace
```

Expanding an array

You can increase the storage space on an array by adding physical drives. Any drive that you want to add must meet the following criteria:

- It must be an unassigned drive.
- It must be of the same type as existing drives in the array (for example, SATA or SAS).
- It must have a capacity no less than that of the smallest drive in the array.



IMPORTANT: An array expansion, logical drive extension, or logical drive migration takes about 15 minutes per gigabyte. While this process is occurring, no other expansion, extension, or migration can occur simultaneously on the same controller. Controllers that do not support a battery-backed write cache do not support this process.

Syntax:

```
<target> add drives=[#:]#:#,[#:]#:#,[#:]#:#-[#:]#:#,...|allunassigned
[forced]
```

where `<target>` is an array (or a logical drive, if the array contains only one logical drive). The `forced` parameter represses any warning message prompts. If you specify a drive range, any drives in the range that do not meet the previous criteria are not used.

If you add an odd number of drives to an array that contains a RAID 1+0 logical drive, you are prompted to convert the RAID 1+0 logical drive to RAID 5 or RAID 6 (ADG). Adding the `forced` parameter to the command prevents this prompt from appearing.

Example commands:

```
=> ctrl slot=3 array A add drives=1:0,1:1
=> ctrl slot=4 ld 1 add drives=allunassigned
=> ctrl slot=5 array A add drives=1:1-1:5
```

Shrinking an array

Some controllers may not support this option or may require a license key to enable this feature.

You can shrink the size of an array by removing a drive from an existing array. Observe the following criteria:

- After the shrink, the array must have sufficient capacity to contain all of the configured logical volumes.
- You may not remove drives from the array if the resulting number of drives does not support the fault tolerance (RAID level) of any existing logical drive. For example, if you have an array with four physical drives and a RAID 5 logical drive, you can remove, at most, one drive, because RAID 5 requires at least three physical drives.
- If the array contains a RAID 1+0 logical drive, you can remove only an even number of drives.
- If the array contains a compound RAID (RAID 50 or RAID 60) logical drive, you can remove drives only in multiples of the number of parity groups. For example, an array with 10 physical drives and a RAID 50 logical drive can be shrunk by removing two or four drives only.

Syntax:

```
<target> remove drives=[#:]#:#-[#:]#:#
```

where <target> is an array, and the specified physical drives are being removed to shrink the array.

For example, in an existing array (array a), six drives (1e:1:4-1e:1:9) are in use. With all criteria met, you can shrink the array to four drives by removing the last two drives with the command: <array a> remove drives=1e:1:8-1e:1:9

Example commands:

```
=> array a remove drives=1e:1:12-1e:1:14  
=> array b remove drives=1c:1:6-1c:1:7
```

Moving an array

Some controllers may not support this option or may require a license key to enable this feature.

You can move an array by designating different physical drives for the array. To move the array, each of the physical drives where the array will reside must meet the following criteria:

- It must be an unassigned drive.
- It must be of the same type as the physical drives currently in the source array (for example, SATA or SAS).
- The destination drives must have sufficient capacity to hold all the logical drives present in the source array.

Like array creation and expansion, the useable space in all drives is reduced to the size of the smallest physical drive in the destination disk set.

Moving an array automatically removes any previously assigned spare drives. If spares are assigned to the existing array, they must be designated for the array when it is moved.

Syntax:

```
<target> modify drives=[#:]#:#-[#:]#:# spares=[#:]#:#-[#:]#:#
```

where <target> is an array, and the specified physical drives are the new destination for the array.

For example, in an existing array, three 72-GB SAS drives (1e:1:4-1e:1:6) are the source. Another drive of the same size is the spare (1e:1:9). With all criteria met, you can move the array to three different 72-GB SAS

drives by specifying the new destination drives in the command (1e:1:12-1e:1:14). To maintain the same spare drive, be sure to designate the spare drive for the moved array.

Example commands:

```
=> array a modify drives=1e:1:12-1e:1:14 spares=1e:1:9
=> array b modify drives=1c:1:6-1c:1:7
```

Replacing an array

Some controllers may not support this option or may require a license key to enable this feature.

You can replace an array by designating different physical drives for the array. To replace the array, each of the physical drives where the array will reside must meet the following criteria:

- The destination drives must all be the same type (for example, SATA or SAS), but do not have to be same type as the drives in the source array.
- A destination drive does not have to be an unassigned drive, but if it is assigned, it must be a drive assigned to the array being replaced.
- The destination drives must have sufficient capacity to hold all the logical drives present in the source array.

Like array creation and expansion, the useable space in all drives is reduced to the size of the smallest physical drive in the destination disk set.

Replacing an array automatically removes any previously assigned spare drives. If spares are assigned to the existing array, they must be designated for the array when it is moved.

Syntax:

```
<target> modify drives=[#:]#:#-[#:]#:# spares=[#:]#:#-[#:]#:#
```

where <target> is an array, and the specified physical drives are the new destination for the array.

For example, in an existing array, three 72-GB SAS drives (1e:1:4-1e:1:6) are the source. Another drive of the same size is the spare (1e:1:9). With all criteria met, you can move the array to three different 72-GB SAS or SATA drives by specifying the new destination drives in the command (1e:1:12-1e:1:14). To maintain the same spare drive, be sure to designate the spare drive for the moved array.

Example commands:

```
=> array a modify drives=1e:1:12-1e:1:14 spares=1e:1:9
=> array b modify drives=1c:1:6-1c:1:7
```

Extending a logical drive

If the operating system supports logical drive extension, you can use any unassigned capacity on an array to enlarge one or more of the logical drives on the array.



IMPORTANT: An array expansion, logical drive extension, or logical drive migration takes about 15 minutes per gigabyte. While this process is occurring, no other expansion, extension, or migration can occur simultaneously on the same controller. Controllers that do not support a battery-backed write cache do not support this process.

Syntax:

```
<target> modify size=#|max|? [forced]
```

where <target> is a logical drive.

If the operating system does not support logical drive extension, carrying out this command makes data on the logical drive unavailable. Therefore, the CLI displays a warning prompt as a safeguard in case you are using such an operating system. To prevent the prompt from appearing, use the `forced` parameter.

Example commands:

```
=> ctrl slot=3 ld 1 modify size=max
=> ctrl slot=4 ld 1 modify size=?
=> ctrl slot=3 ld 2 modify size=500 forced
```

Migrating a logical drive

This command enables you to adjust the stripe size (data block size) or RAID level of a selected logical drive. For more information, see "Selecting a RAID method (on page 113)."

Consider the following factors before performing a migration:

- For some RAID-level migrations to be possible, you might need to add one or more drives to the array.
- For migration to a larger stripe size to be possible, the array might need to contain unused drive space. This extra space is necessary because some of the larger data stripes in the migrated array are likely to be filled inefficiently.



IMPORTANT: An array expansion, logical drive extension, or logical drive migration takes about 15 minutes per gigabyte. While this process is occurring, no other expansion, extension, or migration can occur simultaneously on the same controller. Controllers that do not support a battery-backed write cache do not support this process.

Syntax:

```
<target> modify [raid=0|1+0|1|5|6|adg|?] [ss=8|16|32|64|128|256|default|?]
```

where `<target>` is a logical drive.

The following limitations apply to this command:

- You cannot simultaneously query the RAID level and the stripe size of any given logical drive.
- If you do not specify a RAID level for a query or migration, the CLI uses the existing value by default.
- If you do not specify a stripe size, the CLI uses the default stripe size value for the RAID level that you specify.

Example commands:

```
=> ctrl slot=3 ld 1 modify raid=1
=> ctrl slot=4 ld 2 modify ss=16
=> ctrl slot=2 ld 3 modify raid=5 ss=16
```

Setting the preferred path mode

The preferred path mode determines how I/O traffic to the logical drives is managed on controllers that are in an active/active configuration.

- In Automatic mode, the storage system automatically selects a suitable path for I/O traffic to each logical drive depending on the host I/O patterns at the time. Because the optimum path can vary with time, I/O traffic for any given logical drive can be directed through either controller.
- In Manual mode, all I/O traffic to a given logical drive is directed through a designated controller. In this case, you must also specify the preferred controller for each logical drive ("[Assigning a redundant controller to a logical drive](#)" on page 65).

Syntax:

```
<target> modify [preferredpathmode=automatic|manual|?]
```

where <target> is a redundant controller.

Example command:

```
controller ch="lab 3" modify ppm=manual
```

Assigning a redundant controller to a logical drive

When you have set the preferred path mode ("[Setting the preferred path mode](#)" on page 64) in a redundant system to Manual, you must use the `chassisslot` command to assign each logical drive in the system to one of the redundant controllers.

Syntax:

```
<target> modify [chassisslot=#|?]
```

where <target> is a valid logical drive on a controller that is in an active/active configuration and # denotes the chassis slot number of the redundant controller. (To obtain the chassis slot number, use the `show` command on the controller.)

Example command:

```
controller ch="lab 3" ld 1 modify chs=2
```

Disabling a redundant controller

This command disables a redundant controller that is in an Active-Standby configuration.



IMPORTANT: The redundant controller cannot be re-enabled after you have disabled it.

Syntax:

```
<target> modify redundantcontroller=disable
```

where <target> is a controller that has an enabled redundant controller.

Example command:

```
=> ctrl ch="redundant Lab4" modify rc=disable
```

Changing the Rebuild Priority setting

The Rebuild Priority setting determines the urgency with which the controller treats an internal command to rebuild a failed logical drive.

- At the low setting, normal system operations take priority over a rebuild.
- At the medium setting, rebuilding occurs for half of the time, and normal system operations occur for the rest of the time.
- At the high setting, the rebuild takes precedence over all other system operations.

If the logical drive is part of an array that has an online spare, rebuilding begins automatically when drive failure occurs. If the array does not have an online spare, rebuilding begins when the failed physical drive is replaced.

Syntax:

```
<target> modify rp=high|medium|low|?
```

where <target> is a controller.

Example command:

```
=> ctrl slot=3 modify rp=high
```

Changing the Expand Priority setting

The Expand Priority setting determines the urgency with which the controller treats an internal command to expand an array.

- At the low setting level, normal system operations take priority over an array expansion.
- At the medium setting, expansion occurs for half of the time, and normal system operations occur for the rest of the time.
- At the high setting, the expansion takes precedence over all other system operations.

Syntax:

```
<target> modify ep=high|medium|low|?
```

where <target> is a controller.

Example command:

```
=> ctrl slot=3 modify ep=high
```

Setting the surface scan mode

The `surfacescanmode` keyword sets the surface mode for the controller. The available modes are disable, high, or idle. If you specify the idle mode, you must also specify a surface scan delay value.

Syntax:

```
<target> modify [ surfacescanmode=disable | idle | high | ? ]
```

Example commands:

```
controller slot=1 modify surfacescanmode=high
controller slot=1 modify surfacescanmode=disable
controller slot=1 modify surfacescanmode=idle surfacescandelay=3
```

Changing the surface scan delay time

The setting for the surface scan delay determines the time interval for which a controller must be inactive before a surface scan analysis is started on the physical drives that are connected to it.

Surface scan analysis is an automatic background process that ensures that you can recover data if a drive failure occurs. The scanning process checks physical drives in fault-tolerant logical drives for bad sectors, and in RAID 5 or RAID 6 (ADG) configurations, it also verifies the consistency of parity data.

Syntax:

```
<target> modify ssd=#
```

where <target> is a controller and # is a number between 1 and 30. This number determines the delay time in seconds, but you do not need to include units with the command.

Example command:

```
=> ctrl sn=P56350D9IP903J modify ssd=3
```

Re-enabling a failed logical drive

If a logical drive has failed and the data on it is invalid or non-recoverable, you can re-enable the logical drive so that it can be reused. This process preserves the structure of the logical drive and merely deletes data, whereas a `delete` command applied to a logical drive deletes the logical drive structure as well as the data.

Syntax:

```
<target> modify reenable [forced]
```

Example command:

```
=> ctrl slot=3 ld 1 modify reenable forced
```

Changing the controller cache ratio

The controller cache ratio setting determines the amount of memory allocated to read and write operations. Different types of applications have different optimum settings. You can change the ratio only if the controller has a battery-backed cache (because only battery-backed cache can be used for write cache) and if there are logical drives configured on the controller.

Syntax:

```
<target> modify cr=#/#|?
```

where <target> is a controller, and #/# is the cache ratio in the format read percentage/write percentage.

Example command:

```
=> ctrl slot=3 modify cr=25/75
```

Enabling or disabling the drive cache

On controllers and drives that support physical drive write cache, you can use this command to enable or disable the write cache for all drives on the controller.



CAUTION: Because physical drive write cache is not battery-backed, you could lose data if a power failure occurs during a write process. To minimize this possibility, use a backup power supply.

Syntax:

```
<target> modify drivewritecache=enable|disable|? [forced]
```

where <target> is a controller that supports drive write cache.

Example command:

```
=> ctrl slot=5 modify dwc=enable
```

Enabling or disabling the array accelerator

If the controller has an array accelerator, you can disable or enable it for specified logical drives.

NOTE: Disabling the array accelerator for a logical drive reserves use of the accelerator cache for other logical drives on the array. This feature is useful if you want the other logical drives to have the maximum possible performance (for example, if the logical drives contain database information).

Syntax:

```
<target> modify aa=enable|disable|?
```

where <target> is a logical drive.

Example command:

```
=> ctrl slot=3 ld 1 modify aa=enable
```

Enabling a script to exit on error

When a script encounters an error during execution, the `exitonerror` keyword enables the script to decide whether to continue or to exit the application and return an exit code.

Syntax:

```
set [exitonerror=enable | disable]
```

Example commands:

```
set exitonerror=enable
set eoe=disable
show exitonerror
```

Using ACU scripting

Access ACU with one of the many methods available:

- Accessing ACU in the offline environment (on page 10)
- Accessing ACU in the online environment (on page 15)

The ACU Scripting application has two scripting modes:

- Capture mode for capturing a configuration (on page 68)
ACU inspects the configuration of all internal and external array controllers connected to the server and then writes a script file describing this configuration.
- Input mode for using an Input script (on page 68)
ACU reads the array configuration described in a specified script file. See "Creating an ACU script file (on page 69)." ACU then applies this configuration to a target system.

Capturing a configuration

To capture the configuration of a system, enter the following command at the system command line prompt:

```
hpacuseripting -c [drive:][path]OUTPUTFILENAME.ext [-internal | -external]
-e [drive:][path]ERRORFILENAME.ext
```

OUTPUTFILENAME is the name of the capture file, and *ext.* is the file extension. If you do not specify a name and location for this file, ACU uses the default name ACUOUTPUT.ini, and places the file in the ACU working directory.

The `-internal` and `-external` switches limit capture to internal or external controllers.

The `-e` switch information is used only if ACU must generate an error file. By default, ACU names the error file ERROR.ini and places it in the ACU working directory.

Using an Input script

To use an Input script to configure or reconfigure a system, first locate a suitable ACU script or see "Creating an ACU script file (on page 69)."

Then, enter the following command at the system command line prompt:

```
hpacuseripting -i [drive:][path]FILENAME.ext [-internal | -external] [-reset]
-e [drive:][path]ERRORFILENAME.ext
```

FILENAME is the name of the ACU input file, and *ext* is the file extension. If you do not specify the name and location of this file, ACU searches for ACUINPUT.ini in the ACU working directory.

The `-internal` and `-external` switches limit configuration operations to internal or external controllers.

The `-reset` flag destroys any existing data and overwrites the current configuration with the configuration specified in the script.

The `-e` switch information is used only if ACU must generate an error file. By default, ACU names the error file ERROR.ini and places it in the ACU working directory.

Creating an ACU script file

To create a valid ACU script file, use one of the following methods:

- Modify the sample custom input script (on page 69).
- Create a Capture file for capturing a configuration (on page 68).

You can create a capture file from any server that has ACU loaded, and then modify the values of options in the file as necessary for the target system. This method is useful for applying a standard configuration to several servers that have similar storage resources.

- Write an original script.

Each line of text in an ACU script file is in the format *option=value* and can be written in uppercase or lowercase letters. For information about possible option values and the minimum configuration information that a valid script must have, see the sample custom input script (on page 69).

You can add blank lines and comments to any script to make it easier to read and understand. To create a comment, enter a semicolon, and then enter the comment text. ACU ignores all text on the same line after a semicolon.

Sample custom input script

The sample script in this section gives all possible values for each option.

- If an **option** is shown in bold type, you must enter a value for that option when writing your own script.
- If a **value** is shown in bold type, ACU uses that value as a default setting when creating new logical drives.

You can use this script as a template for your own script.

