Aspect Architecture Document

AAD System Firmware Upgrade

System Firmware Upgrade

Document: 45971-00
Preliminary Revision: P:A.3:A.2:A.2

Revision Date: 4/21/2010

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHANGES TO DOCUMENT

Section	Insertions	Deletions
1.1.2. Assumptions	2	2
1.1.4. Open Issues	1	0
1.2.1. Product Requirements	1	1

Source Document Information

Section 1. AAD System Firmware Upgrade

Type: overview
Document: 45971-00
Revision: A.3
Revision Date: 4/21/2010
Author(s): Arindam Banerjee

REVISION HISTORY

Revision	Description of Changes
A.1	Initial revision.
A.2	AAD as per staged approach
	Incorporated review comments. Updated the overview section.

Section 2.1. Element External Requirement

Type: requirement
Document: 45971-00
Revision: A.2
Revision Date: 4/21/2010
Author(s): Arindam Banerjee

REVISION HISTORY

Revision	Description of Changes
A.1	Initial revision.
A.2	Architectural changes as per staged approach

Section 2.2. Detailed Controller Firmware Architecture

Type: architecture
Document: 45971-00
Revision: A.2
Revision Date: 4/21/2010
Author(s): Arindam Banerjee

REVISION HISTORY

Revision	Description of Changes
A.1	Initial revision.
A.2	Architectural changes as per staged approach

1. AAD System Firmware Upgrade

1.1.Aspect Introduction

1.1.1.Aspect Description

The Orion Unified Storage Platform (USP) product which is intended to provide both file based access and block based access to storage within the same product is accomplished by integrating the NAS based Cougar product into the LSI Open Storage Architecture (OSA). The Orion product will also support block-only initiators. In other words, host machines or servers can also use the underlying storage provided by the system as block storage. The Orion product will be based on the Open Storage Architecture hosted on a Pikes Peak controller platform.

The Orion product runs on a virtualized environment wherein multiple guest Virtual Machines will be running on the platform managed by the Virtual Machine manager. This architecture aspect as defined in this document caters to the Serviceability requirements for the Orion product on a virtualized environment.

1.1.2.Assumptions

Following are the key assumptions made for Orion:

- It is assumed that the Block Virtualization Layer (BVL) functionality is integrated and running on the IO VM.
- ~~An extended DOMI~~ A new Remote Method Invocation (RMI) functionality will be available for inter-VM communications. This aspect will be covered in the hypervisor AAD.
- A staged approach will be used for releasing the Orion product. The initial release or Stage 1 of the product will be a block-only release running the ~~Flint~~ BVL based FW on a hypervisor environment on a Pikes Peak controller board. The array will be managed by the SANtricity management software.
- A file based release would be available during the second stage which is Stage 2 of the Orion product. This will contain the NAS Cougar product running as a different Virtual machine. The management side will be SMI-S based and will be managed by a new virtual machine called the Service VM.

1.1.3.Related Documents

Document Number	Name of Document
	Open Storage Architecture CAS
	Unified Storage Product Architecture Specification
44341-00	Hypervisor AAD – Virtual Machine Management
44327-00	Pikes Peak FPGA Management AAD

1.1.4.Open Issues

Following are the open issues in this revision of the document:

- New lockdown scenarios identified for the OSA based product will also necessitate new seven

segment display codes to be identified. Seven segment displays need to be identified for lockdown scenarios that occur due to VMM environment issues.

- The SYMbol command for powering off an array (PowerDownArray) attempts to quiesce the controllers, power off the drives and then power off the enclosures. However, in a OSA environment, the power supply is controlled by IOVM while a clean shutdown of the controller will require Domain0 to initiate the shutdown which includes quiescing its root file system and shutdown of the controller. However, in such a scenario, the controller cannot power off the enclosures.

1.2.Aspect High Level Requirements

The OSA based Orion product identifies the following Serviceability requirements. Each requirement will be identified as a separate element within the AAD.

- Provide a non-disruptive system SW/FW upgrade mechanism.
  - This feature is supposed to provide a non-disruptive mechanism to upgrade FW for the virtual machines including Domain0. It will provide the facility to upgrade the virtual machine kernel and also Xen upgrades. During upgrade, the array will continue to be "available" for use but in a non-optimal fashion.
  - The upgrade will be performed in 2 steps as is the case in our current RAID controllers. It will consist of a download step and an activation step.
  - Failure to upgrade the FW will result in a rollback of the system to the previously running version of the FW. However, the rollback is possible only if the upgrade fails and the changes as part of the upgrade is not committed. Once the changes are committed, it cannot rollback to the previous version of FW.
  - A single upgrade bundle (referred to as the Solution bundle) will be used for upgrading the OSA FW. The upgrade bundle will consist of image for all the virtual machines. Individual pieces of software/FW will not be allowed to be upgraded.
- Provide a unified Logging and log collection mechanism.
  - A centralized logging will be provided for the OSA platform. This feature attempts to colaesce the logs of all the virtual machines into a single log of events for the system.
  - The logs will be persisted for retrieval by the user.
  - Individual virtual machines will have their own set of traces. Traces will not be coalesced across virtual machines.
- Provide a unified Controller State Management mechanism across all Virtual Machines.
  - A unified controller state management facility will be provided for the OSA based FW. Some of the current controller states like Service Mode, Offline, Lockdown will need to be extended to run on a OSA based FW on a Pikes peak platform.
  - Some of the lockdown scenarios can also be initiated by Domain0 due to Xen failures or other VMM related failures. These failures will need to be handled.
  - The existing lockdown scenarios for out r RAID controllers as initiated by IOVM will be handled by IOVM itself. However, Domain0 will need to be notified of this scenario since the other virtual machines may either be suspended or be destroyed in cases of a lockdown. Wherever possible the lockdown information will be peered to the Alternate controller.
  - ~~Some of the controller offline and heldinreset scenarios will also changes. IOVM reset will no longer de-assert the ALT_RESET_OUT lines and will need to be handled differently for a OSA based platform.~~
- Provide system diagnose-ability. A set of diagnostics will be provided to diagnose the FRUs in the system like the HICs, the memory DIMMs, the flash drive and the controller base board itself.
  - As in XBB-2, a set of diagnostics will be provided to diagnose the FRUs in the system.
  - These diagnostics will be initiated through user commands with the controller in Service Mode.
- Extend ACS to OSA based platforms

- The ACS mechanism for a OSA based platform will be different to the current mechanism of ACS deployed for the current RAID controllers.
- Apart from the controller being foreign, the flash drive which contains the code images also could be from a ~~diferent~~ different array and contain a different version of the OSA FW.
- The mechanics of ACS will change since Domain0 will "own" the solution bundle for the OSA based FW. The image needs to be transferred by Domain0 from the Alternate controller' Domain0. Therefore, it will be transferred using the non-data path over the controller mid-plane.

1.2.1.Product Requirements

ClearQuest PR Number	Feature Name
LSIP200012459	Orion: Provide online system SW/FW upgrade
LSIP200012460	Orion: Provide Log and Event Reporting Management framework for multiple applications
~~LSIP200038450~~ LSIP200038540	Orion: Controller State Management
LSIP200038476	Orion: Extend ACS to OSA platform

1.2.2.Architectural Requirements

Each of the elements has requirements for the Orion product. The detailed requirements for each of these elements will be covered within the external requirements section for each of the elements.

1.3.Future Considerations

All architectural and design decisions for Stage 1 of the product will need to be compatible with the later stages of the product which includes introduction of the NAS Cougar product running within the Pikes Peak controller board.

2. System Firmware Upgrade

2.1. Element External Requirement

2.1.1.Element Functional Behavior Changes

2.1.1.1.Upgrading the OSA firmware

The firmware upgrade process for the BVL based RAID controllers consist of upgrading the RAID firmware running on the controller platform. However, a system firmware upgrade for a OSA based controller platform consists of upgrading the system firmware or software for all the virtual machines running on the controller. A firmware upgrade is considered successful only if all the virtual machines running on the platform can boot from the new version of the firmware and complete its initialization. Upgrade can also include upgrade of the virtual machine management software.

The OSA based firmware upgrade solution also introduces an atomic rollback mechanism wherein the firmware shall rollback to the previously running version in case all the virtual machines cannot be upgraded to the new version of the firmware.

2.1.2.Introduction

2.1.2.1.System Firmware Upgrade

As part of the OSA functionality, a non-disruptive system firmware upgrade utility will be provided which will enable users of the product to upgrade to a new version of the firmware. By non-disruptive it is meant that the storage array will continue to be online and operational during the entire length of the upgrade operation.

The firmware upgrade operation will comprise of 2 steps as in the BVL based FW solutions:

- Download process
- Activation process

The download process will comprise downloading the firmware image to the staging area in flash of both the controllers. Both the controllers in the array will be fully operational during this process.

The activation will be performed one controller at a time thereby, keeping the array online. The controller undergoing the activation shall failover all its activities to the Alternate controller. All the NAS Virtual Servers (vsvrs) and the corresponding LUNs owned by this controller will be failed over to the alternate controller. Also, all LUNs used for block-only access to the controller will be failed over to the alternate controller. All clustering related activities which are performed by different Virtual Machines (VMs) in this controller will also be failed over to the alternate controller.

When the newly staged firmware has been installed fully on controller, the controller will need a reboot to boot from the newly installed firmware. Once it has completed its boot and initialization process, it will attempt to failback all the LUNs from the Alternate controller thus allowing the Alternate controller to perform its upgrade.

In most cases, the controller firmware upgrade is done manually through the host software GUI or CLI. A firmware upgrade can also be triggered through the Auto Code Synchronization feature within the

controller firmware.