```
Action = Configure|Reconfigure
```

```
Method = Custom|Auto ; COMMENT: ACU cannot create a RAID 50 or RAID 60 configuration in Auto mode. You must create such configurations manually using the Custom setting.
```

```
Controller = All | First | Slot [N][:N] | WWN [N] | SerialNumber [N] | IOcabinet [N],IObay [N],IOchassis [N],Slot [N],Cabinet [N],Cell [N]
```

```
ClearConfigurationWithDataLoss = Yes|No ; COMMENT: This option is now deprecated.
```

```
LicenseKey = XXXXX-XXXXX-XXXXX-XXXXX-XXXXX
```

```
DeleteLicenseKey = XXXXX-XXXXX-XXXXX-XXXXX-XXXXX | * ; COMMENT: * is a wild card that enables you to delete all license keys on the specified controller.
```

```
RAIDArrayID = "XXXXXXXXXXXXXXXXXXXXX"
```

```

ReadCache = 0|10|20|25|30|40|50|60|70|75|80|90|100
WriteCache = 0|10|20|25|30|40|50|60|70|75|80|90|100
RebuildPriority = Low|Medium|High
ExpandPriority = Low|Medium|High
SurfaceScanDelay = N
PreferredPathMode = Auto|Manual

; COMMENT: the following five entries are used to optimize the controller
performance for video
MNPDelay = 0|1|2|...|60 ; units are minutes, zero indicates disabled
IRPEnable = Yes|No
DPOEnable = Yes|No
ElevatorSortEnable = Yes|No
QueueDepth = 2|4|8|16|32|Auto

Array = A|B|C|D|E|F|G|...Z|a|b|c|d|e|f

OnlineSpare = None | N | Port:ID,Port:ID... | Box:Bay,Box:Bay... |
Port:Box:Bay,Port:Box:Bay,... ; COMMENT: These values are available only in
Custom method mode. In Auto method mode, the choices are Yes|No.

Drive = * | N | Port:ID,Port:ID... | Box:Bay,Box:Bay... |
Port:Box:Bay,Port:Box:Bay,...

DriveType = SCSI | SAS | SATA

LogicalDrive = 1|2|3|...32
RAID = 0|1|5|50|6|60|adg|auto ; COMMENT: For HP Smart Array G6 and G7
controllers, RAID 6 and 60 are only available when SAAP is installed and the
license key registered.
ParityGroups = 2|N ; COMMENT: Necessary only for RAID 50 or 60. N > 2
Size = [N]|Max
Sectors = 32|63
StripeSize = 8|16|32|64|128|256
ArrayAccelerator = Enable|Disable
PreferredPath = 1|2

HBA_WW_ID = WWN
ConnectionName = UserDefinedName
HostMode = Default | Windows | Windows(degrade | openVMS | Tru64 | Linux |
Solaris | Netware | HP | Windows Sp2 ; COMMENT: The Windows(degrade value
must be entered as written.

```

Script file options

Options in ACU script files are divided into the following categories:

- Control category (on page 72)
- Controller category (on page 72)
- Array category (on page 76)
- Logical Drive category (on page 78)
- HBA category (on page 83)

Each category has several scripting options, but you do not always need to assign values to every option. ACU can use default values in some instances, while in other instances, a listed option might not be relevant for a particular configuration or scripting mode.

The options for each category are listed in the following table and described in more detail in the remainder of this section.

Category	Options	Description
Control	Action Method	These options define the overall behavior of ACU when it processes scripts and creates configurations. Control options can occur only once in a script file and must be listed first.
Controller	Controller CacheState ClearConfigurationWithDataLoss DeleteLicenseKey DPOEnable DriveWriteCache ElevatorSortEnable ExpandPriority IRPEnable LicenseKey MNPDelay NoBatteryWriteCache PreferredPathMode QueueDepth RaidArrayId ReadCache RebuildPriority SurfaceScanDelay SurfaceScanDelayExtended SurfaceScanMode WriteCache	Options in this category specify the controller that is to be configured (or the controller that had its configuration captured). Although the Controller option must begin this section of the script, you can script other options in this category in any order. You can use one script file to configure all controllers in a system, and you can configure the controllers identically or individually. If you define each controller configuration individually, enter the option values for one controller and its arrays and logical drives before specifying the option values for another controller.
Array	Array Drive DriveType Join OnlineSpare Split	These options describe an array that is to be configured on the controller that was previously specified in the script. (If no controller was previously specified, ACU stops processing the script and creates an error file.) Although the Array option must begin this section of the script, you can script the other options in this category in any order.

Category	Options	Description
Logical Drive	ArrayAccelerator LogicalDrive NumberOfParityGroups PreferredPath RAID Renumber Repeat ResourceVolumeOwner Sectors ShrinkSize Size SizeBlocks StripSize StripeSize	These options describe a logical drive that is to be configured on an array that was previously specified in the script. (If no array was previously specified, ACU stops processing the script and creates an error file.) Although the LogicalDrive option must begin this section of the script, you can script the other options in this category in any order.
HBA	ConnectionName HBA_WW_ID HostMode	These options specify an HBA that is to be configured.

Control category

The Control category has the following options:

- Action mode (on page 72)
- Method mode (on page 72)

Action mode

You must specify an Action mode:

- In Configure mode, you can create new arrays, but you cannot modify existing arrays. The controller must be connected to unassigned physical drives for this mode to be available.
- In Reconfigure mode, you can modify existing arrays. For example, you can set up an array expansion, a logical drive extension, or a migration. These procedures do not destroy data, unless you specifically want the data to be deleted. In this mode, ACU does not change an existing option setting unless you specifically script a different value for that option.

If you use the `-reset` command line switch, the existing controller configuration is cleared with data loss as the first step in the configuration process. This command line switch is not compatible with Reconfigure mode.

Method mode

The default value for this option is Auto. If you want to use Custom mode, you must specify it.

In Auto mode, ACU can perform an expansion, extension, or migration without user intervention if the values that you set for other options imply that such an operation is necessary.

Controller category

The Controller category has the following options:

- Controller (on page 73)
- CacheState (on page 73)

- ClearConfigurationWithDataLoss (on page 74)
- DeleteLicenseKey ("LicenseKey, DeleteLicenseKey" on page 74)
- DPOEnable ("Video performance options" on page 75)
- DriveWriteCache (on page 74)
- ElevatorSortEnable ("Video performance options" on page 75)
- ExpandPriority ("RebuildPriority, ExpandPriority" on page 75)
- IRPEnable ("Video performance options" on page 75)
- LicenseKey ("LicenseKey, DeleteLicenseKey" on page 74)
- MNPDelay ("Video performance options" on page 75)
- NoBatteryWriteCache (on page 74)
- PreferredPathMode (on page 74)
- QueueDepth ("Video performance options" on page 75)
- RaidArrayId (on page 74)
- ReadCache ("ReadCache, WriteCache" on page 75)
- RebuildPriority ("RebuildPriority, ExpandPriority" on page 75)
- SurfaceScanDelay (on page 75)
- SurfaceScanDelayExtended (on page 75)
- WriteCache ("ReadCache, WriteCache" on page 75)

Controller

You must enter a value for this option because it identifies the controller that you want to configure:

- `All`—Configure all detected controllers in the system.
- `First`—Configure the first controller found, based on the controller with the lowest PCI slot number. Internal controllers are used before external controllers.
The `-internal` and `-external` command line switches influence what ACU regards as the `First` controller. For example, if you use the `-external` switch, the `First` controller is the first external controller discovered, regardless of the number of internal controllers in the host system.
- `Slot [N] [:M]`—Configure the internal controller in slot number `N`, or the external controller at port `M` in slot `N`.
- `WWN [N]`—Configure the external controller that has the World Wide Name `N`.
- `SerialNumber [N]`—Configure the shared storage controller that has serial number `N`.
- `IOCabinet[N], IOBay[N], IOChassis[N], Slot[N], Cabinet[N], Cell[N]`—Configure the controller in the Integrity server that has the slot path information defined by this sequence of identifiers.

CacheState

This option enables you to flush the cache or disable flushing of the cache. Values are `FlushEnable` and `FlushDisable`.

You can use this option to prevent stale cache issues.

ClearConfigurationWithDataLoss

Clearing the configuration causes data loss because it deletes all logical volumes and arrays on the controller. If you clear a configuration, you can write commands later in the script file to create a new configuration from the liberated drive capacity.

The values for this option are `Yes` or `No`. The default value is `No`.

DriveWriteCache

This option controls the settings of the write cache for all connected physical disks. For this option, the setting is `Enable` or `Disable`. Not all physical disks or controllers support this option.

LicenseKey, DeleteLicenseKey

These options enable you to enter a 25-character license key to activate or uninstall some controller features. Hyphens can be entered, but are not required.

NoBatteryWriteCache

With this option, the controller can enable write cache when no battery is present or when the battery fails. Values are `Enable` or `Disable`. The default is `Disable`.

Some controllers do not support this option.

PreferredPathMode

The setting that you select for this option determines how the preferred I/O path to a particular logical drive is set for a redundant array controller that is in an active/active configuration.

Not all controllers support this feature, and controllers in an active/standby configuration disregard this option.

- **Auto** is the default setting for new configurations. In this case, the storage system automatically selects the I/O path from the redundant controller to the logical drive and dynamically load balances all paths.
- **Manual** enables you to assign the logical drive to a specific redundant controller. If you select this setting, use the `PreferredPath` (on page 80) command to specify the path.

If you are reconfiguring a controller and do not specify a setting for this option, the existing setting remains unchanged.

RaidArrayId

This option specifies the `RaidArrayId` for controllers that support this feature, such as fibre and shared storage controllers. The `RaidArrayId` is a user-defined string that identifies controllers.

```
"XXXXXXXXXXXXXXXXXXXXXXX"
```

The value can be a string of varying length that consists of any of the following characters:

```
a-z A-Z 0-9 ! @ # * ( ) , - _ + : . / [space]
```

This string cannot end in a space character.

This string has a maximum length that varies depending on the type of controller. For RA4x00 controllers, the maximum length is 24 characters. For other controllers, the maximum length is 20 characters.

The quotes enclosing the string are optional. The quotes allow you to use leading space characters in the `RaidArrayId`.

ReadCache, WriteCache

Enter a number between 0 and 100 to specify the percentage of cache that is to be allocated to drive reads or writes. The default value for both options is 50, unless the controller firmware supplies a specific default read/write ratio.

The allowable cache ratios depend on the controller model and whether it has battery-backed write cache.

RebuildPriority, ExpandPriority

These options establish the priority for rebuild and expand functions. Each option has three possible values: Low, Medium, and High.

These options are not required.

SurfaceScanDelay

Enter a number between 0 and 30 to specify the duration of the surface scan delay in seconds. This option is not required. If you do not specify a value for this option, the delay remains unchanged. A value of 0 disables the scan.

SurfaceScanDelayExtended

Enter a number between 0 and 300 to specify the duration of the surface scan delay in tenths of seconds. This option is not required. If you do not specify a value for this option, the delay remains unchanged. A value of 0 disables the scan.

If this parameter and `SurfaceScanDelay` are both in the input file, this parameter takes precedence.

SurfaceScanMode

This parameter specifies the Surface Scan Mode with the following values:

- `Idle`—The delay interval is set with the normal values from `SurfaceScanDelay` or `SurfaceScanDelayExtended`.
- `High`—The surface scan enters a mode guaranteed to make progress despite the level of controller I/O.
- `Disabled`—No mode is selected.

Video performance options

To use this features on an HP Smart Array G6 or G7 controller, a registered SAAP ("[About SAAP](#)" on page 8) license key is required.

To optimize the controller performance for video, set values for the following options as indicated:

```
DPOEnable = No
ElevatorSortEnable = Yes
IRPEnable = No
```

In addition:

- Set the `MNPDelay` to any integer value from 1 to 60 (units are minutes). If you want to disable this option, set a value of zero instead.

- Set the `QueueDepth` to one of the following values:

2|4|8|16|32|Automatic

Array category

The Array category has the following options:

- Array (on page 76)
- CachingArray (on page 76)
- Drive (on page 76)
- DriveType (on page 77)
- Join (on page 77)
- OnlineSpareMode (on page 77)
- OnlineSpare (on page 78)
- Split (on page 78)

Array

Enter a letter or pair of letters to identify the array that is to be created or reconfigured, and observe these additional limitations:

- In Configure mode, ACU creates a new array. The value that you specify for the array option must be the next available letter or pair of letters in the sequence, according to the number of existing arrays on the controller. AA follows Z, and BA follows AZ.
- In Reconfigure mode, ACU can either create a new array or reconfigure an existing array. In this case, the value that you specify can identify an existing array, or it can correspond to the next available array letter or pair of letters in the existing configuration.

CachingArray

This option specifies the Caching Array ID.

- In Configure mode, ACU creates a new caching array. The value that you specify for the array option must be the next available array letter in the existing configuration. AA follows Z, and BA follows AZ.
- In Reconfigure mode, the array letter identifies an existing array, or it can identify the next available array letter in the existing configuration to create a new array.

Drive

You can use this option in the input file to specify new physical drives for the array. Use this option to build a new array or to expand, shrink, or move an existing array.

Observe the following guidelines:

- If you are expanding an array, each drive that you add must have a capacity no less than that of the smallest drive already in the array. The added drives and the existing drives in the array must all be the same type (for example, SAS or SATA).
- If you are moving or shrinking an array, ACU Scripting compares the current drives with the requested drives, and then determines whether you are moving or shrinking. Shrinking or moving an array is supported only in Custom method mode.

- If the value of the `ClearConfigurationWithDataLoss` (on page 74) option is Yes, you can use the `Drive` option to remove drives from an array.

Determine which mode to use:

- **Auto method mode**—ACU configures all the available drives on the controller into one array. If the drives are of different capacities, ACU determines the capacity of the smallest drive and uses the same amount of space on all other available drives.
- **Custom method mode**—To specify the drives to be used in the array (different arrays on the same controller can use different methods), choose one of the following methods:
 - To specify individual drives, use the applicable convention (port:ID, box:bay, or port:box:bay).
 - To specify only the number of drives to use (not which specific drive IDs to use), enter that number as the value for this option. For example, if you enter `drive=3`, ACU uses the first three available drives to build or expand the array that you define in the remainder of the script. ACU automatically determines which drives are suitable to use.
 - To use all available drives, enter an asterisk as the value for this option. An array that is configured using this method cannot have a spare.

DriveType

This option specifies the interface type for drives that ACU must use to build the array.

Choose one of the valid drive types:

[SCSI | SAS | SATA | SATASSD | SASSSD]

The value is often used with the wildcard (*) or number argument for `Drive`.

Join

To use the `Join` command, observe the following requirements:

- The system must be offline.
- The `-offline` command line flag must be set when you start ACU Scripting.

The command fails if the OS reports that any of the volumes involved in the join are in use.



CAUTION: Executing the `Join` command on an online system can result in kernel panics, blue screens, and/or data loss.

The `Split` command converts a RAID 1 or RAID 10 logical volume into two RAID 0 logical volumes. The `Join` option reverses the splitting process by recombining the two RAID 0 volumes into a single RAID 1 or RAID 10 volume, depending on the number of physical disks.

For this process, ID is the ID of the resulting volume that was joined. All data on the other volume is lost.

```
Array = C
Join = A
```

After the command executes, Array C is removed, and its physical drives are now a mirror of the drives in Array A. The original contents of Array C are lost.

OnlineSpareMode

The value for this option specifies the behavior of the spare drives when they are activated.

For controllers that support Auto Replace Drive, setting this value to **AutoReplace** enables a rebuilt spare drive to become a data drive in the array. When the failed data drive is replaced, it assumes the role of the former spare, eliminating the need for a second array rebuild.

Behavior	Description
Dedicated	Default value for spares
AutoReplace	The spare drive becomes a data drive when the rebuild is complete.

OnlineSpare

The value for this option determines whether the array specified previously in the script will be configured with spare drives.

The interface type for drives and spare drives must match, for example, all SAS or all SATA.

Method mode	Possible values	Default value
Custom	To specify exactly which drives to use as spares, use the applicable convention (port:ID, box:bay, or port:box:bay). To specify only the number of spares (not the exact IDs), enter that number as the value for this option. ACU automatically selects only those drives that are suitable for the array. To specify that the array should not have spares, enter <i>None</i> .	In Configure action mode: None In Reconfigure action mode, ACU ignores any value entered for this option and keeps any spares that are already present in the configuration.
Auto	Yes (indicating one spare) No	In Configure action mode: Yes (indicating one spare) In Reconfigure action mode, ACU ignores any value entered for this option and keeps any spares that are already present in the configuration.

Split

This command splits RAID 1 or RAID 10 volumes into individual RAID 0 volumes.

Two values are available:

HIDDEN—The driver hides the new logical volume from the OS.

VISIBLE—The new logical volume is visible immediately to the OS.

Logical Drive category

The Logical Drive category has the following options:

- [ArrayAccelerator](#) (on page 79)
- [LogicalDrive](#) (on page 79)
- [CachingLogicalDrive](#) (on page 79)
- [CachedLogicalDrive](#) (on page 79)
- [NumberOfParityGroups](#) (on page 79)
- [PreferredPath](#) (on page 80)

- RAID (on page 80)
- Renumber (on page 80)
- Repeat (on page 81)
- ResourceVolumeOwner (on page 81)
- Sectors (on page 81)
- ShrinkSize (on page 81)
- Size (on page 81)
- SizeBlocks (on page 82)
- StripeSize (on page 82)
- StripSize (on page 82)

ArrayAccelerator

This option specifies whether the array accelerator is enabled or disabled for the specified logical drive. The default value is Enabled.

LogicalDrive

The value that you enter for this option specifies the ID number of the logical drive that is to be created or modified. The first logical drive on an array must have an ID of 1 (not 0), and logical drive numbering must be contiguous.

- In Configure action mode, ACU accepts only the ID number of the next possible logical drive.
- In Reconfigure action mode, ACU also accepts the ID number of any existing logical drive.

CachingLogicalDrive

The value that you enter for this option specifies the ID number of the caching logical drive that is to be created or modified. The first logical drive on an array must have an ID of 1 (not 0), and logical drive numbering must be contiguous.

- In Configure mode, ACU accepts only the ID number of the next possible logical drive.
- In Reconfigure mode, ACU also accepts the ID number of any existing logical drive.

Caching logical drives and cached data drives must be on the same Array Controller.

CachedLogicalDrive

The value that you enter for this option specifies the data logical drive ID number to associate with the Caching logical drive.

In Configure mode, ACU accepts the ID number of any existing logical drive.

Caching logical drives and cached data drives must be on the same Array Controller.

NumberOfParityGroups

When you create a RAID 50 or RAID 60 configuration, you must also set the number of parity groups.

You can use any integer value greater than 1 for this setting, with the restriction that the total number of physical drives in the array must be exactly divisible by the number of parity groups.

The maximum number of parity groups possible for a particular number of physical drives is the total number of drives divided by the minimum number of drives necessary for that RAID level (three for RAID 50, four for RAID 60).

PreferredPath

If you select the Manual setting for `PreferredPathMode` (on page 74), use the `PreferredPath` command to specify the path for I/O to the logical drive on a redundant controller in active/active mode.

The default setting for this option is **1**. With this setting, the controller in chassis slot 1 is the preferred controller for I/O to the logical drive. If you select **2**, the controller in chassis slot 2 becomes the preferred controller for the logical drive.

To determine the chassis slot numbers, use the `show` command on a controller that supports redundant controllers.

RAID

The value that you enter for this option specifies the RAID level of the logical drive.

- When the Action mode is Configure and the Method mode is Auto, ACU automatically selects the highest RAID level that the controller and drive configuration can support except RAID 50 or RAID 60. To specify RAID 50 or 60 for a controller that supports either of these RAID levels, use the Custom setting. In this case, you must also specify the number of parity groups ("`NumberOfParityGroups`" on page 79).
- When the Action mode is Reconfigure, the default value is the existing RAID level for that logical drive. If you specify a different RAID setting, then ACU either ignores the new setting (when Method mode is Auto) or attempts to migrate the logical drive to the specified RAID level (when Method mode is Custom).

ACU supports the following values for RAID levels:

- 60—RAID 60
- 50—RAID 50
- ADG—RAID ADG is the equivalent of RAID 6
- 6—RAID 6
- 5—RAID 5
- 4—RAID 4
- 10ADM—RAID 1 with 3-way mirroring
- 10—RAID 10 (mirroring with 2 disks)
- 1ADM—RAID 1 with 3-way mirroring
- 1—RAID 1 (mirroring with 2 disks)
- 0—RAID 0

When using an HP Smart Array G6 or G7 controller, some RAID levels require SAAP ("[About SAAP](#)" on page 8).

Renumber

This option rennumbers the logical drive to *N*.

Typically used after a `Join` command, this option ensures the proper logical volume numbering is set, for example, the boot volume is ID 1.

Do not mix `Renumber` and `Join` operations in the same script, because ACU Scripting does not update its internal state after a renumber. If other commands are mixed with `Renumber`, commands targeting specific volumes might target the incorrect volume.

Repeat

The value you enter for this option specifies the number of times that ACU is to repeat this logical drive configuration.

Use one of the following values:

- `N`—In Configure mode, ACU creates *N* new logical drives.
- `MAX`—ACU creates the maximum number of logical drives possible. The number of drives created depends on the number of existing drives and the maximum number of logical drives supported by the controller.

You must specify the logical drive ID as `Next`. This `Size` option controls the size of each logical drive, or if the size is set to `MAX`, the size of the volumes is set to consume all available space on the array.

ResourceVolumeOwner

This option specifies that an existing logical drive (*N*) should be a snapshot resource volume. The option also specifies the owner logical drive ID. If you do not specify this command on some older model controllers, the logical drive remains a normal data volume.

Sectors

This option specifies the number of sectors that are to comprise each track. Enter 32 to disable MaxBoot or 63 to enable it.

- For new logical drives, the default setting is 63 if the logical drive is larger than 502 GB. Otherwise, the default setting is 32.
- For an existing logical drive, the default setting is the existing setting.

Logical drive performance is likely to decrease with MaxBoot enabled.

ShrinkSize

In Reconfigure mode, this option specifies the resulting size of the logical drive (in MB) to be reduced.

`ShrinkSize` is not valid in Configure mode.

Size

This parameter specifies the size of the logical drive.

Use the following values to set the size:

- `N`—The size in MB
- `MAX`—Uses all available unused space in the array for the logical drive. This value is the default.
- `MAXMBR`—Creates the largest size of volume supportable by a 32-bit MBR (2TiB)

In Reconfigure mode, the default setting is the existing size of the logical drive. If you enter a larger value, ACU extends the logical drive to the new size if there is unused drive capacity on the same array, as long as the operating system supports logical drive extension. You cannot reduce the size of the logical drive.



CAUTION: Back up all data before extending a logical drive.

SizeBlocks

This option specifies the size of the logical drive in 512-byte blocks. Use this option to set an exact size without rounding.

When ACU Scripting captures a configuration, it reports volume sizes rounded down to the nearest MB. When the capture is played back, the size is reduced to fit within the already rounded MB size. This option prevents the loss of volume size over successive generations of captures.

However, if the size does not fit the geometry of the array, caused by different disks or RAID levels, the size will be rounded down to fit the geometry.

StripeSize

Beginning with ACU, ACU CLI, and ACU Scripting v8.55, the term **stripe size** has been replaced by **strip size**. This labeling change does not affect functionality.

When distributing data across multiple physical drives (striping), the **strip size** is the amount of data that is written to each physical drive. The **full stripe size** refers to the combined size of all the strips across all physical drives, excluding parity-only drives.

For more information, see the option for StripSize (on page 82).

StripSize

Beginning with ACU, ACU CLI, and ACU Scripting v8.55, the `StripSize` option replaces the `StripeSize` option. When distributing data across multiple physical drives (striping), the **strip size** is the amount of data that is written to each physical drive. The **full stripe size** refers to the combined size of all the strips across all physical drives, excluding parity-only drives.

You can enter a numerical value for this option to specify the size of the data strips (in kilobytes), or you can leave this option blank and allow ACU to use a default value.

Strip sizes available for a RAID level vary based on the controller and the controller firmware level. The maximum strip size changes dynamically and is reduced for arrays with a large number of data drives or with smaller controller cache sizes. (The controller must be able to read an entire stripe of data at a time into cache memory during transformation. Available memory is the limiting factor.)

The following strip sizes are available:

- 8—8 KB
- 16—16 KB
- 32—32 KB
- 64—64 KB
- 128—128 KB
- 256—256 KB

- 512—512 KB
- 1024—1024 KB (supported starting with Gen8 controllers)

The default strip size value depends on the action mode:

- In Configure action mode, the default value is determined by the RAID level that you specified earlier in the script.
- In Reconfigure action mode, the default value for this option is the strip size that is already configured for the logical drive. If you enter a value that is different from the existing strip size, ACU attempts to migrate the logical drive to the strip size that you specify. (If you intend to migrate the logical drive, back up all data before starting the migration procedure.)

HBA category

The HBA category has the following options:

- ConnectionName (on page 83)
- HBA_WW_ID (on page 83)
- HostMode (on page 83)

ConnectionName

This option is a user-defined string used as the connection name for the specified HBA.

The string can consist of:

- A maximum of 16 characters
- Embedded space characters but cannot end with a space character
- Any of the following characters: a–z, A–Z, 0–9, !, @, #, *, (,), -, _ , +, :, ., /, and [space]

HBA_WW_ID

This option specifies which HBA, based on its assigned WWN, is modified by the configuration changes.

HostMode

This option specifies the HostMode for a selected HBA. Setting the Host Mode optimizes the storage array for the selected operating system. The available host modes for an HBA are device-specific. Not all modes are available on all devices. Not all HBAs support a HostMode.

The following operating system options might be available:

- Default
- Microsoft® Windows®
- OpenVMS
- Tru64
- Linux
- Solaris
- Netware
- HP-UX

XML support

ACU scripting v8.30 and later support an XML file format for input and output.

XML output

To create an XML output document, use an XML suffix with the output file name:

```
C:\hpacuscripting -c out.xml
```

The following text is an example XML output file.

```
<?xml version="1.0"?>
<Config.document>
<!-- Date captured: Tue Jun 09 10:03:08 2009 -->
<!-- Version: 8:30:4 -->
<Action>Configure</Action>
<Method>Custom</Method>
<Controller ID="Slot 1">
  <!-- Controller HP Smart Array P410, Firmware Version 1.99 -->
  <ReadCache>25</ReadCache>
  <WriteCache>75</WriteCache>
  <RebuildPriority>Medium</RebuildPriority>
  <ExpandPriority>Medium</ExpandPriority>
  <SurfaceScanDelay>3</SurfaceScanDelay>
  <DriveWriteCache>Disabled</DriveWriteCache>
  <LicenseKey>35DRP-7RH6S-R89GR-4MX6N-8K48X</LicenseKey>
  <!-- LicenseKeyType "Flex License" -->
  <MNPDelay>60</MNPDelay>
  <IRPEnable>Disabled</IRPEnable>
  <DPOEnable>Disabled</DPOEnable>
  <ElevatorSortEnable>Enabled</ElevatorSortEnable>
  <QueueDepth>Automatic</QueueDepth>
  <!-- Unassigned Drives 1I:4:5 (60.0 GB), 2I:2:1 (72 GB) -->
  <Array ID="A">
    <!-- Array Drive Type is Solid State SATA -->
    <!-- Free space 0 GBytes -->
    <!-- 1I:4:8 (120.0 GB),1I:4:7 (120.0 GB) -->
    <Drive>1I:4:8, 1I:4:7</Drive>
    <OnlineSpare>No</OnlineSpare>
    <LogicalDrive ID="1">
      <Raid>1</Raid>
      <Size>114439</Size>
      <Sectors>32</Sectors>
      <StripeSize>128</StripeSize>
      <ArrayAccelerator>Enabled</ArrayAccelerator>
    </LogicalDrive>
```

```

</Array>
<Array ID="B">
  <!-- Array Drive Type is SAS -->
  <!-- Free space 0 GBytes -->
  <!-- 1I:4:6 (72 GB),2I:2:4 (72 GB),2I:2:3 (72 GB),2I:2:2 (72 GB)-->
  <Drive>1I:4:6, 2I:2:4, 2I:2:3, 2I:2:2</Drive>
  <OnlineSpare>No</OnlineSpare>
  <LogicalDrive ID="2">
    <Raid>5</Raid>
    <Size>52478</Size>
    <Sectors>32</Sectors>
    <StripeSize>64</StripeSize>
    <ArrayAccelerator>Enabled</ArrayAccelerator>
  </LogicalDrive>
  <LogicalDrive ID="3">
    <Raid>5</Raid>
    <Size>52478</Size>
    <Sectors>32</Sectors>
    <StripeSize>64</StripeSize>
    <ArrayAccelerator>Enabled</ArrayAccelerator>
  </LogicalDrive>
  <LogicalDrive ID="4">
    <Raid>5</Raid>
    <Size>52478</Size>
    <Sectors>32</Sectors>
    <StripeSize>64</StripeSize>
    <ArrayAccelerator>Enabled</ArrayAccelerator>
  </LogicalDrive>
  <LogicalDrive ID="5">
    <Raid>5</Raid>
    <Size>52478</Size>
    <Sectors>32</Sectors>
    <StripeSize>64</StripeSize>
    <ArrayAccelerator>Enabled</ArrayAccelerator>
  </LogicalDrive>
</Array>
</Controller>
</Config.document>

```

XML input

XML input follows the same format as the document for XML output (on page 84). Use the same parameter order as the standard input format.

The following example shows a simple input script in both standard and XML formats.

Standard format	XML format
Action= Configure Method= Custom Controller= Slot 1 Array=A Drive= 1I:4:8, 1I:4:7 LogicalDrive= 1 RAID= 0 Size= 100000	<pre> <?xml version="1.0"?> <Config.document> <Action>Configure</Action> <Method>Custom</Method> <Controller ID="Slot 1"> <Array ID="A"> <Drive>1I:4:8, 1I:4:7</Drive> <LogicalDrive ID="1"> <Raid>0</Raid> <Size>100000</Size> </LogicalDrive> </Array> </Controller> </Config.document> </pre>

XML input file DTD

The following DTD outlines the parameters for an ACU scripting XML input file.