2.1.2.2.System Firmware Bundle

In order to simplify the upgrade process and also to eliminate run-time co-validation of the different pieces of firmware or software, the OSA product is expected to be upgraded as a single upgrade bundle. Individual pieces of firmware or software therefore will not be allowed to be upgraded. Therefore, in order to upgrade the system firmware (for adding new features, incrementing existing feature limits etc) a full image of the firmware which includes images for all VMs will be provided.

The firmware upgrade bundle will contain images for all the virtual machines running within the controller platform.

Architecture Note: Throughout the length of this document, the firmware upgrade bundle or the solution bundle will be interchangeably used. They essentially mean the same thing.

2.1.2.3.Upgrade Commit

In the OSA environment, there will be multiple virtual machines running in the VMM environment on the controller. Each virtual machine will run a specific software or firmware and each virtual machine will be storing its own configuration data. A firmware upgrade can necessitate changes to the existing configuration data. All such configurations changes will be committed to their respective repository or database during a special phase of the upgrade process called the commit phase.

This phase is not explicitly invoked by the user but implicitly invoked by the firmware which is handling the upgrade sequence. This phase is the last stage in the activation process. All the schema changes to the configuration databases are applied and committed to the database during this phase. Failure to do so will result in a rollback operation.

The firmware will support rollback to the previously running version till this commit point. Once the upgrade has been committed, rollback to the previously running version is not supported.

2.1.2.4.Rollback in case of Upgrade failure

If there is any error during or before the commit phase of the upgrade, the firmware shall rollback to the previously running version. This is again not explicitly invoked by the user but is managed by the firmware which is handling the upgrade sequence.

However, if the upgrade succeeds and the changes made as part of the upgrade has been committed to the configuration databases, rollback will not be possible at this stage.

Any attempt to activate a previous (older) version of the firmware after the new firmware has been activated and committed will result in an error.

2.1.3.Operational Behavior

2.1.3.1.General Behavior

2.1.3.1.1. Firmware Versioning

Topic ID: 2010-04-01T13:54:00Z-1085-6186-IDAGPWIE

The firmware upgrade bundle contains images for all the virtual machines that will be running within the

Pikes Peak platform. The upgrade firmware will have its own version number which will be the version number for the entire firmware bundle. There will be no version number for the individual images for the respective virtual images. All such images will have an internal version number that is meaningful only to the download management software.

The firmware versioning will be similar to the standard firmware release string. It will be in the order of *aa.bb.cc.dd*. This is the firmware revision of the system firmware bundle and not the version of the individual virtual machine images.

2.1.3.1.2. Pre-upgrade health check

Topic ID: 2010-04-01T18:43:00Z-1260-7184-IDAWLEXD

A pre-upgrade health check will need to be run prior to both the download and activation process. The pre-upgrade health check will be initiated by the array management software either through the EMW or the CLI. An invocation of the download or activation commands will implicitly invoke the pre-upgrade health check. If the pre-upgrade health check fails, the command will not proceed with the download or activation. It will throw an error indicating a health check failure.

The pre-upgrade health check will include all of the currently available health checks for the BVL based products. However, the following checks will be applied for OSA based products in order to ensure that the system is operating optimally prior to the upgrade. The activation can proceed only after the health check succeeds.

- The flash has sufficient storage capacity for the download and extraction operations
- IOVM configuration database is not corrupt
- VMM environment is stable and more than 1 unscheduled VM reboot critical event was not reported within the last 7 days

If any of the above conditions are not met, the health check fails, the utility marks the array as not upgradeable and the user is prevented from initiating a system firmware upgrade on the array.

For details on the array validation checks prior to a firmware upgrade, please refer to the Firmware Upgrade FFD, document number 349-1056450.

2.1.3.1.3. Flash partitioning

Topic ID: 2010-04-02T07:08:00Z-1723-9824-IDACYPKE

The iSATA flash device which is attached to the controller board will be partitioned for storing the solution bundle or the code images for the firmware code. The flash will be partitioned for storing the currently running or active image and also the downloaded image which will be required for a firmware upgrade.

The current running version of firmware code will be stored within the ACTIVE area. The ACTIVE area will reside on only one flash in case multiple flash drives are attached to the controller board. The ACTIVE area will not be striped across both the flash drives. The ACTIVE area will be further partitioned into individual partitions for each of the virtual machines running on the OSA platform. The code image for each virtual machine will reside within this partition. The virtual machine will therefore boot off the partition allocated for that virtual machine. This partition can also be referred to as a VBD (Virtual Block Device) for the virtual machine. This is each virtual machine's own view of the flash drive.

The flash will be further partitioned for storing the downloaded image for a firmware upgrade. This will be a single partition required to store the entire solution bundle image. This will be referred to as the STAGING area of the solution bundle within the flash drive. For a OSA based firmware solution, the STAGING area is an area within the iSATA flash drive which is used for intermediate storage of the downloaded solution bundle prior to its activation. The STAGING area can also be used to store an

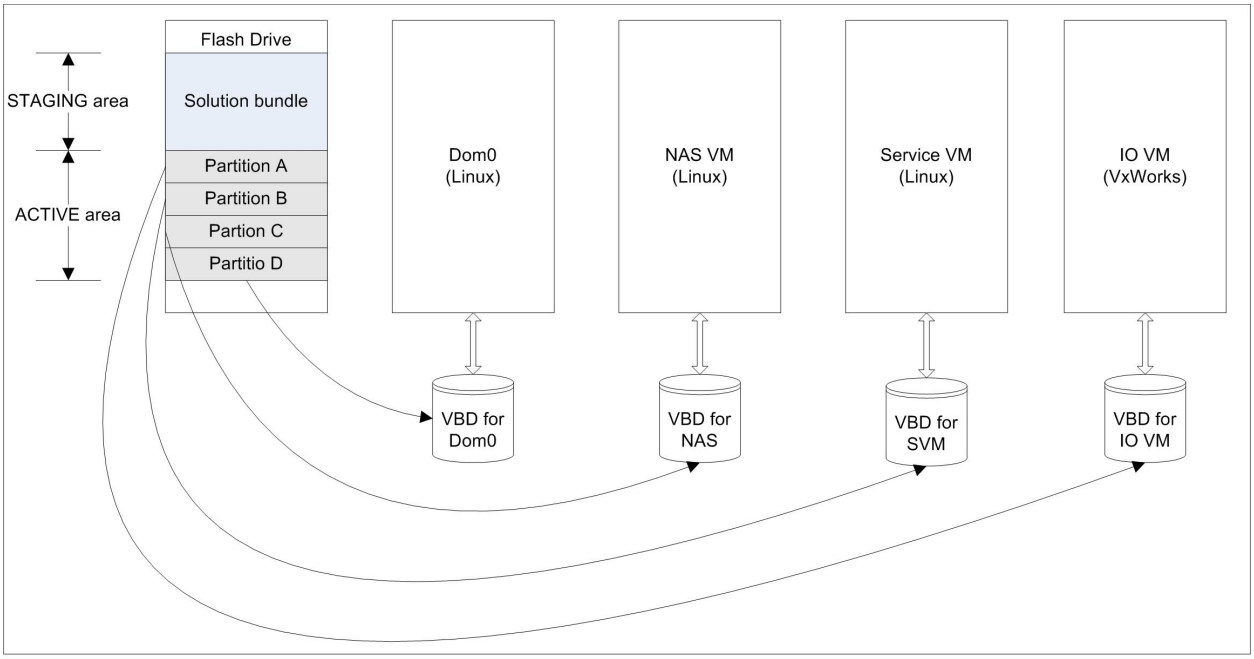
already active (running) solution bundle. Under these circumstances, this solution bundle will be used for ACS purposes.

Whenever a new Solution Bundle (for a different firmware version) is downloaded to the STAGING area, it will overwrite the solution bundle for the current firmware version. A subsequent activation will install the new firmware within the current STAGING area which will become the new ACTIVE area. The current ACTIVE area will then become the new STAGING area. However, in case of an activation failure, the newly installed firmware will be discarded and the controller will reboot from the current ACTIVE area again.

The following diagram depicts the view of the ACTIVE and STAGING area on the flash drive. This also provides a pictorial view of the virtual block device associated to each virtual machine.

Architecture Note: This figure is just a pictorial representation of how the firmware code image is stored in the flash drive. The flash drive will be further partitioned for storing other information which is related to the operational behavior of the controller. Those partitions have not been depicted in this figure. The figure is also not drawn to scale.