```

<!DOCTYPE Config.document [
<!ELEMENT Config.document ( Action, Method, Controller+ ) >
<!ELEMENT Action ( Configure | Reconfigure ) >
<!ELEMENT Method ( Auto | Custom ) >

<!ELEMENT Controller, Array ) ChassisName? |
ClearConfigurationWithDataLoss | DPOEnable? | DriveWriteCache? |
ElevatorSortEnable? | ExpandPriority? | IRPEnable? | Initiator? |
LicenseKey? | MNPDelay? | PreferredPathMode? | QueueDepth? | ReadCache? |
RebuildPriority? | SurfaceScanDelay? | WriteCache? ) >
  <!ATTLIST Controller ID PCDATA #REQUIRED >
  <!ELEMENT ClearConfigurationWithDataLoss ( YES | NO ) NO >
  <!ELEMENT DPOEnable ( YES | NO ) >
  <!ELEMENT DriveWriteCache ( ENABLE | DISABLE ) >
  <!ELEMENT ElevatorSortEnable ( YES | NO ) >
  <!ELEMENT ExpandPriority ( HIGH | MEDIUM | LOW ) >
  <!ELEMENT IRPEnable ( YES | NO ) >
  <!ELEMENT LicenseKey ( #PCDATA ) >
  <!ELEMENT MNPDelay ( #PCDATA ) >
  <!ELEMENT PreferredPathMode ( AUTO | MANUAL ) >
  <!ELEMENT ReadCache ( 0 | 10 | 20 | 30 | 40 | 50 | 60 | 70 | 80 | 90 | 100
  ) >
  <!ELEMENT RebuildPriority ( HIGH | MEDIUM | LOW ) >
  <!ELEMENT SurfaceScanDelay ( #PCDATA ) >
  <!ELEMENT QueueDepth ( #PCDATA ) >
  <!ELEMENT WriteCache ( 0 | 10 | 20 | 30 | 40 | 50 | 60 | 70 | 80 | 90 |
  100 ) >

  <!ELEMENT Array, LogicalDrive+ ( Drive, OnlineSpare ) >
  <!ATTLIST Array ID PCDATA #REQUIRED >

```

```

<!ELEMENT Drive ( ALL | #PCDATA ) >
<!ELEMENT OnlineSpare ( YES | NO | #PCDATA ) >

<!ELEMENT LogicalDrive ( Raid, Repeat | Size, Sectors, StripeSize,
ArrayAccelerator? ) >
<!ATTLIST LogicalDrive ID PCDATA #REQUIRED >
<!ELEMENT ArrayAccelerator ( ENABLE | DISABLE ) ENABLE >
<!ELEMENT Raid ( 0 | 1 | 1ADM | 10 | 10ADM | 4 | 5 | 6 | ADG | 50 | 60 )
>
<!ELEMENT Repeat ( MAX | #PCDATA ) >
<!ELEMENT Sectors ( 32 | 63 ) >
<!ELEMENT Size ( MAX | #PCDATA ) MAX >
<!ELEMENT StripeSize ( 8 | 16 | 32 | 64 | 128 | 256 | 512 ) >