Figure 1: Flash Partitioning



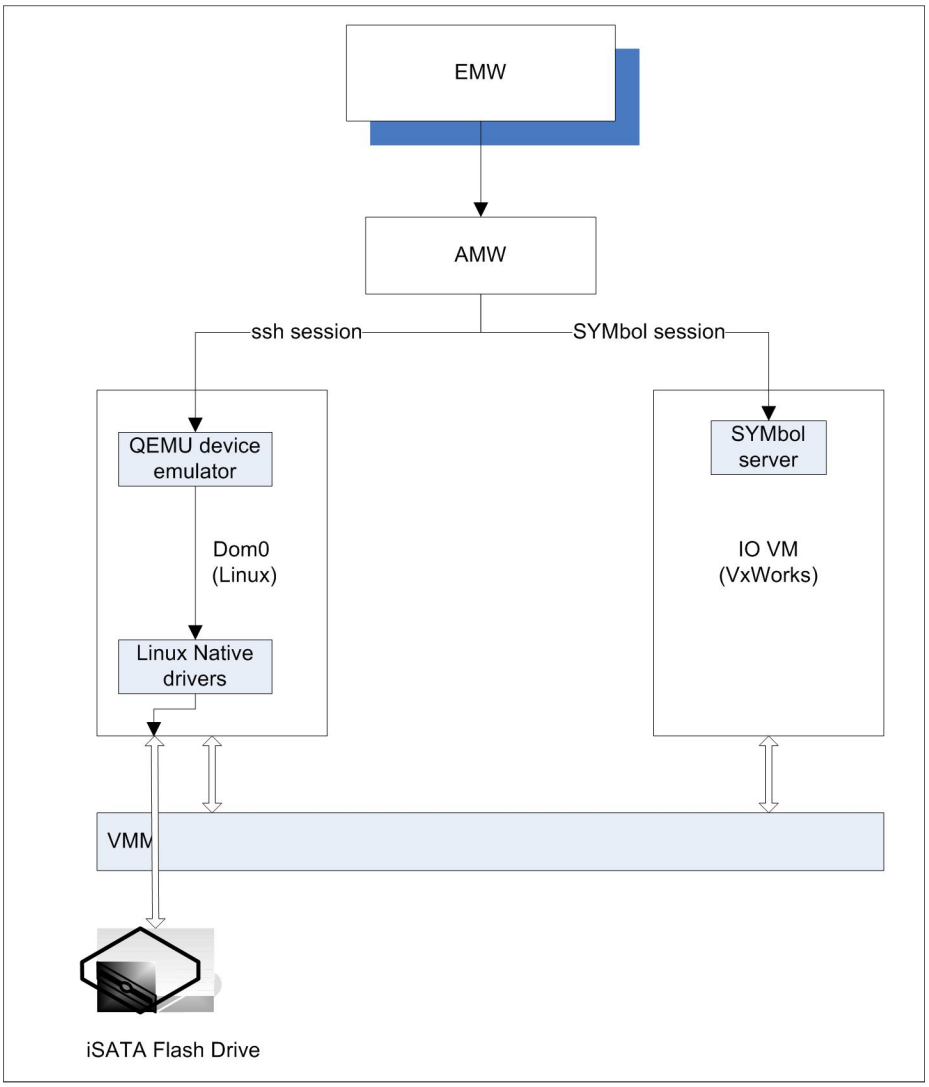
Architecture Note: For the Stage 1 release of the product, there will be 2 virtual machines running on the controller platform (Domain0 and IOVM). The partitioning will therefore ensure that virtual block devices are created for the currently running virtual machines only.

2.1.3.1.4. Upgrade workflow

Topic ID: 2010-04-01T18:50:00Z-1327-7569-IDAEOMXD

A single instance of the OSA firmware bundle will need to be downloaded on to the storage array controller and activated. This bundle will be downloaded as a whole using management commands from the GUI or the CLI. There will be two commands, one for the download and one for the activation phase.

Figure 2: Upgrade workflow coordination



The firmware download and activation commands will be serviced by the IOVM virtual machine running on the controller platform. Once a firmware upgrade bundle has been downloaded, it will be stored within the iSATA flash on the controller board.

Architecture Note: The firmware download and activation commands will be handled by IOVM only for the Stage 1 release. For later releases, the commands will be handled by the Service VM.

The solution bundle will be directly downloaded to the flash owned by Domain0. Even though the management commands will be serviced by IOVM, the management client will open a special ssh session to Domain0 to initiate the download. The ssh server on Domain0 will be configured to listen to a special port configured for Domain0 ssh sessions. The steps for a download operation will be:

1. The management client will call a SYMbol API to get the download authentication and configuration information
2. The management client will open a special ssh session to Domain0

3. The management client will download the solution bundle over the ssh session. The bundle will be written to a specified location within the flash
4. Once the download is complete, the management client will call a SYMbol API to inform the controller firmware that download is complete
5. The controller firmware starts copying the solution bundle to the Alternate controller's flash
6. The management client will call a SYMbol API to check status if the solution bundle has been copied to the Alternate controller's flash
7. The management client completes the download operation

The downloaded bundle will be written to the STAGING area within the iSATA flash drive. Once a solution bundle is downloaded, it will overwrite any image that is currently stored in the STAGING area. Only if the existing solution bundle in the STAGING area is having the same firmware version as of the image to be downloaded, the management software will prevent download of the new solution bundle.

2.1.3.1.5. Firmware download

Topic ID: 2010-04-02T09:44:00Z-2392-13641-IDA1BHWE

The firmware download process for OSA based firmware will be a two step process. The first step will be to retrieve the firmware solution bundle which is currently running on the controller and save that to a known location within the management host. The second step will be to download the new version of the firmware solution bundle to the flash resident on the controller. The two steps will be implicitly performed as part of the execution of the download management command.

The firmware download mechanism for OSA based firmware will be continuous stream based. It will not be "chunk" or segment based as is adopted for the BVL based RAID controllers. The entire solution bundle will be transferred as one continuous stream of data to the controller. As mentioned in section Section 2.1.3.1.4. Upgrade workflow, it will be initiated through a SYMbol command serviced by the IOVM. Invocation of the download command will implicitly invoke the pre-upgrade health check utility for the management software running on the host.

In dual controller systems, a primary controller will be selected for the download of the firmware bundle. The primary controller is the controller which services the management command, i.e. the SYMbol command. The download management software shall first copy the solution bundle for the currently running version of the firmware which is resident in the STAGING area in the flash to a known location within the management host. It will then download the solution bundle fully to the STAGING area of the flash for the primary controller. Once the download to the primary controller is complete, the firmware running on the controller (which manages the download operation) will transfer the firmware bundle to the STAGING area of the flash for the Alternate controller. The controller firmware which manages the download operation will use the Ethernet link though the mid-plane for transferring the firmware image to the Alternate controller. The download processing is complete when the firmware image or the solution bundle is successfully copied to the STAGING area of the flash for both the controllers.

Firmware validation occurs during the transfer of the solution bundle. CRC verification will occur for the solution bundle prior to the download. This CRC will be generated as part of the solution bundling process. The management software will need to validate the CRC prior to the download. If the CRC verification fails, the management software will not attempt to download the solution bundle. The download command will return with an error indicating CRC verification failed the download.

Firmware validation will also occur after the solution bundle has been copied to the STAGING area of the flash. The firmware code that handles download management will perform a validation of the transferred image. The validation will include CRC verification. If the validation is successful, the solution bundle will be marked "Ready", else the solution bundle will be marked "Invalid". If the solution bundle for any one controller is marked as "Invalid", the firmware will attempt to transfer the "Ready" image from the Alternate controller. If the solution bundle for both the controllers is marked as "Invalid", an error will be

returned to the host.

Architecture Note: An "Invalid" firmware solution bundle will fail activation.

2.1.3.1.6. Firmware activation - Dual controller system

Topic ID: 2010-04-02T09:44:00Z-2392-13641-IDA0GHWE

The activation process for the STAGED firmware bundle is initiated by a management command through the GUI or CLI. The process of activation for the solution bundle will move the firmware image from a "Ready" state to an "Active" state. A firmware in "Active" state is the current running version of the firmware on the controller. Similar to the download, the activation command is also serviced by the IOVM virtual machine. Invoking the command to activate the STAGED firmware bundle will also implicitly invoke the pre-upgrade health check.

As in the case of the download command, for dual controller systems, a primary controller is selected for activation.

During the activation process, all user initiated configuration changes using SYMbol commands through the AMW will not be allowed on the controller. In case of dual controller systems, user initiated configuration changes through SYMbol will not be allowed for either controller. Only the commands which are necessary for activation or any other recovery related command will be allowed. This will hold good till the activation process reaches a point where the firmware upgrade is successful and the changes committed or if the upgrade failed and both the controllers have performed a rollback to the previously running version of the firmware.

The following operational steps are performed as part of activation for a dual controller system running a OSA firmware (assume primary controller is Controller A):

1. Mark necessary data structures in SYMbol to prevent any user initiated configuration changes.
2. Write caching is disabled for all volumes within the array. A cache sync operation is performed.
3. The virtual volumes in Controller A are transferred to Controller B. This also invokes the transfer of virtual drives owned by Controller A to Controller B.
4. The firmware solution bundle is moved to a temporary scratch area in the flash. The bundle is then CRC verified. The individual virtual machines images are also extracted and CRC verified.
5. Controller A creates the necessary (VBD) partitions for all the virtual machines running on the controller. It installs the Hypervisor+Domain0 image on to the partition created for Domain0.
6. The STAGING area becomes the new ACTIVE area while the ACTIVE area becomes the new STAGING area.
7. The boot loader is modified to boot from the new Domain0 partition. Controller A reboots itself.
8. After reboot, only the Domain0 virtual machine is initialized on Controller A. The other virtual machines are not started at this point.
9. The firmware managing the activation operation will now install the images for the other virtual machines on to the specified (VBD) partition for the virtual machine.
10. All other virtual machines are spawned at this time from the new partition containing the new version of the virtual machine image.
11. Each virtual machine will start initializing. Once each virtual machine has completed its initialization sequence, Controller A will now transfer all virtual volumes from Controller B (this also transfers the virtual drives).
12. Controller A indicates Controller B to perform the same steps from 3 - 8.
13. Once all the virtual machines have initialized with the new firmware image on Controller B and the upgrade changes are committed, the virtual volumes are transferred to their preferred owner controllers.
14. The SYMbol command restriction is now removed.

2.1.3.1.7. Disable write caching of volumes during activation

Topic ID: 2010-04-15T17:15:00Z-6168-35158-IDALCOUD

During activation a scratch area is required to extract the individual virtual machine packages from the solution bundle and perform CRC verifications and run necessary pre-install scripts. The cache offload area within the iSATA flash will be used as a scratch area during the activation. Hence, as part of the activation process, the solution bundle will be moved or copied to the cache offload area.

Since the cache offload area will be used for extraction, the controller firmware will not be able to perform a cache offload in the event of a AC power loss during the activation process. Therefore, write caching will be disabled for all volumes prior to moving the data to the cache offload area during the activation phase. Write caching will remain disabled for all volumes till the activation reaches the commit phase of the upgrade process.

2.1.3.1.8. Restore firmware solution bundle

Topic ID: 2010-04-07T09:01:00Z-4977-28371-IDAU0VUC

In a OSA based environment, the download operation will consist of retrieving a copy of the solution bundle corresponding to the current running version of firmware and then downloading the new solution bundle. The new solution bundle will overwrite the solution bundle of the current running version within the STAGING area.

In case the controller fails activation with the new version of the firmware, the controller firmware will perform a rollback to the previously running version that is currently installed in the current ACTIVE area. However, post the rollback the (un-extracted) solution bundle for the previously running firmware will not be available. This will be needed for ACS purposes in the event a spare controller with a different version of the firmware is inserted to the array. Performing ACS from the installed area is cumbersome and can also lead to corruption because data on disk may not be in sync with the file system cache at the time of ACS.

Therefore, in the event of an activation failure, the solution bundle for the previously running firmware version will need to be restored to the controller flash from the management host (saved during the download operation). This will be done using a management command invoked through the GUI or CLI.

Once the activation fails, the controller will boot off the previously running version of firmware. It will log a critical event and a Needs Attention for the array. The recovery action will be to invoke the command to restore the previously running firmware solution bundle.

2.1.3.1.9. Download package header format

Topic ID: 2010-04-17T15:37:00Z-6921-39453-IDAXE51D

The solution bundle will contain a payload and a header. The header will contain metadata about the payload and also firmware version and compatibility information. The header format is defined in the Feature Architecture Specification for firmware download, document number 349-1005040.

The following information will be part of the header (in addition to the information specified within the above document):

1. The CRC information for the solution bundle (this will be the CRC of the payload).
2. The solution bundle size (to perform space availability checks).
3. OSA compatibility (a non-OSA firmware will not be downloaded to a OSA platform and vice versa).

2.1.3.1.10. Configuration database update sequence

Topic ID: 2010-04-02T11:37:00Z-3162-18026-IDAXNDYE

If there is a change in the NAS cluster database schema, the NAS cluster database upgrades will be performed after all the NAS nodes in the clusters have been upgraded. The NAS software will check after a designated period of time to see if a cluster DB migration is necessary. This check is done only on the node which is the Primary Cluster Controller (PCC) node. It will check the current version of the cluster DB on disk with that which is built into the code. (Note that this is only necessary for upgrades that require an upgrade to the cluster DB schema).

Once it determines that a cluster DB schema migration is necessary it will send a version request to each node in the cluster, polling to get their system versions. The timeout period is set to ensure that all the services on the node finish their initialization phase and finish their specific updates to the NAS cluster DB. When each node in the cluster is running the new code the PCC will notify each node to lock the cluster DB. Finally a backup copy is made and the cluster DB schema is upgraded by walking across the cluster DB records and changing each record as needed.

Therefore, as mentioned above, there is a wait time before the NAS cluster DB can be upgraded in case of a schema upgrade. In the interim the IOVM stable storage and the SVM setup volume can be upgraded if required but rollback in case of a NAS cluster DB upgrade failure is non-trivial. Hence to circumvent this scenario (at least for the first out Orion product), the following sequence will be adopted for committing the upgrade for each database (note that committing the change essentially involves a configuration database upgrade if required as well):

1. Once both the controllers come up with the new version of firmware, none of the databases will be upgraded till the timer expires for the NAS cluster DB. As part of the timeout handling, it will commit the changes to the NAS cluster DB and upgrade the DB schema if required.
2. If the NAS cluster DB upgrade succeeds, commit changes if required to the SVM setup volume.
3. Then commit any necessary changes to the IOVM stable storage.

Architecture Note: For the Stage 1 release of Orion, the NAS components will not be running on the controller. Also, a DACStore revision upgrade is not envisioned for the Stage 1 release of Orion. Hence this sequence is applicable for later stages of Orion which will be integrated with the NAS software.

2.1.4.Administrative and Configuration Interfaces

2.1.4.1.SYMbol interfaces

2.1.4.1.1. Get download configuration

Topic ID: 2010-04-17T14:59:00Z-6748-38464-IDACM3EE

A SYMbol interface *getDownloadConfiguration* is defined in the SYMbol API. This interface will provide the functionality to get the configuration information to set up the ssh session with Domain0. The following information will be exchanged between the management client and the SYMbol server:

- ssh authentication keys
- ssh port number
- absolute download path within Domain0 file system

This command will require SYMbol password authentication.

The details of this SYMbol API will be covered under the Detailed Architecture section of this element.

2.1.4.1.2. Start OSA firmware download

Topic ID: 2010-04-02T12:36:00Z-3644-20772-IDAKQDWE

A SYMbol interface *startOSAFirmwareDownload* is defined in the SYMbol API. This interface will be used to indicate to the controller firmware that the management client will initiate the download of the OSA firmware solution bundle through a parallel session. This command will require SYMbol password authentication.

The details of this SYMbol API will be covered under the Detailed Architecture section of this element.

2.1.4.1.3. OSA firmware download complete

Topic ID: 2010-04-17T14:59:00Z-6748-38464-IDAZT3EE

A SYMbol interface *OSAFirmwareDownloadComplete* is defined in the SYMbol API. This interface will be used by the management client to indicate to the controller firmware that the solution bundle download to Domain0 is complete. This command will require a SYMbol password authentication.

The details of this SYMbol API will be covered under the Detailed Architecture section of this element.

2.1.4.1.4. Get solution bundle peering status

Topic ID: 2010-04-17T14:59:00Z-6748-38464-IDAFW3EE

A SYMbol interface *getSolutionBundlePeeringStatus* is defined in the SYMbol API. This interface will be used by the management client to fetch the status of peering of the solution bundle. The management client returns the download command only when the solution bundle has been peered to the alternate controller. This command will require a SYMbol password authentication.

The details of this SYMbol API will be covered within the Detailed Architecture section of this element.

2.1.4.1.5. Activate OSA firmware

Topic ID: 2010-04-02T12:36:00Z-3644-20772-IDA2SDWE

A SYMbol interface *activateOSAFirmware* is defined in the SYMbol API. This interface will provide the functionality to activate the STAGED firmware. This command will require SYMbol password authentication.

The details of this SYMbol API will be covered under the Detailed Architecture section of this element.

2.1.4.1.6. Save Firmware Solution Bundle to Host

Topic ID: 2010-04-14T17:36:00Z-5982-34098-IDAP3MND

A SYMbol interface *saveFirmwareBundleToHost* is defined in the SYMbol API. This interface will provide the functionality to save off the firmware solution bundle from the STAGING area to a known location within the management client (host). This command will require SYMbol password authentication.

The details of this SYMbol API will be covered under the Detailed Architecture section of this document.

2.1.4.1.7. Save Firmware Solution Bundle to Host complete

Topic ID: 2010-04-20T12:44:00Z-7123-40606-IDA4N1OE

A SYMbol interface *saveFirmwareBundleToHostComplete* is defined in the SYMbol API. This interface will provide the functionality to indicate to the controller firmware that the current solution bundle has been saved to a known location within the management client (host). This command will require SYMbol password authentication.

The details of this SYMbol API will be covered under the Detailed Architecture section of this document.

2.1.4.1.8. Restore previously running OSA firmware

Topic ID: 2010-04-07T07:02:00Z-4684-26705-IDAUCOTC

A SYMbol interface *restorePreviousOSAFirmware* is defined in the SYMbol API. This interface will provide the functionality to restore the solution bundle (which was copied to the management host as part of the firmware download operation) to the flash resident within the controller. The restore operation will be a synchronous operation. A successful exit can be construed as a successful restore to the controller. This command will require SYMbol password authentication.

The details of this SYMbol API will be covered under the Detailed Architecture section of this element.

2.1.4.1.9. SYMbol restrictions

Topic ID: 2010-04-16T10:33:00Z-6340-36140-IDA5MFVD

All configuration changes to the array using SYMbol commands will not be allowed during activation. All active SYMbol commands which require a password authentication will not be allowed during activation. Invocation of these commands will return an error code *RETCODE_ACTIVATION_IN_PROGRESS_TRY_LATER*.

If a download or an activation command is tried in parallel to another download operation, the command which was initiated at a later point in time will return *RETCODE_DOWNLOAD_IN_PROGRESS*.

However, all commands related to download, activation and subsequent recovery will be allowed (even if they require password authentication).

If a download or activation is tried while activation is already in progress, it will return an error code *RETCODE_ACTIVATION_IN_PROGRESS*.

2.1.4.2.GUI interfaces

2.1.4.2.1. Download OSA firmware

Topic ID: 2010-04-02T12:36:00Z-3644-20772-IDA3VDWE

An option is displayed under the system GUI to execute the command to download the integrated OSA firmware bundle. When this option is selected, it will perform the FW download operation to the controller. The download operation will be a synchronous operation. A successful exit can be construed as a successful FW download to the controller. The GUI performs the download by calling the *downloadOSAFirmware* SYMbol call.

The details of this GUI option will be covered under the Detailed Architecture section of this element.