<!ELEMENT Initiator ( HBA_WW_ID, ConnectionName?, HostMode? ) >
  <!ELEMENT HBA_WW_ID ( #PCDATA ) >
  <!ELEMENT ConnectionName ( #PCDATA ) >
  <!ELEMENT HostMode ( #PCDATA ) >
]>

```

ACU scripting warning messages

Warning code	Warning message	Comment or clarification
4000	License key is already installed	The input file specified a license key. However, a license key is already installed on the controller.
4001	Clear configuration command failed - configuration is already cleared	The <code>-reset</code> option was used on a controller that does not have a configuration.
4002	Chassis name is already set to this value	The chassis name is already set to the supplied value. The command is ignored.
4003	One or more controller commands were skipped because the controller is not configured	Certain commands require that the controller be configured before they can be sent to the controller.
4004	Using Repeat function	Some commands have been ignored because the Repeat parameter has been specified in the input file.
4005	The system must be rebooted for the firmware flash to complete	A controller has been flashed with new firmware. The new firmware requires a reboot to take effect.
4006	Unable to set the array accelerator for this volume. The cache board may be missing or have a bad status, or the controller may not support a cache	The controller may not support a cache (for example, some Software RAID controllers) or the cache may be missing or defective.

ACU scripting error messages

Error code	Error message	Comment or clarification
0	—	No error exists. The program completed

Error code	Error message	Comment or clarification
		successfully.
2053	Too many coinciding expansion, migration, or extension operations	The new configuration requires more transformations than are possible at one time. For example, you cannot expand a logical volume and transform its RAID level at the same time.
2056	Controller does not support license keys	The controller does not support license key entry or deletion.
2059	Invalid license key	The entered license key is not a valid license key.
2064	Controller does not support SSP	The controller does not support SSP functions.
2817	Invalid Action	The requested actions are invalid, for example, combining <code>-reset</code> with capture mode.
2818	Invalid Method	The method must be either Custom or Auto.
2819	Invalid Controller	An invalid controller value was specified.
2821	No controllers detected	No controllers were detected. This error applies to Input mode only.
2823	Invalid Rebuild Priority	The specified rebuild priority is not supported.
2824	Invalid Expand Priority	The specified expand priority is not supported. This error also occurs if the controller does not allow expansion, and therefore does not support expand priority.
2825	Invalid Array	The array ID is invalid.
2826	Array not specified	The Array command is missing from the script file. Some commands were found that require an Array to be specified.
2827	New Array ID does not match the next available Array ID.	The array ID in the script file does not match the array ID of the next available array. For example, if the configuration has an Array A and the input file specifies Array C (without Array B), then the script generates this error.
2828	New Array ID already exists	The array ID specified in the script file (in Configure mode) already exists in the configuration. Configure mode can only create new arrays.
2829	Cannot create Array	The controller will not allow a new array to be created, either because the controller has no unassigned physical drives attached or because the maximum number of arrays or logical drives has been reached already.
2830	Cannot expand Array	The Array could not be expanded, either because the controller does not support expansion, or the current configuration does not allow expansion to occur on the array.
2831	Cannot change Array Spare	The spare state of the array could not be changed. This error can occur when you attempt to add or drop a spare and the current configuration does not allow a spare state change for the array.
2832	Invalid physical drive	A specified physical drive is not a valid physical drive, or it cannot be placed in the array.
2833	Invalid Spare	A specified spare is not a valid spare drive, or it cannot be placed in the array as a spare.
2834	Invalid logical drive	The logical drive ID is not valid.

Error code	Error message	Comment or clarification
2836	New Logical Drive ID does not match the next available logical drive ID. If you are attempting to replicate a configuration that has non-consecutive logical drive numbers, then you must change the script file so that the logical drive numbers are consecutive.	The script file specifies a logical drive ID that is not the first unused ID in the sequence. For example, this message appears if the controller has only Logical Drive 1 and the script file specifies creation of Logical Drive 3 (omitting Logical Drive 2). A common cause of this error is that the input file specifies nonsequential logical drive numbers. In this case, change the logical drive numbers in the input file so that they are sequential.
2837	New Logical Drive ID already exists	This error occurs in Configure mode when the logical drive ID specified in the script file already exists in the configuration. In Configure mode, you can create new logical drives only.
2838	Cannot create Logical Drive	The array has no free space, or the maximum number of logical drives has been reached already.
2839	Cannot migrate Logical Drive RAID	The controller does not support RAID migration, or migration is not possible with the current controller configuration.
2841	Cannot extend Logical Drive	The controller does not support extension, or the current controller configuration cannot be extended. For example, extension is not possible if the array has no free space.
2842	Invalid RAID	The specified RAID level is invalid or is not possible with the current physical disk and array configuration.
2843	Invalid Size	The specified size is invalid or is not possible with the current configuration.
2844	Invalid Stripe Size	The specified stripe size is invalid, not supported by the current RAID level, or not possible with the current configuration.
2849	Invalid ClearConfigurationWithDataLoss parameter	The valid parameters are Yes and No (default).
2850	Controller does not support Chassis Name	The controller does not support setting a chassis name.
2851	Invalid Chassis Name	The entered chassis name is invalid. Use characters from the set a-z, A-Z, 0-9, !, @, #, *, (,), ,, -, _ +, :, ., /, and [space]. The name cannot end with a space character or exceed the maximum number of characters allowed by the controller.
2852	Invalid SSP State	The requested SSP state is not a valid SSP state.
2853	Cannot change SSP settings	The SSP settings cannot be changed for the controller or logical drive.
2854	Invalid SSP Adapter ID	The adapter ID is not a valid adapter ID that was detected by the controller.
2857	Invalid Surface Scan Delay	—
2861	Controller does not support redundancy settings	The controller is not redundant or does not support redundancy settings.
2864	Invalid Preferred Path Mode	The specified value for the preferred path mode is not valid, or the controller is not available.
2865	Invalid Preferred Path	The specified preferred path is not a valid chassis

Error code	Error message	Comment or clarification
		slot for an available, active controller, or the controller is not available.
2866	Failure opening capture file <text>	—
2867	Failure opening input file <text>	—
2868	Failure opening error file <text>	—
2877	There are no suitable spares available	ACU found no drives that could be used as spares for the specified array.
2880	Invalid Physical Disk Type Specified	—
2882	Invalid MNP delay	The specified value for MNP delay is invalid.
3000	Invalid Option	The value of the option supplied to this parameter is invalid.
3002	Command Failed	The controller returned an error for a command.
3003	License Key Delete Failed	ACU is unable to delete the license key.
3004	Invalid Sector Size	—
3005	Cannot delete Array	—
3006	Invalid Number of Parity Groups	—
3007	Chassis name is too long	—
3008	Chassis name is already in use	Another controller is already using the entered chassis name.
3009	Auto Configure failed	The Auto Configure mode was unable to complete auto configuration.
3010	Cannot extend logical drive, not enough free space for the requested size	—
3011	Cannot extend logical drive, requested size is too small	—
3012	Cannot specify both SIZE and SHRINKSIZE	The input file cannot specify both a SIZE and SHRINKSIZE parameter.
3013	Cannot shrink Array	The array shrink operation was not successful.
3014	Cannot move Array	The array move operation was not successful.
3015	Invalid operation - Advanced Pack support required	The requested operation requires a valid license key to be entered.
3016	Spare drives cannot be specified by a count in Reconfigure mode	When in Reconfigure mode, the requested spare drives must be specified by their addresses. A simple count cannot be used.
3017	Disk drives cannot be specified by a count in Reconfigure mode	When in Reconfigure mode, the requested data drives must be specified by their addresses. A simple count cannot be used.
3018	Invalid number of physical disks	—
3019	Cannot create Array - no physical disks specified	ACU cannot create an array unless physical disks are specified in the input file DRIVE parameter.
3020	SSP must be enabled in order to perform this operation	For the specified operation, ACU requires SSP to be supported and enabled.
3021	Invalid connection name	—

Error code	Error message	Comment or clarification
3022	The <code>connectionname</code> cannot be removed when the <code>hostmode</code> has a non-default value.	—
3023	Invalid Host Mode	—
3024	Invalid Adapter ID	—
3025	This controller does not have host mode modification capability	—
3026	You need to have administrator rights to continue.	—
3027	Another instance of ACU is already running (possibly a service). Please terminate the ACU application before running ACU scripting.	—
3028	Invalid Drive Cache setting. Valid options are ENABLE and DISABLE.	—
3029	Invalid or out of order Command	Verify the ordering of the commands in the input configuration file.
3030	Invalid or missing Array for Reconfigure	Expecting a valid array in Reconfigure mode
3031	Invalid or missing Filename for Firmware Update	The firmware filename supplied is not a valid filename.
3032	Firmware Update Failed	The firmware flash operation on the controller failed.
3033	This controller has been configured with a more recent version of software. To prevent data loss, configuration changes to this controller are not allowed. Please upgrade to the latest version to be able to continue to configure this controller.	Some changes to SOULAPI are not backward-compatible. This check prevents the use of an older version of software on a controller configured with a newer version, possibly overriding changes and destroying data.
3034	Operations on this Array are temporarily unavailable while the Array is transforming.	The user requested too many simultaneous changes. For example, the user added new disks to an array (expand array) and changed the size or RAID level of logical volumes on the array. The solution is for the user to wait until the array transformation is complete.
3035	Invalid value for <code>NoBatteryWriteCache</code>	A problem exists with the <code>NOBATTERYWRITECACHE</code> command. Check the supplied argument. Not all controllers support this operation.
3036	Cannot delete Logical Drive	A problem occurred during the attempt to delete the specified logical drive. The drive may be in use by the OS, it may not exist, or it may not be the last logical drive on the array.

HP Array Diagnostics and SmartSSD Wear Gauge Utility

About the utility

Formerly known as the Array Diagnostics Utility, the HP Array Diagnostics and SmartSSD Wear Gauge Utility collects all possible information about storage devices in the system, detects all problems, and provides a detailed configuration report in .zip format.

After downloading and installing the software, you can run the utility as a GUI or a CLI in an online environment. The functionality in this utility is mirrored in the diagnostics features of the HP Array Configuration Utility, which can be run in an offline environment.

The utility generates two types of reports:

- Array diagnostic report
This report contains information about all devices, such as array controllers, storage enclosures, drive cages, as well as logical, physical, and tape drives. For supported solid state drives, this report also contains SmartSSD Wear Gauge information.
- SmartSSD Wear Gauge report
This report contains information about the current usage level and remaining expected lifetime of solid state drives attached to the system.

For more information, see "Reported information (on page 92)."

Reported information

The array diagnostic report provides detailed information about devices (array controllers, storage enclosures, drive cages, physical drives, logical drives, and tape drives).

For example, device information on a typical embedded controller might include:

- Software versions
- Errors
- Controller information:
 - Name
 - Attached devices
 - Description
 - PCI bus
 - PCI device
 - PCI function
- Drive information:

- Interface
- WWID
- Drive model
- Serial number
- Firmware revision
- Total blocks

The SmartSSD Wear Gauge report contains information on the current usage level of and expected lifetime remaining for solid state drives attached to the system.

For discovered SSDs, the report summary page provides the following calculated totals:

- Total Solid State Drives with Wearout Status
- Total Smart Array Solid State Drives
- Total Non Smart Array Solid State Drives
- Total Solid State SAS Drives
- Total Solid State SATA Drives
- Total Solid State Drives

In addition to these totals, the summary page also displays the following tables:

- Solid State Drives with Wearout Status
- Solid State Drives with Less Than an Estimated 56 Days of Life Remaining
- Solid State Drives with Less Than 2% Usage Remaining
- Solid State Drives with Less Than 5% Usage Remaining
- Smart Array Controllers
- Non Smart Array Controllers

When the generated report is viewed in a browser, the report page displays the following fields in the SmartSSD Status table.

Field	Description
SSD Wear Status	<p>Indicates the SSD's wear status with one of the following messages:</p> <ul style="list-style-type: none"> • OK • Not Supported • The SmartSSD Wear Gauge log is full. Wear Gauge parameters are not available. • SSD has less than 5% usage remaining before wearout. • SSD has less than 2% usage remaining before wearout. • SSD has less than an estimated 56 days before it reaches the maximum usage limit for writes (wearout) and should be replaced as soon as possible. • SSD has less than 5% of usage remaining before wearout. It has less than an estimated 56 days before it reaches the maximum usage limit and should be replaced as soon as possible. • SSD has less than 2% of usage remaining before wearout. It has less than an estimated 56 days before it reaches the maximum usage limit and should be replaced as soon as possible. • SSD has reached the maximum rated usage limit for writes (wearout) and should be replaced immediately.
Power Cycles	Indicates the number of times the SSD has powered on from the powered off state
Power On Hours	Indicates the number of hours the SSD has been powered on
Estimated Life Remaining Based On Workload To Date	Indicates an estimate of the number of days the SSD has before SSD Utilization reaches 100%. This field is not displayed when the SSD Utilization is still at 0%.
Usage Remaining	Indicates the percentage of the SSD that has not worn out. Usage remaining is equal to the difference of 100 and the SSD Utilization percentage.
SSD Utilization	Indicates the percentage of the SSD that has worn out

Installing the utility

1. Browse to the HP Array Configuration Utility website (<http://h18004.www1.hp.com/products/servers/proliantstorage/software-management/acumatrix/index.html>).
2. Click **Download software**.
3. Select an OS.
4. Identify the preferred software and version, and then click **Download**.
5. Save, and then run, the executable file.

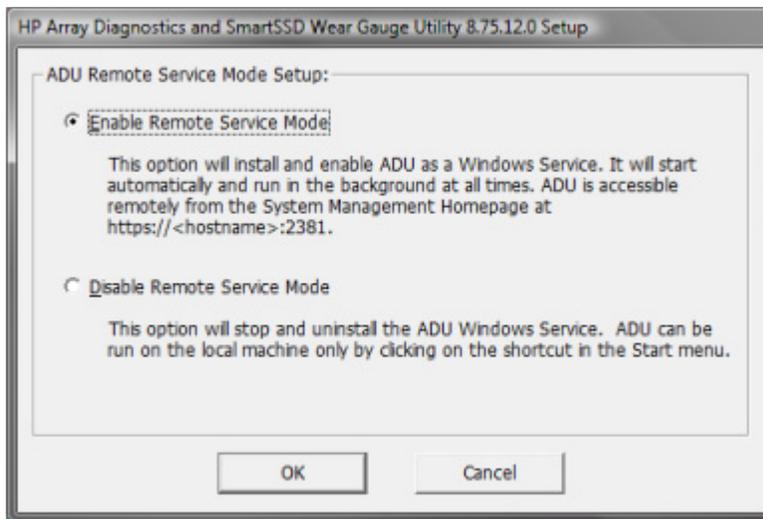
By default, the software installs at C:\Program Files\HP System Tools\.

Setting up ADU Remote Service Mode

You can run ADU as a Windows Service, so that it starts automatically and runs in the background at all times. ADU is accessible remotely from the System Management Homepage (<https://localhost:2381>).

To set up the service:

1. Click **Start>All Programs>HP System Tools>HP Array Diagnostics and SmartSSD Wear Gauge Utility>Setup ADU Remote Mode**.
2. Select **Enable Remote Service Mode**.



3. Click **OK**.

Launching the utility in CLI mode

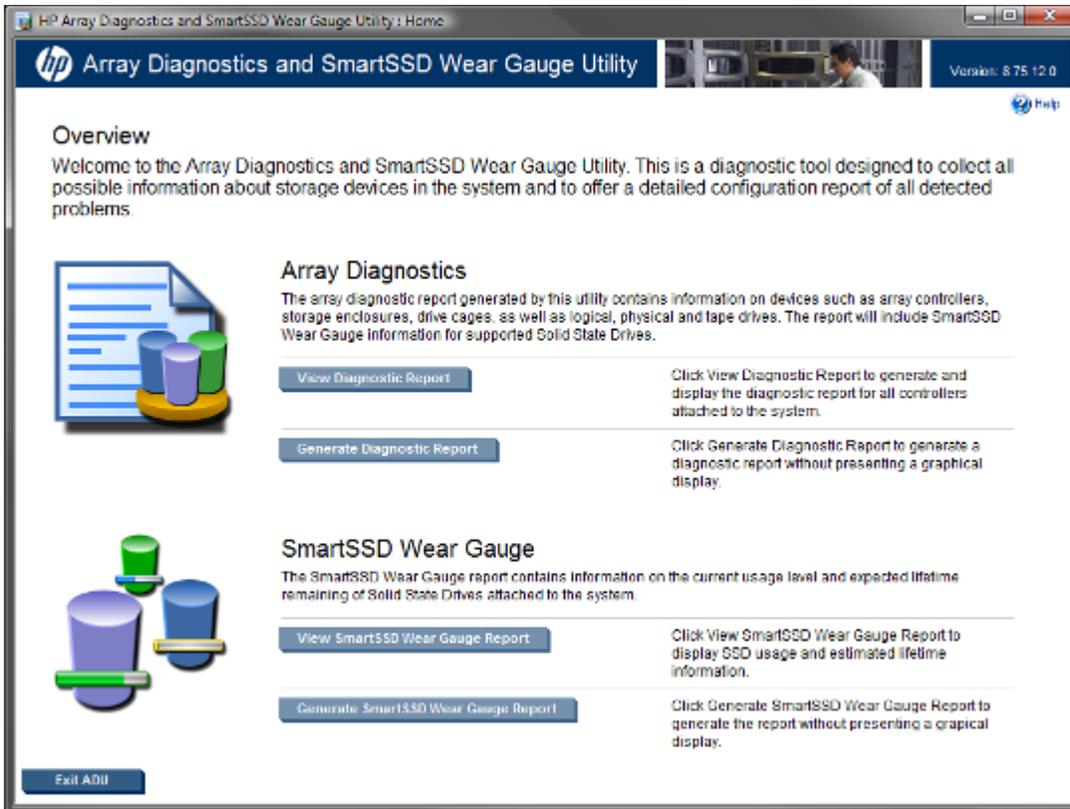
1. Open a command prompt.
2. Change directory (cd) to the location where `hpaducli.exe` is installed. This directory is commonly `C:\Program Files\Compaq\hpadu\bin`.
3. Do one of the following:
 - o Generate a diagnostic report with the following command:
`hpaducli -f adu-report.zip`
 - o Generate a SmartSSD Wear Gauge report with the following command:
`hpaducli -ssdrpt -f ssd-report.zip`

For more options, use the following command:
`hpaducli -help`

Launching the utility in GUI mode

1. Click **Start>All Programs>HP System Tools>HP Array Diagnostics and SmartSSD Wear Gauge Utility>HP Array Diagnostics and SmartSSD Wear Gauge Utility**.

The splash screen appears.



2. Select one of the following tasks:
 - o Viewing the diagnostic report (on page 96)
 - o Generating the diagnostic report (on page 98)
 - o Viewing the SmartSSD Wear Gauge report (on page 99)
 - o Generating the SmartSSD Wear Gauge report (on page 100)

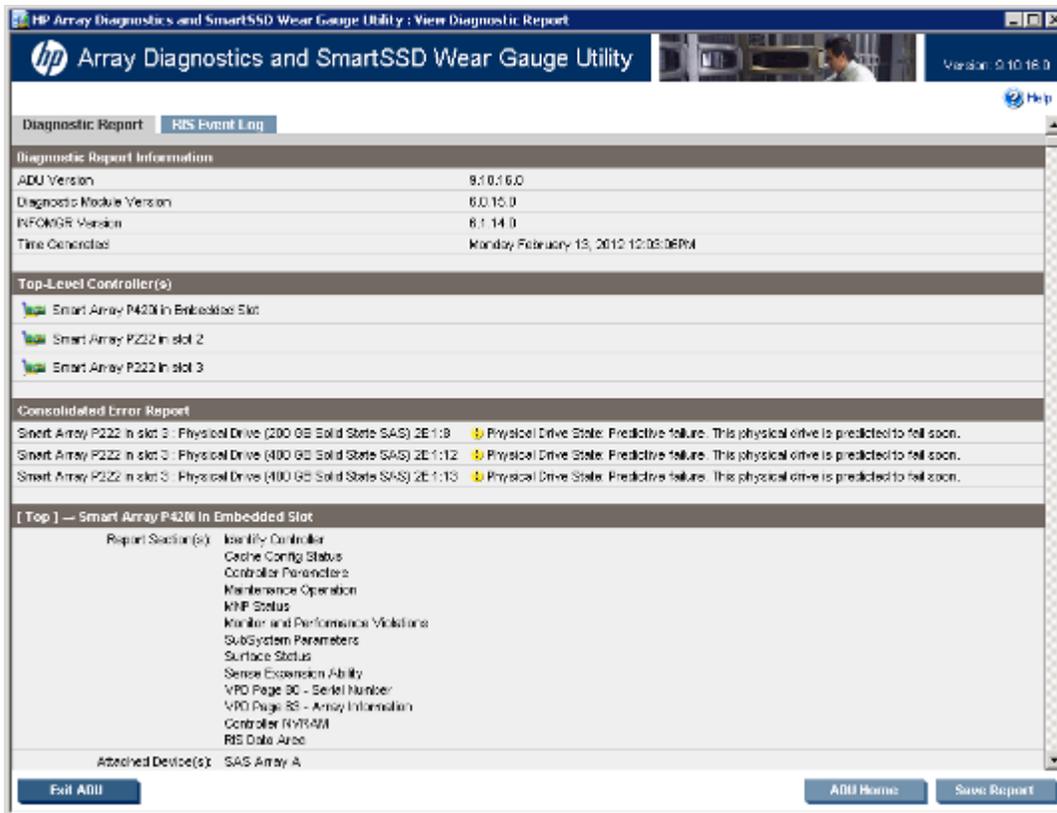
For more information about the interface, select **Help**.

Diagnostic report procedures

Viewing the diagnostic report

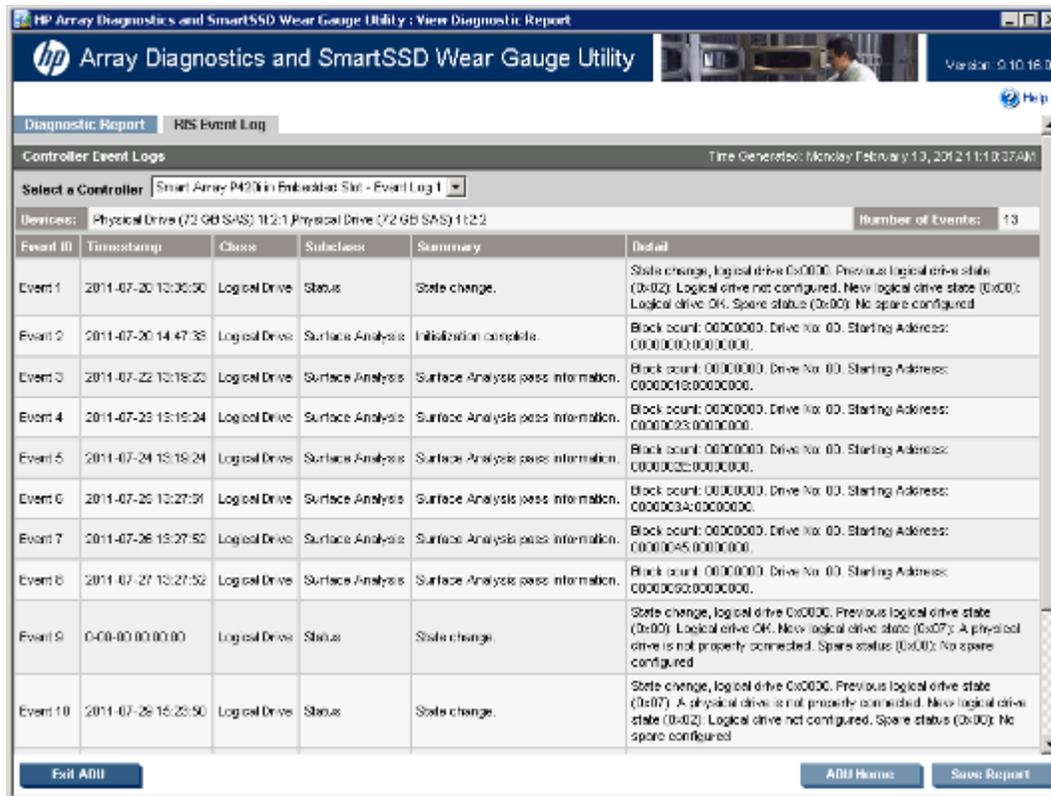
1. Launch the GUI ("Launching the utility in GUI mode" on page 95).
2. Select **View Diagnostic Report**.

The utility generates and displays the report.



The diagnostic report includes an additional **RIS Event Log** tab. Smart Array controllers store configuration data in the RAID information sector of all physical drives. The RIS Event Log reads the data and displays the decoded change logs.

3. Click **RIS Event Log**.



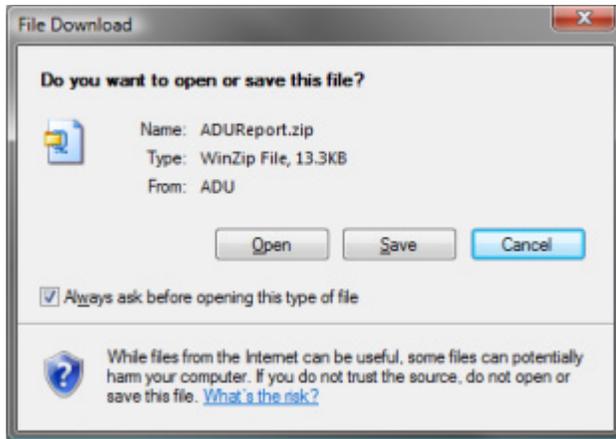
The log appears.

4. After reviewing the report, do one of the following:
- Click **Save Report**, and then save the generated file.
 - Click **ADU Home**, and then perform additional tasks.
 - Click **Exit ADU**.

Generating the diagnostic report

1. Launch the GUI ("Launching the utility in GUI mode" on page 95).
2. Select **Generate Diagnostic Report**.

The utility generates the report, and then displays the **File Download** dialog box.



3. Open or save the file.
4. When you are finished with the file, do one of the following:
 - o Click **Save Report**, and then save the generated file.
 - o Click **ADU Home**, and then perform additional tasks.
 - o Click **Exit ADU**.

Identifying and viewing diagnostic report files

The diagnostic report output archive contains the following files:

- ADUReport.txt—Diagnostic report in text format
- ADUReport.xml—Diagnostic report in XML format
- ADUReportViewer.htm—HTML viewer for XML diagnostic report
- SlotX.txt (SlotX.old)—Controller serial output log

The serial output log file(s) are only available if the HP Smart Array SAS/SATA Event Notification Service is installed and running.

To view the diagnostic report files in a browser:

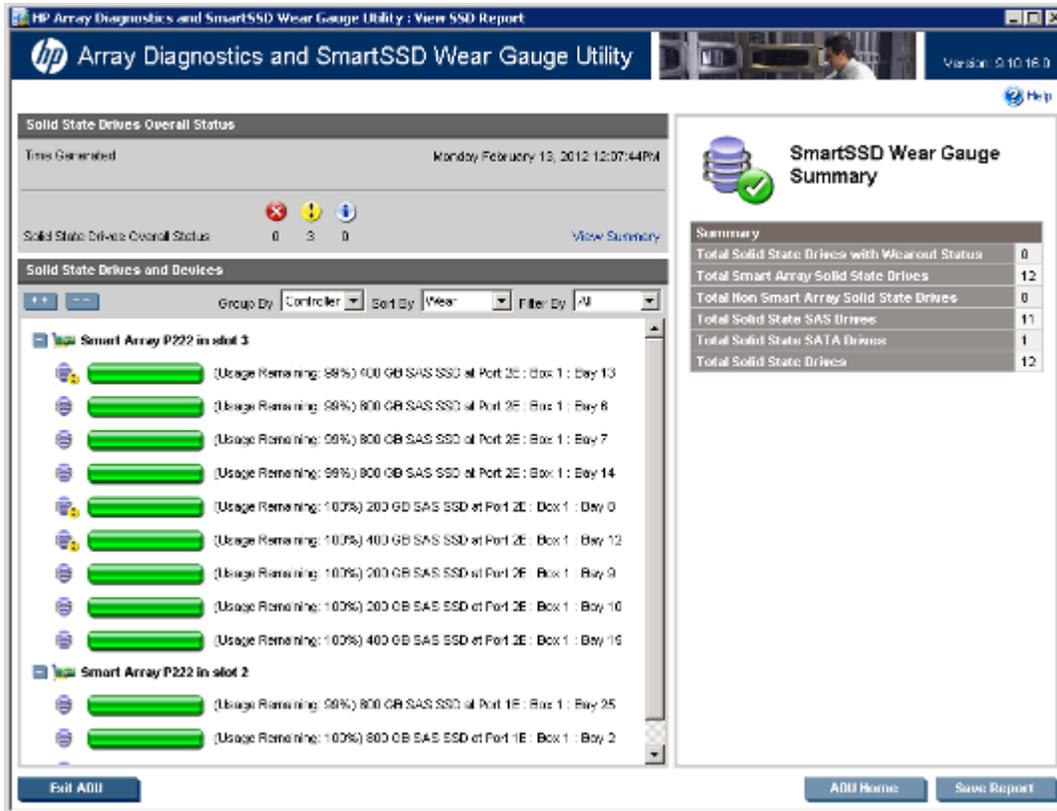
1. Extract ADUReportViewer.htm to a directory.
2. Open ADUReportViewer.htm in the browser.

SmartSSD Wear Gauge report procedures

Viewing the SmartSSD Wear Gauge report

1. Launch the GUI ("Launching the utility in GUI mode" on page 95).
2. Select **View SmartSSD Wear Gauge Report**.

The utility generates and displays the report.

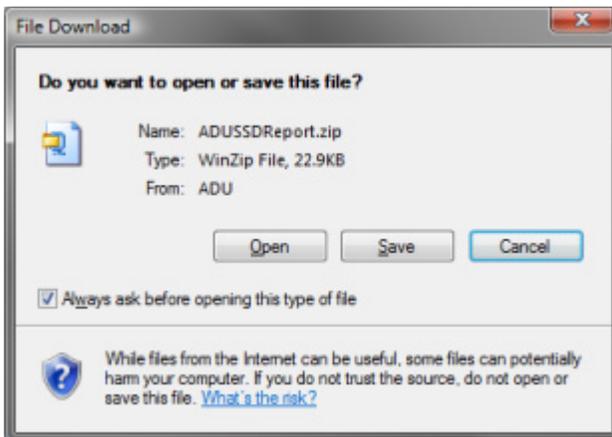


3. After reviewing the report, do one of the following:
 - o Click **Save Report**, and then save the generated file.
 - o Click **ADU Home**, and then perform additional tasks.
 - o Click **Exit ADU**.

Generating the SmartSSD Wear Gauge report

1. Launch the GUI ("Launching the utility in GUI mode" on page 95).
2. Select **Generate SmartSSD Wear Gauge Report**.

The utility generates the report, and then displays the **File Download** dialog box.



3. Open or save the file.
4. After reviewing the report, do one of the following:
 - Click **Save Report**, and then save the generated file.
 - Click **ADU Home**, and then perform additional tasks.
 - Click **Exit ADU**.

Identifying and viewing SmartSSD Wear Gauge report files

The SmartSSD Wear Gauge report output archive contains the following files:

- `SmartSSDWearGaugeReport.txt`—SmartSSD wear gauge report in text format
- `SmartSSDWearGaugeReport.json`—SmartSSD wear gauge report in JSON format
- `SmartSSDWearGaugeReport.htm`—HTML viewer for the JSON wear gauge report

To view the SmartSSD Wear Gauge report files in a browser:

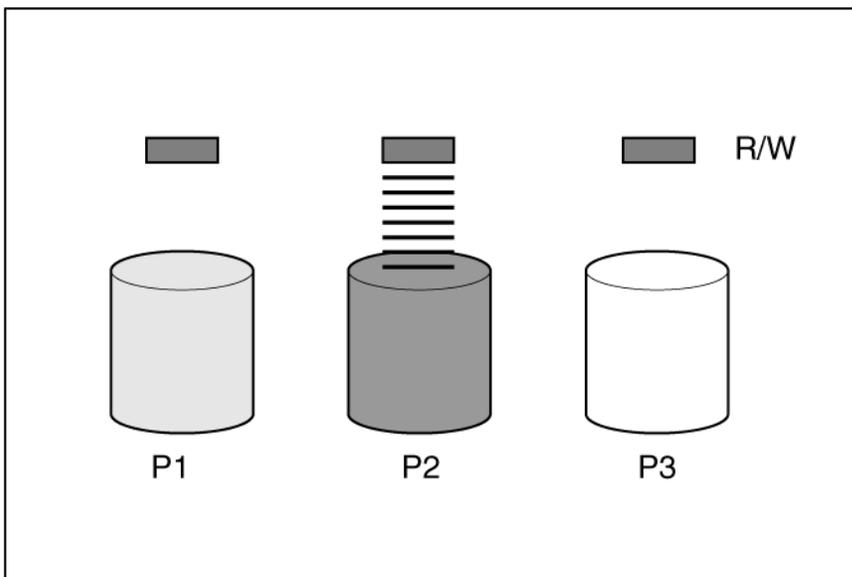
1. Extract the following files to a directory:
 - `SmartSSDWearGaugeReport.json`
 - `SmartSSDWearGaugeReport.htm`All files must reside in the same directory.
2. Open `SmartSSDWearGaugeReport.htm` in the browser.

Drive arrays and fault-tolerance methods

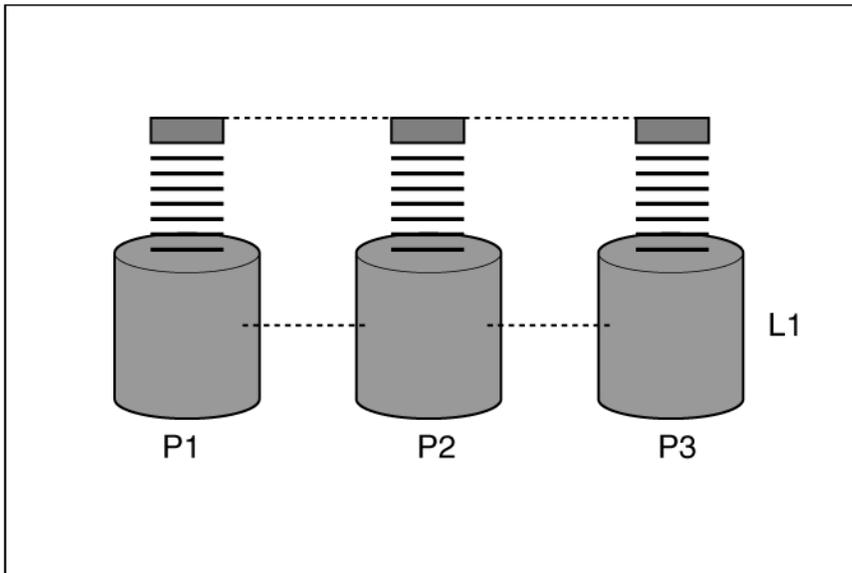
Drive arrays

The capacity and performance of a single physical (hard) drive is adequate for home users. However, business users demand higher storage capacities, higher data transfer rates, and greater protection against data loss when drives fail.

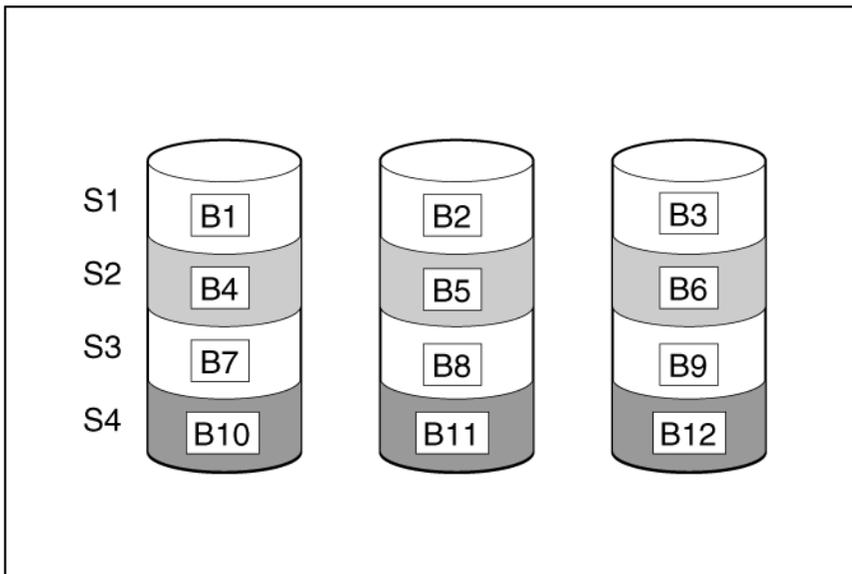
Connecting extra physical drives (P_n in the figure) to a system increases the total storage capacity but has no effect on the efficiency of read/write (R/W) operations. Data can still be transferred to only one physical drive at a time.



With an array controller installed in the system, the capacity of several physical drives can be combined into one or more virtual units called **logical drives** (also called **logical volumes** and denoted by L_n in the figures in this section). Then, the read/write heads of all the constituent physical drives are active simultaneously, reducing the total time required for data transfer.



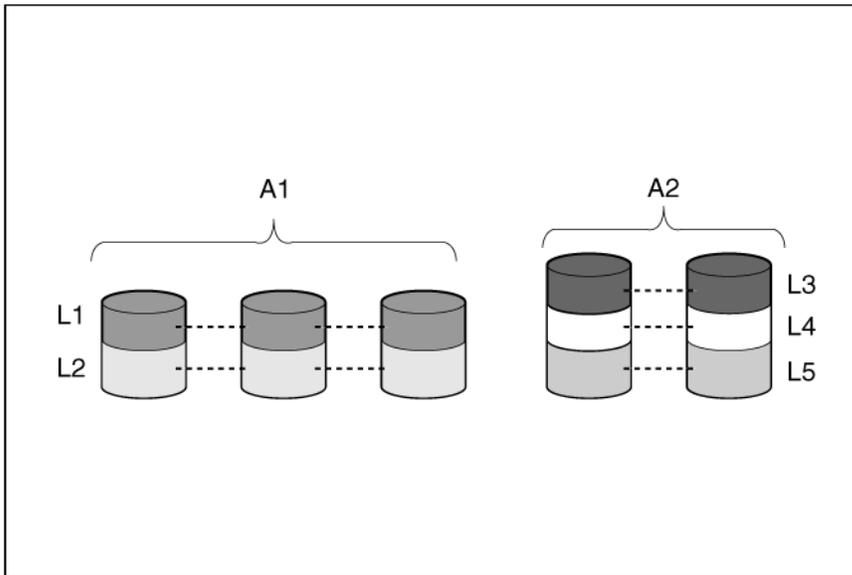
Because the read/write heads are active simultaneously, the same amount of data is written to each drive during any given time interval. Each unit of data is called a **block** (denoted by B_n in the figure), and adjacent blocks form a set of data **stripes** (S_n) across all the physical drives that comprise the logical drive.



For data in the logical drive to be readable, the data block sequence must be the same in every stripe. This sequencing process is performed by the array controller, which sends the data blocks to the drive write heads in the correct order.

A natural consequence of the striping process is that each physical drive in a given logical drive will contain the same amount of data. If one physical drive has a larger capacity than other physical drives in the same logical drive, the extra capacity is wasted because it cannot be used by the logical drive.

The group of physical drives containing the logical drive is called a **drive array**, or just **array** (denoted by A_n in the figure). Because all the physical drives in an array are commonly configured into just one logical drive, the term array is often used as a synonym for logical drive. However, an array can contain several logical drives, each of a different size.



Each logical drive in an array is distributed across all of the physical drives within the array. A logical drive can also extend across more than one port on the same controller, but it cannot extend across more than one controller.

Drive failure, although rare, is potentially catastrophic. For arrays that are configured as shown in the previous figure, failure of any physical drive in the array causes every logical drive in the array to suffer irretrievable data loss. To protect against data loss due to physical drive failure, logical drives are configured with **fault tolerance** ("[Fault-tolerance methods](#)" on page 105).

For any configuration except RAID 0, further protection against data loss can be achieved by assigning a drive as an **online spare** (or **hot spare**). This drive contains no data and is connected to the same controller as the array. When any other physical drive in the array fails, the controller automatically rebuilds information that was originally on the failed drive to the online spare. The system is thus restored to full RAID-level data protection, although it now no longer has an online spare. (However, in the unlikely event that another drive in the array fails while data is being rewritten to the spare, the logical drive will still fail.)

When you configure an online spare, it is automatically assigned to all logical drives in the same array. Additionally, you do not need to assign a separate online spare to each array. Instead, you can configure one hard drive to be the online spare for several arrays if the arrays are all on the same controller.

Effects of a hard drive failure on logical drives

When a drive fails, all logical drives that are in the same array are affected. Each logical drive in an array might be using a different fault-tolerance method, so each logical drive can be affected differently.

- RAID 0 configurations do not tolerate drive failure. If any physical drive in the array fails, all RAID 0 logical drives in the same array also fail.
- RAID 10 configurations tolerate multiple drive failures if no failed drives are mirrored to one another.
- RAID 5 configurations tolerate one drive failure.

- RAID 50 configurations tolerate one failed drive in each parity group.
- RAID 6 configurations tolerate two failed drives at a given time.
- RAID 60 configurations tolerate two failed drives in each parity group.
- RAID 1 (ADM) and RAID 10 (ADM) configurations tolerate multiple drive failures if no more than two drives, mirrored to one another, fail.

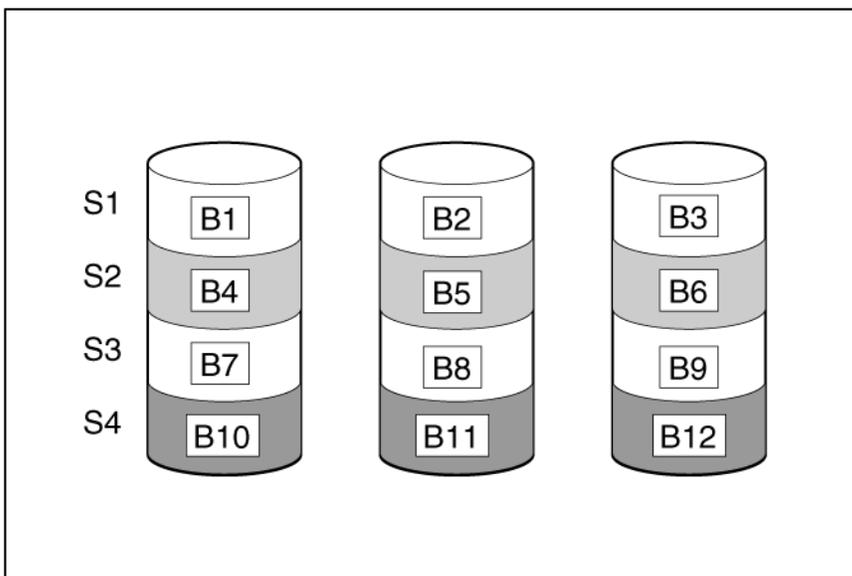