2.1.4.2.2. Activate OSA firmware

Topic ID: 2010-04-02T12:36:00Z-3644-20772-IDA4XDWE

An option is displayed under the system GUI to execute the command to activate the OSA FW. When this option is selected, it will perform the FW activation on the controller. The GUI performs the activation by calling the *activateOSAFirmware* SYMbol call.

The details of this GUI option will be covered under the Detailed Architecture section of this element.

2.1.4.2.3. Restore previously running OSA firmware

Topic ID: 2010-04-07T10:33:00Z-5363-30572-IDA4OIVC

An option is displayed under the system GUI to execute the command to restore the previously running version of OSA firmware in case activation fails and the controller firmware performs a rollback to the previously running version of firmware. The GUI performs the restore operation by calling the *restorePreviousOSAFirmware* SYMbol command.

The details of this GUI option will be covered under the Detailed Architecture section of this element.

2.1.4.3.CLI interfaces

2.1.4.3.1. Download OSA firmware

Topic ID: 2010-04-02T12:36:00Z-3644-20772-IDAA1DWE

A command is provided within the system CLI to download the integrated OSA firmware bundle. When this option is selected, it will perform the FW download operation to the controller. The download operation will be a synchronous operation. A successful exit can be construed as a successful FW download to the controller. The CLI performs the download by calling the *downloadOSAFirmware* SYMbol call.

The details of this command will be covered under the Detailed Architecture section of this element.

2.1.4.3.2. Activate OSA firmware

Topic ID: 2010-04-02T12:36:00Z-3644-20772-IDA32DWE

A command is provided within the system CLI to activate the OSA FW. When this option is selected, it will perform the FW activation on the controller. The CLI performs the activation by calling the *activateOSAFirmware* SYMbol call.

The details of this command will be covered under the Detailed Architecture section of this element.

2.1.4.3.3. Restore previously running OSA firmware

Topic ID: 2010-04-07T10:33:00Z-5363-30572-IDA2VIVC

A command is provided within the system CLI to restore the previous running version of the OSA firmware solution bundle. This command will be run when the activation fails for the new firmware and the controller firmware performs a rollback to the previously running version of the firmware. The CLI will perform the restore by calling the *restorePreviousOSAFirmware* SYMbol command.

The details of this command will be covered under the Detailed Architecture section of this element.

2.1.4.4. Object Graph

Topic ID: 2010-04-02T13:09:00Z-3802-21674-IDA5ZZKE

The object graph returned by the *getObjectGraph* function will contain all the necessary pieces of information required by the management client to manage the OSA firmware download and activation feature. This information consists of:

- A list of firmware images resident on the flash and their attributes
  - Version
  - Image state
  - Date/Time when transferred
- Information in the *SAData* structure which indicates time of reboot and firmware version required to determine when the activation was completed

2.1.5.Error Handling and Event Notification

2.1.5.1.Error Handling and Recovery

The error handling scenarios during download and activation have been identified in the Firmware Upgrade FFD (document number 349-1056450). Please refer to the FFD for more details on error handling.

Listed in the following subsection(s) are the error scenarios which will be specific to the OSA firmware. This is in addition to the error scenarios specified in the Firmware Upgrade FFD.

2.1.5.1.1.Failure during download

The following error conditions can be encountered during the download phase of the firmware upgrade.

2.1.5.1.1.1. Unable to copy firmware solution bundle to the management host

Topic ID: 2010-04-07T08:45:00Z-4899-27926-IDAYSYQC

This error can occur if the solution bundle for the currently running version of the firmware within the controller cannot be copied to the known location within the management host. This is possible if sufficient storage space is not available within the designated directory or disk partition owned by the management host. The download command will fail in such a scenario. The recovery action will be to ensure enough storage space is available for copying the solution bundle (for the currently running version of the firmware) to the management host and then re-initiate the download command.

The copy operation can also fail if the session established by the management client to the controller for the download expires or is broken. The recovery action again will be to re-initiate the download command.

2.1.5.1.2.Failure during Activation

The following error conditions can be encountered during the activation phase of the firmware upgrade.

2.1.5.1.2.1. Domain0 unable to boot with new firmware

Topic ID: 2010-04-02T13:38:00Z-4062-23160-IDA5A0XE

This can occur if the controller firmware fails to perform step 8 as identified in section Section 2.1.3.1.6. Firmware activation - Dual controller system. This can happen due to a possible software bug within the VMM environment. In such a scenario the controller becomes unresponsive. The 7-segment display code set by the BIOS as part of the boot sequence will be used to determine that the controller is unable to

boot up. The system recovery method that will be adopted in such a scenario is to press the push button. This will initiate a PXE (Pre-eXecution Environment) boot. The PXE boot mechanism will boot the controller from an external TFTP server within the network.

2.1.5.1.2.2. Guest Virtual Machines unable to boot with new firmware

Topic ID: 2010-04-02T13:38:00Z-4062-23160-IDAYB0XE

This error condition can be encountered if the controller firmware fails to perform step 11 as specified in section Section 2.1.3.1.6. Firmware activation - Dual controller system. If the failing controller is the designated primary controller for the activation then it will attempt to rollback to the previous running version of the firmware for the primary controller only (note that the secondary controller has still not initiated its activation process).

Primary Controller:

The steps followed for rollback by the primary controller:

1. The firmware handling the activation in Domain0 will wait for a specified timeout period allowing the guest virtual machines to complete their initialization process.
2. If any of the guest virtual machines do not complete their initialization by this time, the virtual machine manager running on Domain0 will destroy or bring down all of the guest virtual machines.
3. The boot loader will be modified to boot from the previously running version of the firmware.
4. The primary controller will reboot itself.
5. Once the primary controller comes up with the previously running version of the firmware, the virtual volumes will be re-balanced.
6. The SYMbol command restriction will be removed.

Secondary Controller:

If the failing controller is the secondary controller, then it will first attempt to rollback to the previous running version for the secondary controller first and then attempt to rollback to the previously running firmware version for the primary controller.

The steps performed by the secondary controller:

1. Perform steps 1-3 as specified above for the primary controller.
2. The secondary controller will reboot itself.
3. Once the secondary controller comes up with the previously running version of the firmware, all virtual volumes will be transferred to the secondary controller.
4. The primary controller will perform steps 1-5 as specified for the primary controller sequence above.
5. The SYMbol command restriction will be removed.

2.1.5.2.Event Notification

Event notifications for the firmware download and activation process has been defined in the Firmware Upgrade FFD (document number 349-1056450). Please refer to the FFD for more details on event notifications.

Listed below are the events that are introduced as part of the OSA based firmware or events that will be deprecated within the OSA firmware

2.1.5.2.1. Controller Firmware Rollback Start

Topic ID: 2010-04-20T12:44:00Z-7123-40606-IDAIUERE

An informational event will be logged in the event the controller fails activation and starts rollback to the previous running version of the firmware.

EVENT SYNOPSIS: An event is logged when the controller fails to activate the newly downloaded package. This event triggers the start of the rollback process.
MEL EVENT TYPE: Informational
MEL AFFECTED COMPONENT: Storage Array
FAILURE TYPE NAME: N/A
RECOVERY PROCEDURE: None
ADDITIONAL EVENT DETAILS: None

2.1.5.2.2. Controller Firmware Rollback Complete

Topic ID: 2010-04-07T07:02:00Z-4684-26705-IDAF2OTC

A critical event will be logged in the event that the controller fails activation with the new version of the firmware. A Needs Attention condition will be displayed and a Recovery Action will also be displayed in the GUI.

EVENT SYNOPSIS: An event is logged when the controller fails activation with a new version of firmware and successfully performs a rollback to the previous running version of the firmware.
MEL EVENT TYPE: Critical
MEL AFFECTED COMPONENT: Storage Array
FAILURE TYPE NAME: N/A
RECOVERY PROCEDURE: Restore previously running firmware solution bundle
ADDITIONAL EVENT DETAILS: None

2.1.5.2.3. Controller Firmware not ACS capable

Topic ID: 2010-04-14T17:36:00Z-5982-34098-IDATDOND

A critical event will be logged in the event that the solution bundle image in the STAGED area is discarded.

EVENT SYNOPSIS: An event is logged when the downloaded solution bundle in the STAGED area is discarded. This occurs when a spare controller with the same version of firmware is inserted between a download and activation.
MEL EVENT TYPE: Critical
MEL AFFECTED COMPONENT: Storage Array
FAILURE TYPE NAME: N/A
RECOVERY PROCEDURE: Restore firmware solution bundle
ADDITIONAL EVENT DETAILS: None

2.1.5.2.4. Solution bundle restore complete

Topic ID: 2010-04-17T15:37:00Z-6921-39453-IDAUIB2D

An informational event will be logged when the solution bundle for the previously running firmware has been restored to the controller.

EVENT SYNOPSIS: An event is logged when the solution bundle for the previously running firm-

ware version is restored to the controller.
MEL EVENT TYPE: Informational
MEL AFFECTED COMPONENT: Storage Array
FAILURE TYPE NAME: N/A
RECOVERY PROCEDURE: None
ADDITIONAL EVENT DETAILS: None

2.1.5.2.5. Deprecated Events

Topic ID: 2010-04-06T12:22:00Z-4520-25764-IDAW20PD

Listed below are the events which will not be supported for OSA based firmware:

- Controller Firmware Download Checkpoint
- SYMbol SOFD Invalidate Start
- SYMbol SOFD Invalidate Complete
- SYMbol SOFD Invalidate Failure

2.1.6.Compatibility and Migration

2.1.6.1. Migration to File Based release

Topic ID: 2010-03-26T08:52:00Z-1265-7213-IDAPLZOD

The initial release for OSA based products will be a block-only release. However, the current external behavior will need to be forward compatible to future releases of the product which includes a file based release.