Fault-tolerance methods

Several fault-tolerance methods exist. Those most often used with Smart Array controllers are hardware-based RAID methods.

Two alternative fault-tolerance methods that are sometimes used are also described ("[Alternative fault-tolerance methods](#)" on page 113). However, hardware-based RAID methods provide a much more robust and controlled fault-tolerance environment, so these alternative methods are seldom used.

RAID 0—No fault tolerance

A RAID 0 configuration provides data striping, but there is no protection against data loss when a drive fails. However, it is useful for rapid storage of large amounts of noncritical data (for printing or image editing, for example) or when cost is the most important consideration.



Advantages:

- Has the highest write performance of all RAID methods.
- Has the lowest cost per unit of stored data of all RAID methods.
- All drive capacity is used to store data (none is needed for fault tolerance).

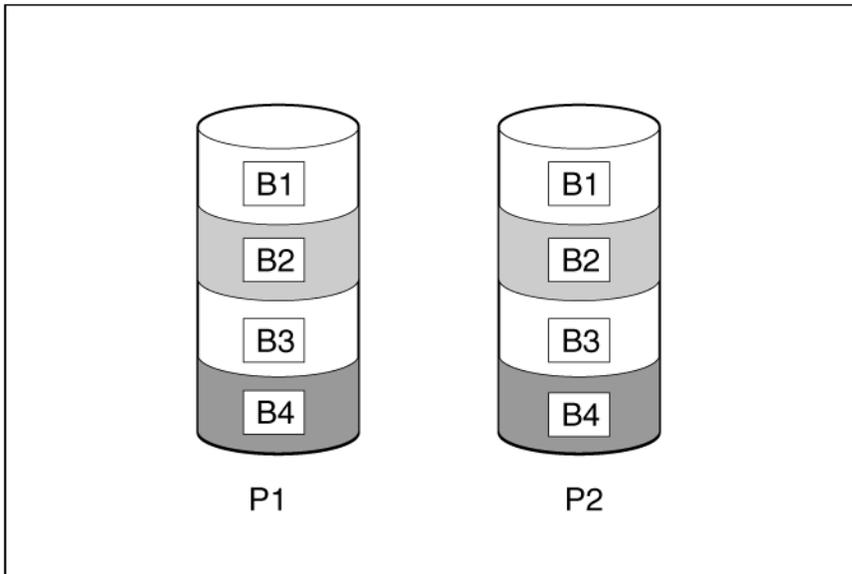
Disadvantages:

- All data on the logical drive is lost if a physical drive fails.
- Cannot use an online spare.
- Can only preserve data by backing it up to external drives.

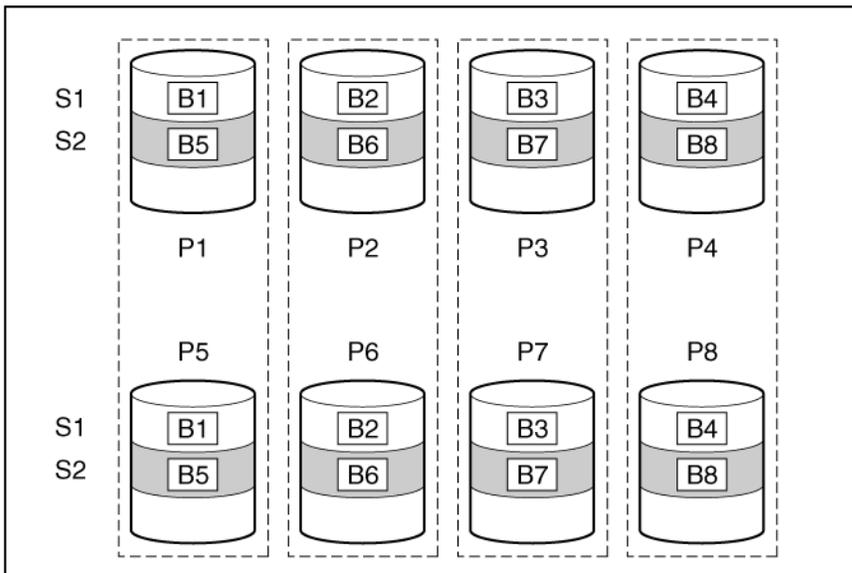
RAID 1 and RAID 1+0 (RAID 10)

In RAID 1 and RAID 1+0 (RAID 10) configurations, data is duplicated to a second drive.

When the array contains only two physical drives, the fault-tolerance method is known as RAID 1.



When the array has more than two physical drives, drives are mirrored in pairs, and the fault-tolerance method is known as RAID 1+0 or RAID 10.



In each mirrored pair, the physical drive that is not busy answering other requests answers any read requests that are sent to the array. This behavior is called **load balancing**. If a physical drive fails, the remaining drive in the mirrored pair can still provide all the necessary data. Several drives in the array can fail without incurring data loss, as long as no two failed drives belong to the same mirrored pair.

This fault-tolerance method is useful when high performance and data protection are more important than the cost of physical drives.

Advantages:

- This method has the second highest read performance of any fault-tolerant configuration.
- No data is lost when a drive fails, as long as no failed drive is mirrored to another failed drive.
- Up to half of the physical drives in the array can fail.

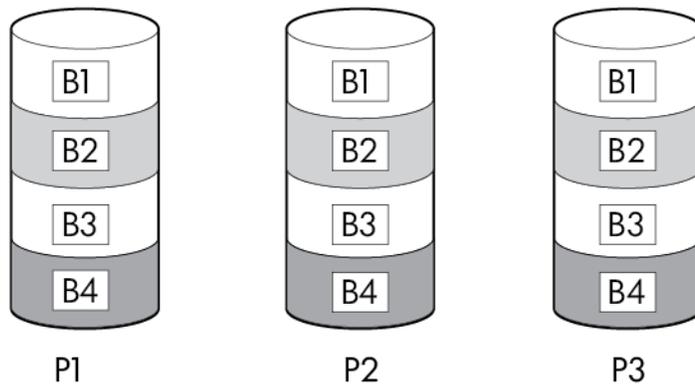
Disadvantages:

- This method is expensive, because many drives are needed for fault tolerance.
- Only half of the total drive capacity is usable for data storage.

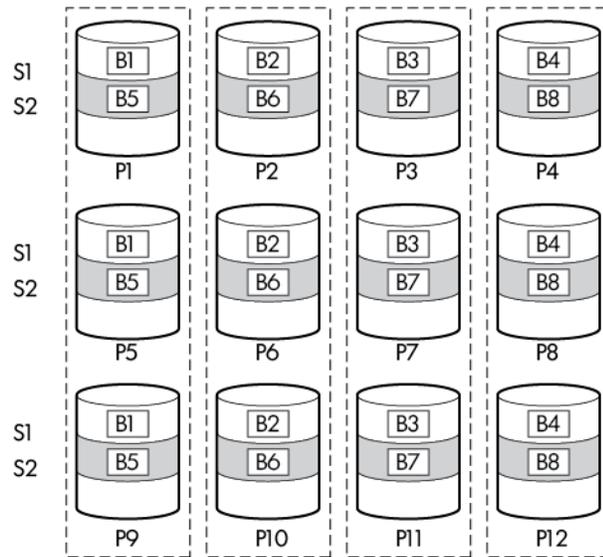
RAID 1 (ADM) and RAID 10 (ADM)

In RAID 1 (ADM) and RAID 10 (ADM) configurations, data is duplicated to two additional drives.

When the array contains only three physical drives, the fault-tolerance method is known as RAID 1 (ADM).



When the array has more than three physical drives, drives are mirrored in trios, and the fault-tolerance method is known as RAID 10 (ADM).



In each mirrored trio, the physical drives that are not busy answering other requests answer any read requests that are sent to the array. This behavior is called **load balancing**. If a physical drive fails, the remaining two drives in the mirrored trio can still provide all the necessary data. Several drives in the array can fail without incurring data loss, as long as no three failed drives belong to the same mirrored trio.

This fault-tolerance method is useful when high performance and data protection are more important than the cost of physical drives.

Advantages:

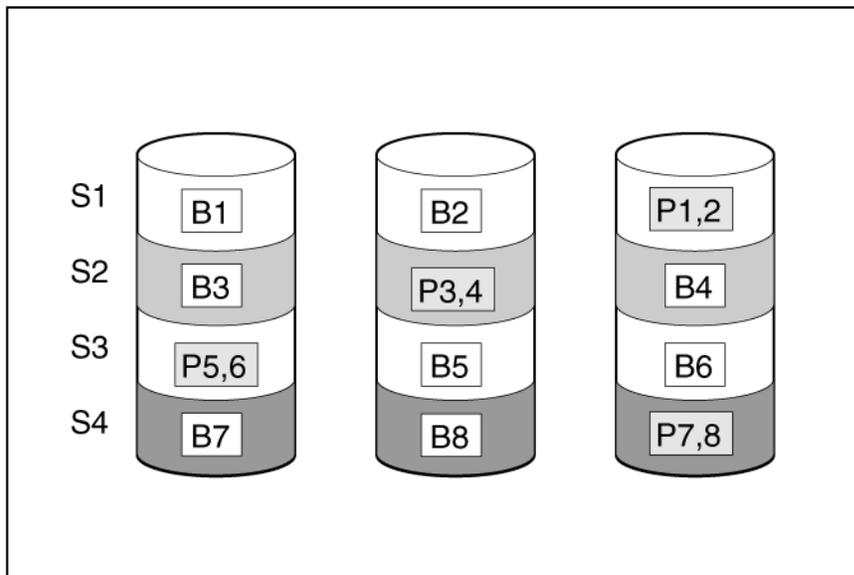
- This method has the highest read performance of any fault-tolerant configuration.
- No data is lost when two drives fail, as long as no two failed drives are mirrored to another failed drive.
- Up to two-thirds of the physical drives in the array can fail.

Disadvantages:

- This method is expensive, because many drives are needed for fault tolerance.
- Only one-third of the total drive capacity is usable for data storage.

RAID 5—distributed data guarding

In a RAID 5 configuration, data protection is provided by **parity data** (denoted by $P_{x,y}$ in the figure). This parity data is calculated stripe by stripe from the user data that is written to all other blocks within that stripe. The blocks of parity data are distributed evenly over every physical drive within the logical drive.



When a physical drive fails, data that was on the failed drive can be calculated from the remaining parity data and user data on the other drives in the array. This recovered data is usually written to an online spare in a process called a **rebuild**.

This configuration is useful when cost, performance, and data availability are equally important.

Advantages:

- Has high read performance.
- Data is not lost if one physical drive fails.
- More drive capacity is usable than with RAID 1+0—parity information requires only the storage space equivalent to one physical drive.

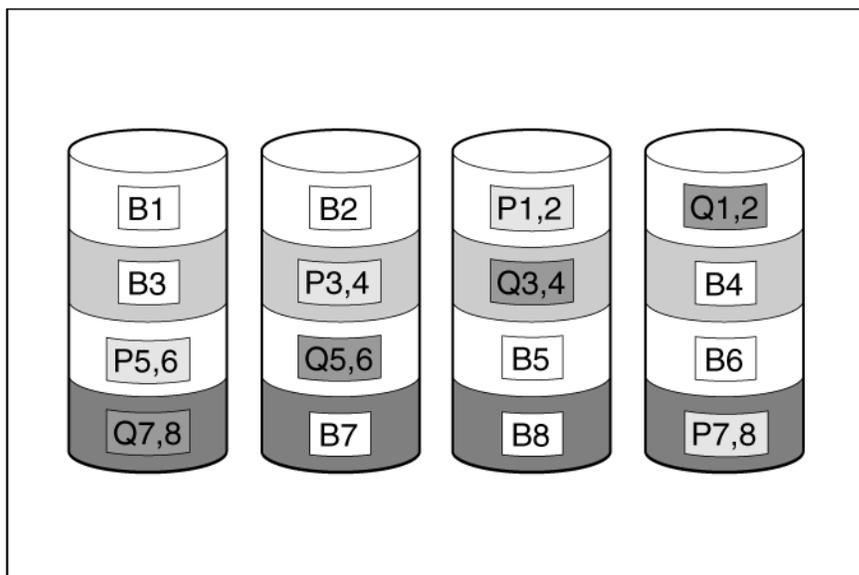
Disadvantages:

- Has relatively low write performance.
- Data is lost if a second drive fails before data from the first failed drive is rebuilt.

RAID 6 (ADG)—Advanced Data Guarding

NOTE: Not all controllers support RAID 6 (ADG).

RAID 6 (ADG), like RAID 5, generates and stores parity information to protect against data loss caused by drive failure. With RAID 6 (ADG), however, two different sets of parity data are used (denoted by $P_{x,y}$ and $Q_{x,y}$ in the figure), allowing data to still be preserved if two drives fail. Each set of parity data uses a capacity equivalent to that of one of the constituent drives.



This method is most useful when data loss is unacceptable but cost is also an important factor. The probability that data loss will occur when an array is configured with RAID 6 (ADG) is less than it would be if it was configured with RAID 5.

Advantages:

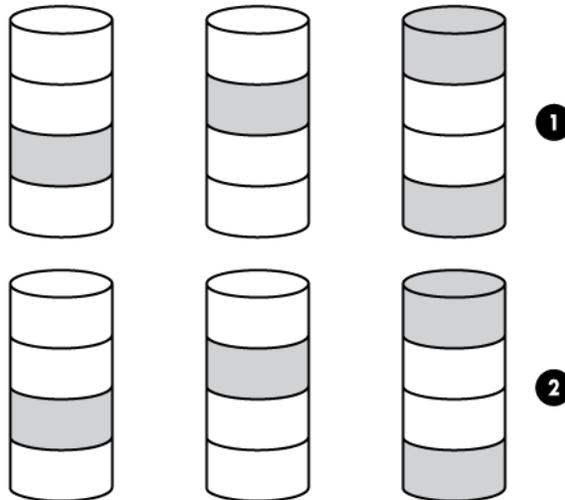
- This method has a high read performance.
- This method allows high data availability—Any two drives can fail without loss of critical data.
- More drive capacity is usable than with RAID 1+0—Parity information requires only the storage space equivalent to two physical drives.

Disadvantages:

The main disadvantage of RAID 6 (ADG) is a relatively low write performance (lower than RAID 5) because of the need for two sets of parity data.

RAID 50

RAID 50 is a nested RAID method in which the constituent hard drives are organized into several identical RAID 5 logical drive sets (parity groups). The smallest possible RAID 50 configuration has six drives organized into two parity groups of three drives each.



For any given number of hard drives, data loss is least likely to occur when the drives are arranged into the configuration that has the largest possible number of parity groups. For example, four parity groups of three drives are more secure than three parity groups of four drives. However, less data can be stored on the array with the larger number of parity groups.

RAID 50 is particularly useful for large databases, file servers, and application servers.

Advantages:

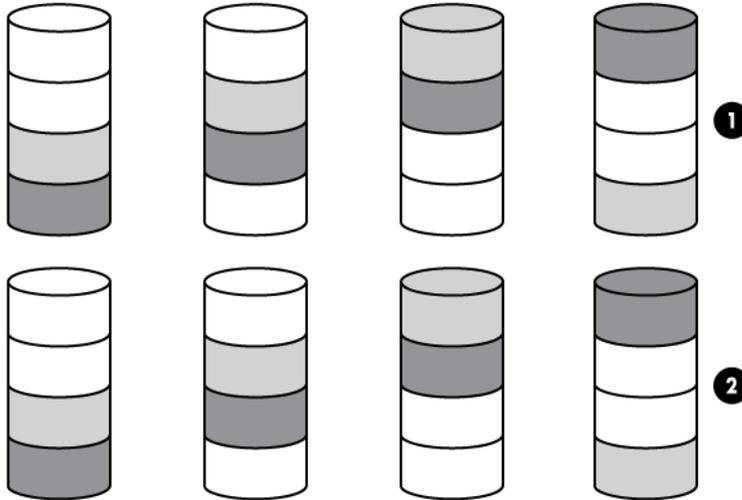
- Higher performance than for RAID 5, especially during writes.
- Better fault tolerance than either RAID 0 or RAID 5.
- Up to n physical drives can fail (where n is the number of parity groups) without loss of data, as long as the failed drives are in different parity groups.

Disadvantages:

- All data is lost if a second drive fails in the same parity group before data from the first failed drive has finished rebuilding.
- A greater percentage of array capacity is used to store redundant or parity data than with non-nested RAID methods.

RAID 60

RAID 60 is a nested RAID method in which the constituent hard drives are organized into several identical RAID 6 logical drive sets (parity groups). The smallest possible RAID 60 configuration has eight drives organized into two parity groups of four drives each.



For any given number of hard drives, data loss is least likely to occur when the drives are arranged into the configuration that has the largest possible number of parity groups. For example, five parity groups of four drives are more secure than four parity groups of five drives. However, less data can be stored on the array with the larger number of parity groups.

RAID 60 is particularly useful for data archives and high-availability solutions.

Advantages:

- Higher performance than for RAID 6, especially during writes.
- Better fault tolerance than either RAID 0 or RAID 6.
- Up to $2n$ physical drives can fail (where n is the number of parity groups) without loss of data, as long as no more than two failed drives are in the same parity group.

Disadvantages:

- All data is lost if a third drive in a parity group fails before one of the other failed drives in the parity group has finished rebuilding.
- A greater percentage of array capacity is used to store redundant or parity data than with non-nested RAID methods.

Comparing the hardware-based RAID methods

Not all controllers support all RAID levels. To support some RAID levels, HP Smart Array G6 and G7 controllers require SAAP ("[About SAAP](#)" on page 8).

Item	RAID 0	RAID 1+0	RAID 5	RAID 6 (ADG)	RAID 1(0) (ADM)
Alternative name	Striping (no fault tolerance)	Mirroring	Distributed Data Guarding	Advanced Data Guarding	Advanced Data Mirroring
Formula for number of drives usable for data (n = total number of drives in array)	n	$n/2$	$n-1$	$n-2$	$n/3$
Percentage of drive space usable*	100%	50%	67% to 93%	50% to 96%	33%
Minimum number of physical drives	1	2	3	4	3
Tolerates failure of one physical drive	No	Yes	Yes	Yes	Yes
Tolerates simultaneous failure of more than one physical drive	No	Only if no two failed drives are in the same mirrored pair	No	Yes	Only if no three drives are in the same mirror group
Read performance	High	High	High	High	High
Write performance	High	Medium	Low	Low	Medium
Relative cost	Low	High	Medium	Medium	Very high

*Values for the percentage of drive space usable are calculated with these assumptions: (1) all physical drives in the array have the same capacity; (2) online spares are not used; (3) no more than 14 physical drives are used per array for RAID 5; and (4) no more than 56 drives are used with RAID 6 (ADG).

Selecting a RAID method

Not all controllers support all RAID levels. To determine the RAID capabilities of your controller, see the model-specific information for your controller on the HP website (<http://www.hp.com/products/smartarray>).

Most important criterion	Also important	Suggested RAID level
Fault tolerance	Cost effectiveness I/O performance	RAID 6 RAID 10 (ADM), RAID 1+0, RAID 50, RAID 60
Cost effectiveness	Fault tolerance I/O performance	RAID 6 RAID 5 (RAID 0 if fault tolerance is not required)
I/O performance	Cost effectiveness Fault tolerance	RAID 5 (RAID 0 if fault tolerance is not required) RAID 10 (ADM), RAID 1+0, RAID 50, RAID 60

Alternative fault-tolerance methods

Your operating system may also support software-based RAID or controller duplexing.

- **Software-based RAID** resembles hardware-based RAID, except that the operating system works with logical drives as if they were physical drives. To protect against data loss caused by physical drive failure, each logical drive must be in a different array from the others.
- **Controller duplexing** uses two identical controllers with independent, identical sets of drives containing identical data. In the unlikely event of a controller failure, the remaining controller and drives will service all requests.

Neither of these alternative fault-tolerance methods supports online spares or automatic data recovery, nor do they support auto-reliability monitoring or interim data recovery.

If you decide to use one of these alternative methods, configure your arrays with RAID 0 for maximum storage capacity and refer to your operating system documentation for further implementation details.

Diagnosing array problems

Diagnostic tools

To troubleshoot array problems and generate feedback about arrays, use the following diagnostic tools:

- **ACU**

For more recent products, array diagnostics is available with ACU v8.28.13.0 and later. This utility is available on the SmartStart CD in the controller kit and also on the HP website (<http://www.hp.com/support>). For more information about ACU, see "About ACU (on page 10)." For more information about error messages, see the *HP ProLiant Servers Troubleshooting Guide*.
- **ADU**

For products that support SmartStart v8.25 and earlier, this utility is available on the SmartStart CD in the controller kit and also on the HP website (<http://www.hp.com/support>). When prompted for product information, enter the server model name. For more information about the meanings of the various ADU error messages, see the *HP ProLiant Servers Troubleshooting Guide*.
- **HP Array Diagnostics and SmartSSD Wear Gauge Utility**

This standalone diagnostic utility provides configuration and error information about array controllers, storage enclosures, HBAs, drive cages, logical drives, physical drives, and tape drives. For any supported SSDs, the utility provides current usage level and remaining expected lifetime. For more information, see "HP Array Diagnostics and SmartSSD Wear Gauge Utility (on page 92)."
- **Event Notification Service**

This utility reports array events to the Microsoft® Windows® system event log and IML. You can obtain the utility from the SmartStart CD or the HP website (<http://www.hp.com/support>). When prompted for product information, enter the server model name.
- **HP Insight Diagnostics**

HP Insight Diagnostics is a tool that displays information about the system hardware configuration and performs tests on the system and its components, including drives if they are connected to Smart Array controllers. This utility is available on the SmartStart CD and also on the HP website (<http://www.hp.com/servers/diags>).
- **POST messages**

Smart Array controllers produce diagnostic error messages (POST messages) at reboot. Many POST messages suggest corrective actions. For more information about POST messages, see the *HP ProLiant Servers Troubleshooting Guide*.

Troubleshooting resources

HP ProLiant G7 (and earlier) products

The *HP ProLiant Servers Troubleshooting Guide* provides procedures for resolving common problems and comprehensive courses of action for fault isolation and identification, error message interpretation, issue resolution, and software maintenance on ProLiant servers and server blades. This guide includes

problem-specific flowcharts to help you navigate complex troubleshooting processes. To view the guide, select a language:

- English (http://www.hp.com/support/ProLiant_TSG_en)
- French (http://www.hp.com/support/ProLiant_TSG_fr)
- Italian (http://www.hp.com/support/ProLiant_TSG_it)
- Spanish (http://www.hp.com/support/ProLiant_TSG_sp)
- German (http://www.hp.com/support/ProLiant_TSG_gr)
- Dutch (http://www.hp.com/support/ProLiant_TSG_nl)
- Japanese (http://www.hp.com/support/ProLiant_TSG_jp)

HP ProLiant Gen8 products

The *HP ProLiant Gen8 Troubleshooting Guide, Volume I: Troubleshooting* provides procedures for resolving common problems and comprehensive courses of action for fault isolation and identification, issue resolution, and software maintenance on ProLiant servers and server blades. To view the guide, select a language:

- English (http://www.hp.com/support/ProLiant_TSG_v1_en)
- French (http://www.hp.com/support/ProLiant_TSG_v1_fr)
- Spanish (http://www.hp.com/support/ProLiant_TSG_v1_sp)
- German (http://www.hp.com/support/ProLiant_TSG_v1_gr)
- Japanese (http://www.hp.com/support/ProLiant_TSG_v1_jp)
- Simplified Chinese (http://www.hp.com/support/ProLiant_TSG_v1_sc)

The *HP ProLiant Gen8 Troubleshooting Guide, Volume II: Error Messages* provides a list of error messages and information to assist with interpreting and resolving error messages on ProLiant servers and server blades. To view the guide, select a language:

- English (http://www.hp.com/support/ProLiant_EMG_v1_en)
- French (http://www.hp.com/support/ProLiant_EMG_v1_fr)
- Spanish (http://www.hp.com/support/ProLiant_EMG_v1_sp)
- German (http://www.hp.com/support/ProLiant_EMG_v1_gr)
- Japanese (http://www.hp.com/support/ProLiant_EMG_v1_jp)
- Simplified Chinese (http://www.hp.com/support/ProLiant_EMG_v1_sc)

Acronyms and abbreviations

ACU

Array Configuration Utility

ADG

Advanced Data Guarding (also known as RAID 6)

ADM

Advanced Data Mirroring

ADU

Array Diagnostics Utility

CPQONLIN

NetWare Online Array Configuration Utility

DHCP

Dynamic Host Configuration Protocol

HBA

host bus adapter

ORCA

Option ROM Configuration for Arrays

POST

Power-On Self Test

RBSU

ROM-Based Setup Utility

RIS

reserve information sector

SAAP

Smart Array Advanced Pack

SAS

serial attached SCSI

SATA

serial ATA

SSD

solid-state drive

SSP

Selective Storage Presentation

TFTP

Trivial File Transfer Protocol

WBEM

Web-Based Enterprise Management

WWN

World Wide Name

Documentation feedback

HP is committed to providing documentation that meets your needs. To help us improve the documentation, send any errors, suggestions, or comments to Documentation Feedback (<mailto:docsfeedback@hp.com>). Include the document title and part number, version number, or the URL when submitting your feedback.

Index

A

- action mode, ACU scripting 72
- ACU (Array Configuration Utility) 10
- ACU GUI, methods for opening 19
- ACU GUI, procedure overview 30
- ACU help 28
- ACU scripting 72, 76, 83
- additional information 114
- ADG (advanced data guarding) 109
- ADU (Array Diagnostic Utility) 114
- ADU Remote Service Mode 95
- advanced configuration tasks, support for 6
- advanced data guarding (ADG) 109
- array accelerator, enabling or disabling, ACU CLI 67
- array category options, ACU scripting 76
- array concepts 102
- array configuration utilities, features 5
- Array Configuration Utility (ACU) 10
- array configuration, copying 68, 69
- Array Diagnostic Utility (ADU) 114
- array, creating, ACU GUI 30, 44
- array, creating, ACU scripting 72, 76
- array, expanding, ACU CLI 61
- array, expanding, ACU GUI 30
- array, expanding, ACU scripting 76
- array, healing 39
- array, moving 62, 76
- array, moving, ACU CLI 62
- array, moving, ACU GUI 30
- array, moving, ACU scripting 76
- array, replacing 39, 63
- array, shrinking 30, 62, 76
- array, shrinking, ACU CLI 62
- array, shrinking, ACU GUI 30
- array, shrinking, ACU scripting 76
- array, specifying, ACU scripting 76
- ArrayAccelerator 79
- auto-configuration process 9

B

- boot controller, setting 51
- boot volume, setting 52

C

- cache ratio, ACU CLI 67
- cache ratio, ACU scripting 75
- CacheState 73
- capturing configurations 68
- choosing a RAID level 112
- ClearConfigurationWithDataLoss 74
- clearing a configuration 74
- CLI (Command Line Interface) 45
- CLI abbreviations 48
- CLI syntax 46
- CLI, accessing 10, 15, 95
- Command Line Interface (CLI) 45
- command mode, opening ACU in 46
- command variable, ACU CLI 47
- comparison of RAID features 111
- comparison of the utilities 5
- Configuration screen 20
- Configuration tasks 28, 30
- configuration tasks, advanced, support for 6
- configuration tasks, performing 30, 42, 44, 51, 68
- configuration tasks, standard, support for 6
- configuration utilities, comparison of 5
- configuration utilities, description of 5
- configuration utility, choosing 5
- connection name 83
- connection profile 83
- ConnectionName 83
- console mode, opening ACU in 46
- control category, ACU scripting 72
- controller duplexing 112
- controller name 60
- controller options, ACU scripting 72
- controller, specifying, ACU scripting 73
- criteria for selecting a RAID level 112

D

- data protection methods 105, 112
- data striping 102, 105
- default settings 69
- DeleteLicenseKey 74
- deleting a device 53

- device information, obtaining 49
- devices, discovering 19
- devices, identifying 53
- diagnostic report 39, 40, 53, 96, 98, 99
- diagnostic tools 114
- diagnostics 22, 39, 40, 53
- Diagnostics screen 22
- Diagnostics tasks 39, 40
- disabling drive cache 67
- disabling the array accelerator, ACU CLI 67
- disabling the array accelerator, ACU scripting 79
- disabling the redundant controller 65
- distributed data guarding 108
- DPOEnable 75
- drive array concepts 102
- drive cache, enabling or disabling 67, 74
- drive information 58, 59
- drive mirroring 106, 107
- drive type, ACU scripting 77
- DriveType 77
- DriveWriteCache 74
- duplexing 112

E

- ElevatorSortEnable 75
- enabling drive cache 67
- enabling the array accelerator, ACU CLI 67
- enabling the array accelerator, ACU GUI 30, 42
- enabling the array accelerator, ACU scripting 79
- enclosure information 58
- erase a drive 54
- error messages 87
- example CLI script, creating logical drive 56
- exit on error 68
- exiting the ACU GUI 19
- expand priority, ACU CLI 66
- expand priority, ACU scripting 75
- expanding an array, ACU CLI 61
- expanding an array, ACU GUI 30, 42
- expanding an array, ACU scripting 76
- ExpandPriority 75
- extending logical drive capacity 63, 81

F

- F5 prompt 11
- failed logical drive, re-enabling 66
- fault-tolerance methods 102, 105
- features of the utilities 5
- forced parameter 48

G

- GUI, accessing 10, 15, 95

H

- HBA category options, ACU scripting 83
- HBA WWN 83
- HBA_WW_ID 83
- healing an array 39
- help command 51
- help resources 28
- hiding warning prompts, ACU CLI 48
- host mode 83
- HostMode 83
- HP SmartCaching 34, 35, 59

I

- I/O path, modifying 64, 65, 74
- identifying devices 53
- information about devices, obtaining 49
- Input script, using 68
- installing ACU 10
- Intelligent Provisioning 11
- IRPEnable 75
- ISO image 11, 12, 13
- ISO image path, specifying 14

J

- Join 77

K

- keyword abbreviations 48

L

- languages 5
- LEDs, activating 53
- license key 33, 54, 74
- LicenseKey 74
- local application, using ACU as 11, 16
- logical drive capacity extension 63, 81
- logical drive, caching 79
- logical drive, creating, ACU CLI 55
- logical drive, creating, ACU GUI 30, 42, 44
- logical drive, description of 102
- logical drive, failed, re-enabling 66
- logical drive, migrating, ACU CLI 64
- logical drive, migrating, ACU GUI 30
- logical drive, migrating, ACU scripting 80, 82

logical drive, moving, ACU CLI 57
logical drive, specifying, ACU scripting 79
LogicalDrive 79

M

MaxBoot setting 81
menu options, ACU GUI 30, 42
method mode, ACU scripting 72
methods for opening the ACU GUI 19
migrating stripe size or RAID level, ACU CLI 64
migrating stripe size or RAID level, ACU GUI 30
migrating stripe size or RAID level, ACU scripting 80, 82
mirrored arrays 36
mirrored drives 106, 107
MNPDelay 75
mounting virtual media 12, 13
moving an array 62, 76

N

navigating the interface 19
nested RAID 110, 111
NoBatteryWriteCache 74
Novell NetWare, configuration tool for 5
NumberOfParityGroups 79

O

offline deployment 10
online deployment 15
online spare, ACU CLI 60
online spare, ACU GUI 30, 42
online spare, ACU scripting 78
OnlineSpare 77, 78
opening the ACU GUI 19
operating system support 10
Option ROM Configuration for Arrays (ORCA) 9
options, scripting, list of 70
ORCA (Option ROM Configuration for Arrays) 9
overview of configuration utilities 5

P

parameters, in CLI, obtaining values of 47
parity groups 79, 110, 111
physical drive write cache, enabling or disabling 67
physical drives, adding, ACU CLI 61
physical drives, adding, ACU GUI 30, 42
POST error messages 114
PreferredPath 80

PreferredPathMode 74
prompts, hiding, ACU CLI 48
PXE-based deployment 13
PXElinux setup 13

Q

querying a device 47
QueueDepth 75

R

RAID 80
RAID level selection criteria 112
RAID levels 105, 106, 107, 108, 109, 110, 111
RAID levels, comparison of features 111
RAID, software-based 112
RaidArrayId 74
RAID-level migration, ACU CLI 64
RAID-level migration, ACU GUI 30, 42
RAID-level migration, ACU scripting 80
Rapid Parity Initialization 32, 59
ReadCache 75
read-write ratio, ACU CLI 67
read-write ratio, ACU GUI 30
read-write ratio, ACU scripting 75, 79
rebuild priority, ACU CLI 65
rebuild priority, ACU GUI 30
rebuild priority, ACU scripting 75
RebuildPriority 75
recombining a split mirrored array 37
redundant controller, disabling 65
redundant controller, modifying I/O path of 64, 65, 74, 80
redundant controller, preferred path settings for 64, 65, 74, 80
re-enabling a failed logical drive 66
registering a license key 33
re-mirroring a split mirror backup 38
remote server, configuring 17
remote service, using ACU as 18
renaming a controller 60
ReNumber 80
Repeat 81
repeat configuration 81
replacing an array 39, 63
required hardware 8
rescan 19, 54
ResourceVolumeOwner 81
running the ACU GUI 19

S

- sample script 69
- screen description 19, 20, 22, 25, 28
- script files 69
- scripting modes 68
- scripting options 70
- scripting syntax 69
- Sectors 81
- selecting a RAID level 112
- shorthand in CLI 48
- show (CLI command) 49
- shrinking an array 30, 62, 76
- ShrinkSize 81
- Size 81
- SizeBlocks 82
- Smart Array Advanced Pack (SAAP) 8
- SmartSSD Wear Gauge report 40, 99, 100, 101
- SmartStart CD as source of ACU 11
- software-based RAID 112
- spare activation mode 60
- spare drives, ACU CLI 60
- spare drives, ACU GUI 30, 42
- spare drives, ACU scripting 78
- Split 78
- split mirror backup 37
- splitting a mirrored array 37
- standard configuration tasks, support for 6
- standby controller, disabling 65
- stripe size migration, ACU CLI 64
- stripe size migration, ACU GUI 30, 42
- stripe size migration, ACU scripting 82
- stripe size values 82
- StripeSize 82
- striping data 102, 105
- StripSize 82
- supported operating systems 15
- supported tasks 6
- surface scan delay 66, 75
- surface scan mode 66
- SurfaceScanDelay 75
- SurfaceScanDelayExtended 75
- SurfaceScanMode 75
- syntax, CLI 46

T

- target device, setting 52
- target variable, ACU CLI 47
- tasks 28, 30, 39, 40
- tasks supported in each utility 6

- troubleshooting 114
- troubleshooting resources 114
- typical procedures, ACU CLI 51

U

- utilities, description of 5

V

- version command 51
- video performance optimization 55, 75

W

- warning messages 87
- warning prompts, hiding, ACU CLI 48
- wizards mode, ACU GUI 41, 42
- Wizards screen 25
- write cache, on physical drives, enabling or disabling 67
- WriteCache 75

X

- XML DTD 86
- XML input file 85
- XML output file 84
- XML support 84