Architecture Note: For future releases of the OSA based product, the management commands to perform the download; activation and restore will be serviced by a different guest VM (Service VM) which will service all management commands henceforth. The existing commands serviced by IOVM will therefore need to seamlessly move over to the Service VM for later releases.

2.1.6.2. Forward compatibility of management commands

Topic ID: 2010-03-26T08:52:00Z-1265-7213-IDAMMZOD

All management commands to download the solution bundle; activate the solution bundle and restore the solution bundle will need to be forward compatible to all future releases of a OSA based product. The syntax and semantics of the CLI command will need to be forward compatible for all future releases.

2.1.6.3. Upgrade from non-OSA firmware

Topic ID: 2010-04-16T08:13:00Z-6216-35435-IDAJBMBE

A firmware upgrade from non-OSA based firmware to a OSA based firmware will not be supported.

2.1.7.Restrictions and Limits

2.1.7.1.Restrictions

The restrictions pertaining to firmware upgrade have been identified in the Firmware Upgrade FFD (document number 349-1056450). Please refer to the FFD for more details on restrictions during download and activation.

Listed in the following subsection(s) are the restrictions which are imposed by the OSA firmware. This is in addition to the restrictions specified in the Firmware Upgrade FFD.

2.1.7.1.1. Prohibit user initiated configuration changes

Topic ID: 2010-04-02T13:28:00Z-3896-22213-IDAYFEWE

All user initiated configuration changes using SYMbol commands through the GUI or CLI will be prohibited during the activation phase. This is enforced by the controller firmware. Any attempt to change any configuration will result in an error indicating to the user to retry the operation at a later point in time. This enforcement will hold good till the activation process reaches a point where the firmware upgrade has been successful and the changes committed or if the upgrade failed and both the controllers have performed a rollback to the previously running version of the firmware.

2.1.7.1.2. Auto Code Sync support

Topic ID: 2010-04-14T10:44:00Z-5814-33141-IDAVCSAC

In case a controller is removed and a spare controller running a different version of firmware is inserted during the time window between a download operation and the subsequent activation, auto code sync cannot be performed by the incumbent controller. The incumbent controller will identify that the STAGED area contains staged (downloaded) image that has not been activated. Hence it will fail the ACS request.

The FOREIGN controller will perform a self lockdown in such a scenario.

The situation can be circumvented if a controller with the same running version of the firmware is inserted as the FOREIGN controller. In such a scenario, the FOREIGN controller will boot with the firmware. The staged solution bundle in the STAGING area of the flash will be discarded. A critical event will be logged to indicate that the auto code sync capability is lost.

2.1.7.2.Limitations

2.1.7.2.1. BIOS upgrades

Topic ID: 2010-04-06T12:59:00Z-4621-26344-IDACG1SC

The firmware upgrade will not handle updates to the BIOS. The BIOS upgrade will be performed by trained service personnel. BIOS upgrades will not be allowed using standard management commands invoked from a management host.

2.1.7.2.2. Drive and ESM FW download

Topic ID: 2010-04-06T12:59:00Z-4621-26344-IDAHH1SC

Controller firmware download will not be allowed simultaneously with drive FW or ESM FW downloads. If any of these operations are in progress within the array, the download operation will return an error.

2.2. Detailed Controller Firmware Architecture

2.2.1.High Level Design

2.2.1.1.Overview

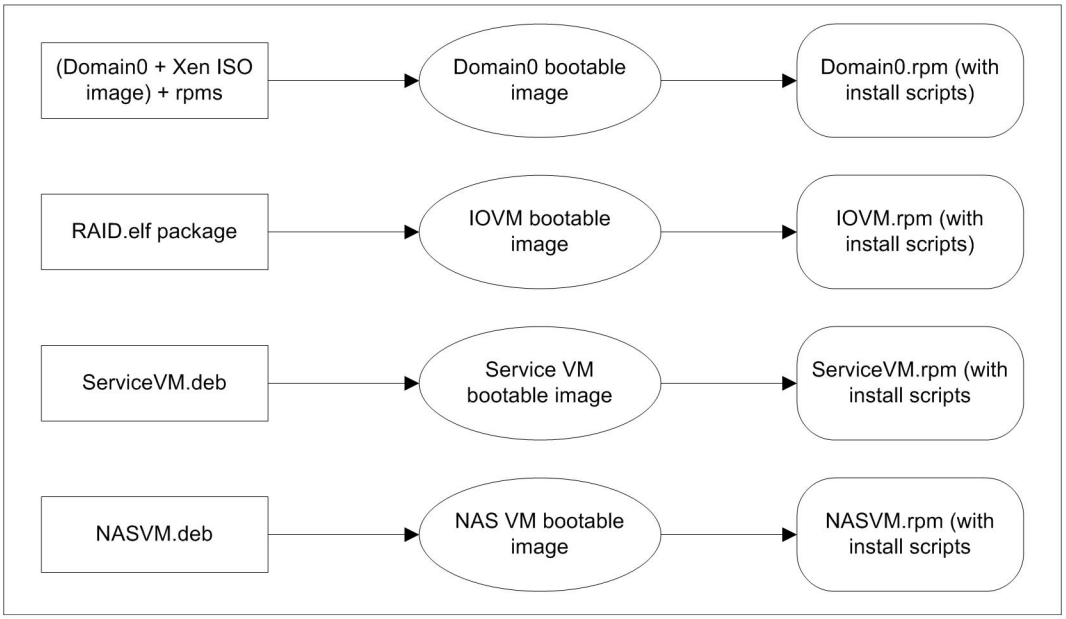
2.2.1.1.1.Solution bundle

For OSA based firmware, a single solution bundle will contain the images for all the virtual machines which will be running on the controller. For the initial release of the OSA based firmware (Orion Stage 1), the unified firmware image will comprise of the Domain0 binary (which includes the Xen kernel, the Domain0 kernel and the Domain0 application and LSI packages) and the IOVM binary (which include the VxWorks image and the RAID application). For later release of Orion which includes a file based application, the unified firmware solution bundle will also include the NAS VM binary (which includes the Linux kernel and the NAS Cougar application) and the Service VM binary (which includes Linux kernel and the applications on the Service VM).

The solution bundle generation will be a 2-step process. The first step will comprise taking the package for each VM and create a bootable image from the package. The bootable image will then be wrapped around in rpm (Redhat Package Manager) format with its pre-install, install and post-install scripts.

The following figure illustrates the workflow for step 1 of the solution bundling process:

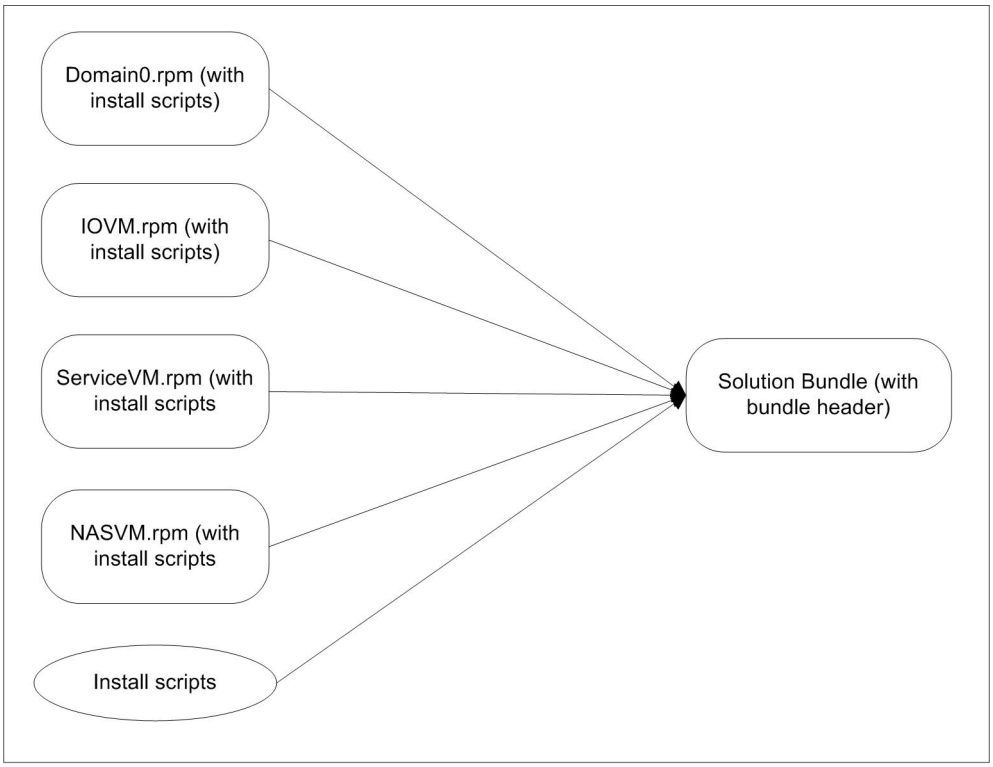
Figure 3: Solution Bundle Process Step 1



It is to be noted that the package for each VM will be built and maintained at its own cadence. The solution bundling process will attempt to integrate the individual packages for each VM into a single unified solution bundle. Please refer to the OSA Development and Build Environment document (44682-00) for more details on the same.

The second step of the solution bundling process will comprise integrating the individual rpms created in step-1 to a single unified solution bundle with a bundle header.

Figure 4: Solution Bundle Process Step 2



The rpms for each of the virtual machines (as created in Step 1) will now be integrated as a single solution bundle. The solution bundle will be a compressed image and include a bundle header. The bundle header will include information regarding the firmware version of the bundle, CRC information, OSA compatibility etc. This information will be read by the management software prior to and as part of the firmware download operation. The bundle header will also be read by the firmware as part of the solution bundle extraction.

The entire solution bundle will be downloaded for an install or a upgrade operation. Individual pieces of firmware or software will not be allowed to be installed or upgraded.

2.2.1.1.2.First time install

This can also be referred to as a Manufacturing install. The flash drives which will host the system firmware image will be shipped from SMART with a bare bone Linux OS. This will be a Linux distribution with a very small footprint just enough to boot the controller in a baremetal mode. The bare bone Linux image will also allow basic network configurations to be set and also contain a basic set of utilities which will allow the controller to download a solution bundle over the internal network, install it and boot from the download firmware.

Following will be the steps performed during a first time install process:

1. SMART will ship iSATA flash drives with a bare bone Linux (also referred to as the pre-programmed boot image)
2. The controller will boot from this bare bone Linux image. It will get an IP address via DHCP
3. The Quanta diagnostics will then be loaded to the flash drive. The controller boots up the diagnostics

and performs manufacturing diagnostics

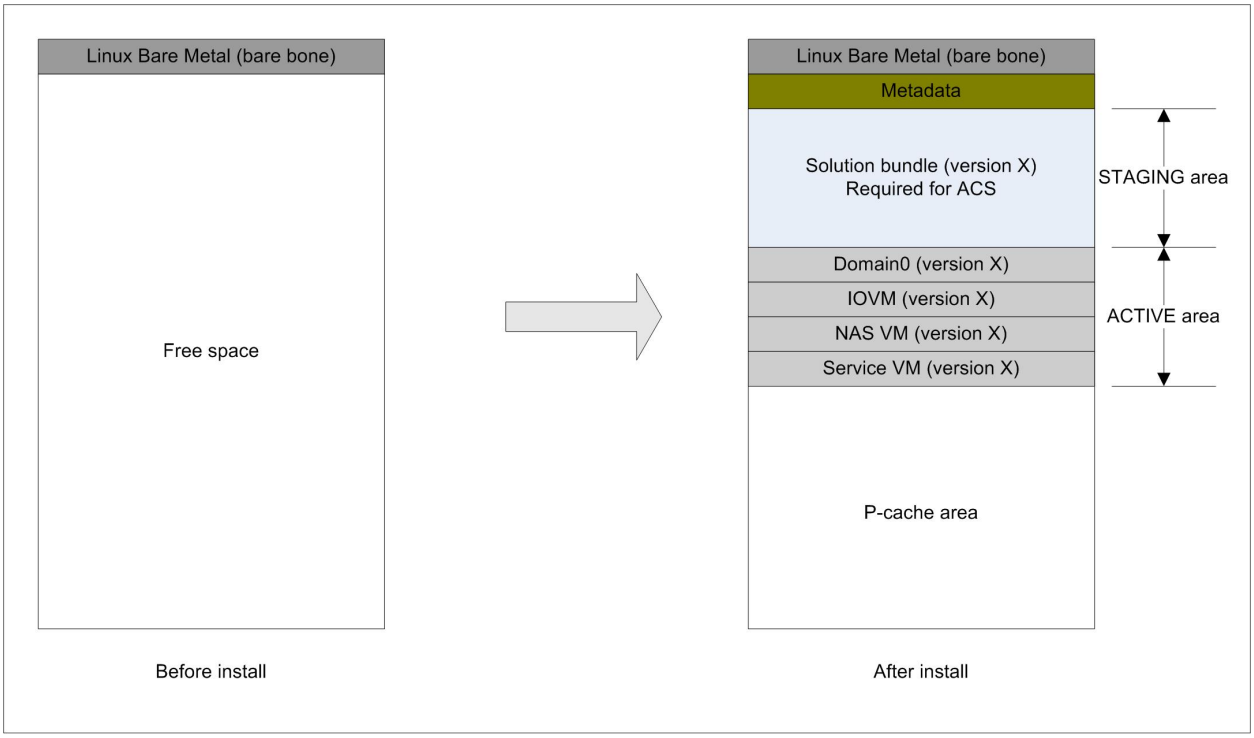
4. When the manufacturing diagnostics steps are complete, a command will be issued to download the firmware solution bundle over the network
5. The newly downloaded firmware solution bundle will be installed and the controller will boot from this new firmware

2.2.1.1.2.1.Install the firmware solution bundle

The steps 1 - 3 as identified in Section 2.2.1.1.2. First time install will be covered in more details in the Manufacturing AAD. This AAD element will specify steps 4 and 5 in more detail. These steps will identify the install process in more detail which includes the flash partitioning for the first time install and the subsequent reboot from the new firmware. The following diagram depicts the flash partitioning before and after the first time install process.

Architecture Note: Please note that the diagram is not to scale and the partitions of individual virtual machines within the STAGING area can vary in size from one virtual machine to the other.

Figure 5: Flash partitioning after first time install



The following steps will be performed as part of the first time install process:

1. The administrator/CTO site can login to the controller shell and issue a command to copy the solution bundle over the network over ssh or ftp
2. A RAM disk will be created. The solution bundle will be copied to the RAM disk and the top-level install script within the solution bundle will be invoked
3. The top-level install script will verify the CRC (MD5 checksum) of the solution bundle, create a logical disk partition for the solution bundle and then copy it over to the flash
4. The solution bundle will then be extracted and the install scripts will be invoked

5. The install script will verify the CRC (MD5 checksum) of all the individual VM images and create a logical disk partition for Domain0 and the guest VMs
6. The install script will then install the Xen+Domain0 image onto the logical partition created for Xen+Domain0, update grub to boot Xen and reboot the controller
7. The controller will now come up in a virtualized (hypervisor) environment. Domain0 will setup the initial configuration (if required), install the rpms for the guest VMs within their respective partitions and launch the virtual machines.
8. The logical partitions which host the individual VM images is marked as the ACTIVE area while the logical partition which stores the solution bundle (step 3) is marked as the STAGING area.

The logical disk partitions created for the individual VM images will be LVM (Logical Volume Manager) backed partitions. Each VM will have its own view of the flash drive through its LVM backed partition. Each partition can also be referred to as a VBD (virtual block device) for the virtual machine.

As part of the install process, the rpm install script will first copy the bootable image (created as part of the Solution bundle step 1) for the VM from the extracted solution bundle to the designated VBD for the VM. The standard UNIX based utility 'dd' (disk dump) will be used to copy the image in terms of multiple blocks to the designated VBD within the flash.

It is to be noted that both the ACTIVE and the STAGING area contain the same firmware version. The ACTIVE area contains the installed VM images while the STAGING area contains the un-extracted solution bundle. This bundle will be used for ACS purposes when a spare controller (running a different FW version) is inserted to the array. The VM images from the ACTIVE area cannot be used for ACS purposes.

Architecture Note: Using LVM backed partitions over raw disk partitions or image (.img) files provides a lot of flexibility when managing the VBD for each VM. Also the performance of LVM backed partitions when using the UNIX utility 'dd' was significantly higher than for raw partitions. However, for creating LVM backed partitions, support for grub version 1.96 or higher is required. It is expected that Citrix will be able to provide grub support of version 1.96 or higher.

2.2.1.1.3.Two copies of FW image

As depicted in Figure 5, the flash drive resident within the controller will host 2 copies of the system firmware. One copy would be the "installed" version of the firmware in the ACTIVE area. The other copy will be in the form of a solution bundle image stored within the STAGING area. This solution bundle image is required for ACS purposes.

ACS cannot be performed from the installed copy. For ACS to be performed from the installed copy of the firmware, each installed file will need to be copied from one controller to the other over the mid-plane. This is cumbersome considering that the number of installed files will be high. Installed files will comprise shared libraries, executables or even kernel modules. Moreover, some of the configuration files will also need to be copied individually from one controller to the other. This method will have issues if the file system cache is not in sync with the contents on disc for the file. This may result in copying corrupted or out of sync configuration files.

Copying individual files will also impair the ability to perform CRC verifications once the files have been copied over the internal network.

In order to circumvent the above, the entire solution bundle will be copied over the internal network to the controller which requires ACS to be performed. The solution bundle will first be copied over to the STAGING area, extracted and then installed before the controller reboots to come up with the new firmware. Please refer to the ACS and Controller Sparing AAD element for details on ACS.

2.2.1.1.4.Firmware download

The firmware download process will download a new firmware solution bundle to the STAGING area within the flash drive. As part of the download, it will overwrite solution bundle of the firmware stored within the STAGING area of the flash. The firmware download operation will be initiated by a SYMbol command. The SYMbol command will reach the IOVM guest VM within the controller. However, the download of the actual solution bundle of the solution bundle will be performed by using a ‘scp’ command over a ssh session which will be established from the management host to the controller.

The ssh configuration file within Domain0 (sshd.conf) will be modified to listen to a different port than the standard port 22 for ssh commands. The sshd daemon will then be restarted so that it can listen to the port configured within the sshd.conf file. All network traffic on the specified port will then be routed to Domain0 itself. The internal network within the controller will conform to the NAT method for routing data to specified VMs within the controller (please refer to the Hypervisor AAD – Virtualized HW Resources document number 44340-00 for more information on NAT). The SYMbol server will listen to port 2463. Therefore, all inbound packets on port 2463 will be redirected to IOVM by the NAT port forwarding mechanism. However, all inbound traffic on the new sshd port will be routed to Domain0 itself. It will therefore be copied to a destination filesystem on Domain0.

The firmware download will therefore, be managed by the management client using multiple sessions. Firstly the client will issue a SYMbol command. This command will be serviced by the SYMbol server running on IOVM. The SYMbol command will make the necessary verifications before initiating the download. The verifications include (but not limited to):

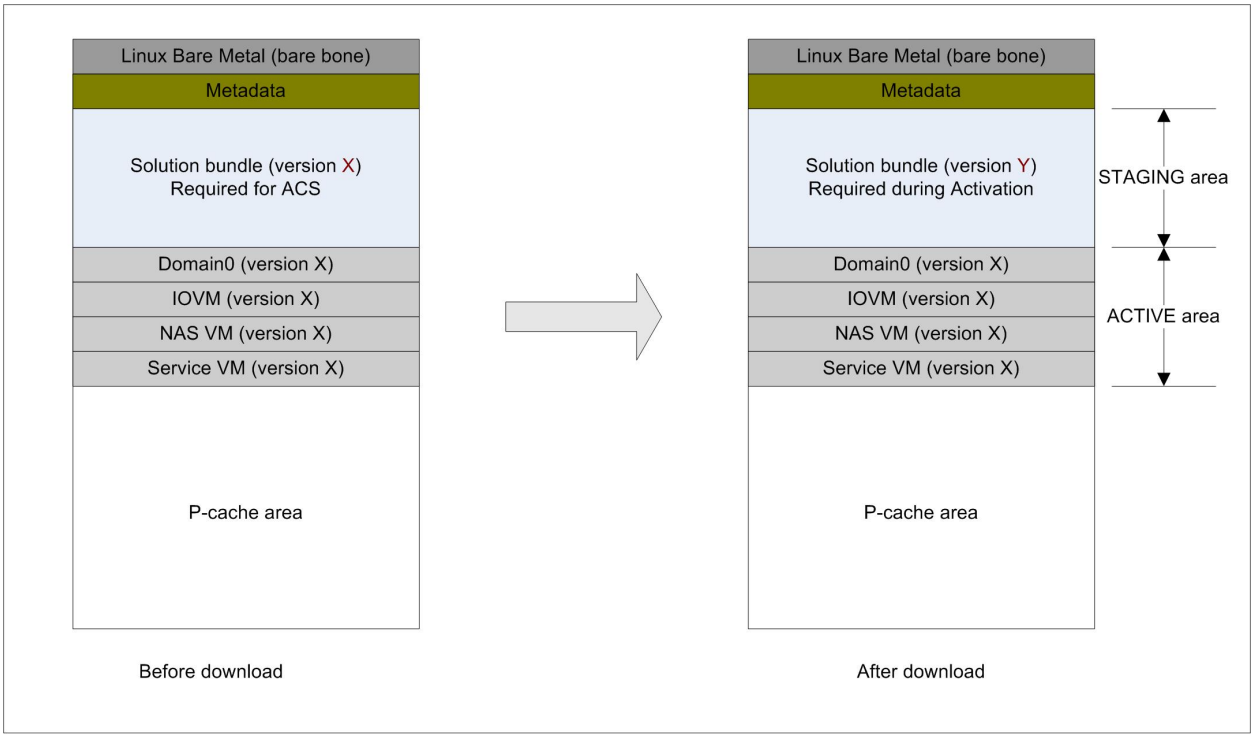
1. If both the controllers are optimal
2. If a download operation is already in progress in any one controller
3. If a cache restore operation is in progress

Once the preliminary checks have been performed, the management client will initiate the actual download operation. It will create a separate session which will perform the 'scp' operation.

The download operation will need to be completed on both the controllers

The following figure depicts the firmware solution bundle within the flash after completion of the download operation:

Figure 6: Flash state after download



Note that the STAGING area contains the solution bundle for FW version 'X' before the download operation but contains the solution bundle for FW version 'Y' after the download completion.

2.2.1.1.5.Effect on ACS and Controller Sparing

As depicted in Figure 6, the solution bundle for FW version 'X' will be overwritten by the solution bundle for FW version 'Y' as part of the download operation. Therefore, in case a controller is removed and a spare controller running a different version of firmware is inserted between the time window of a download operation and the subsequent activation, ACS cannot be supported for the spare controller. The spare controller will come up in SOD, identify that it is a FOREIGN controller and try to initiate ACS. However, when it sends the notification to the incumbent controller to start the ACS operation, the incumbent controller will identify that it contains a staged image within the STAGING area. Hence the ACS operation will be failed. The FOREIGN controller will not complete SOD. Instead it will perform a self lockdown.

However, if the FOREIGN controller has the same running firmware version as the incumbent controller, then the controller will come up and complete SOD. In such a scenario, the staged solution bundle will be discarded by the incumbent controller. The controller will also log a critical event indicating that the solution bundle of the currently installed firmware version be restored from the management host.

Architecture Note: Versions 'X' and 'Y' depicted above are not actual firmware version format strings but have been mentioned just as an example.

2.2.1.1.6.Restoring solution bundle

In the scenario that the staged firmware bundle (version 'Y' as depicted in Figure 6) is discarded as specified in Section 2.2.1.1.5. Effect on ACS and Controller Sparing or in the event the firmware activation fails, a critical MEL event will be logged by the controller indicating that the controller firmware has lost the capability for auto code sync. A Needs Attention will also be generated. The Recovery Action in such a

scenario will be to restore the solution bundle for the currently installed firmware version (version 'X' as depicted in Figure 6) from the management host.

A new SYMbol API will be provided to restore the solution bundle which was retrieved as part of the staging operation to the management host. The retrieved solution bundle which will be stored within a known location in the management host will be restored to the STAGING area of the flash as part of this command. This command will also need to check if the solution bundle which will be restored is the same version as the installed firmware running on the controller. If the version as stored on the management host is different, the command will fail the restore.

2.2.1.1.7.Firmware activation

Since Linux packages will need to be installed prior to being booted from, the activation process for OSA based firmware will consist of the following steps:

1. Extract the individual VM images from the solution bundle.
2. Install the firmware for the different VMs at their respective partitions.
3. Reboot from the newly installed firmware.
4. Commit the firmware.

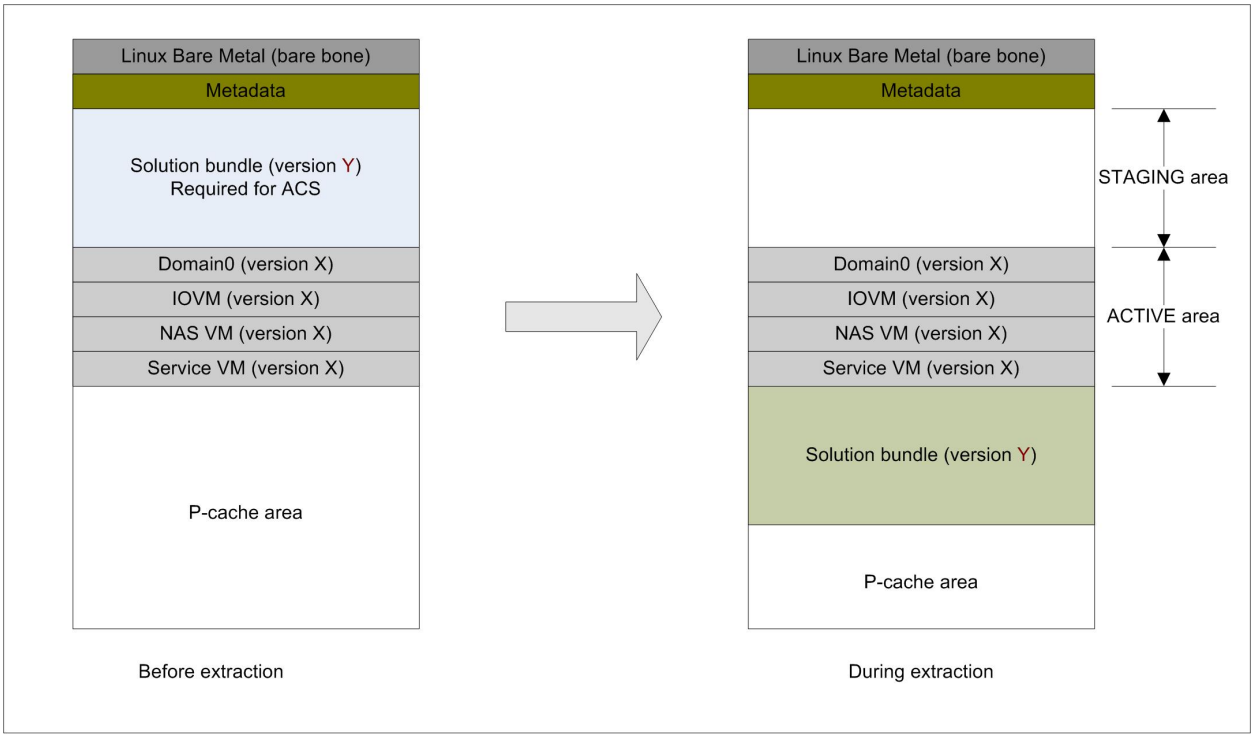
The entire activation process will be performed with the array being online, i.e. one controller at a time.

2.2.1.1.7.1.Using P-cache area in flash for FW extraction

A scratch area within the flash drive will be required to perform the extraction process. This process will consist of copying the solution bundle to the scratch area and extract the packages for each individual VM. CRC verification (MD5 checksum) will also be performed for each VM image after the extraction is completed.

Due to lack of available space for the scratch area on file system partitions on the flash, the cache offload or the P-cache area within the flash will be used as the scratch area for the extraction process. Therefore, the solution bundle will first be copied to the P-cache area for the extraction.

Figure 7: During extraction



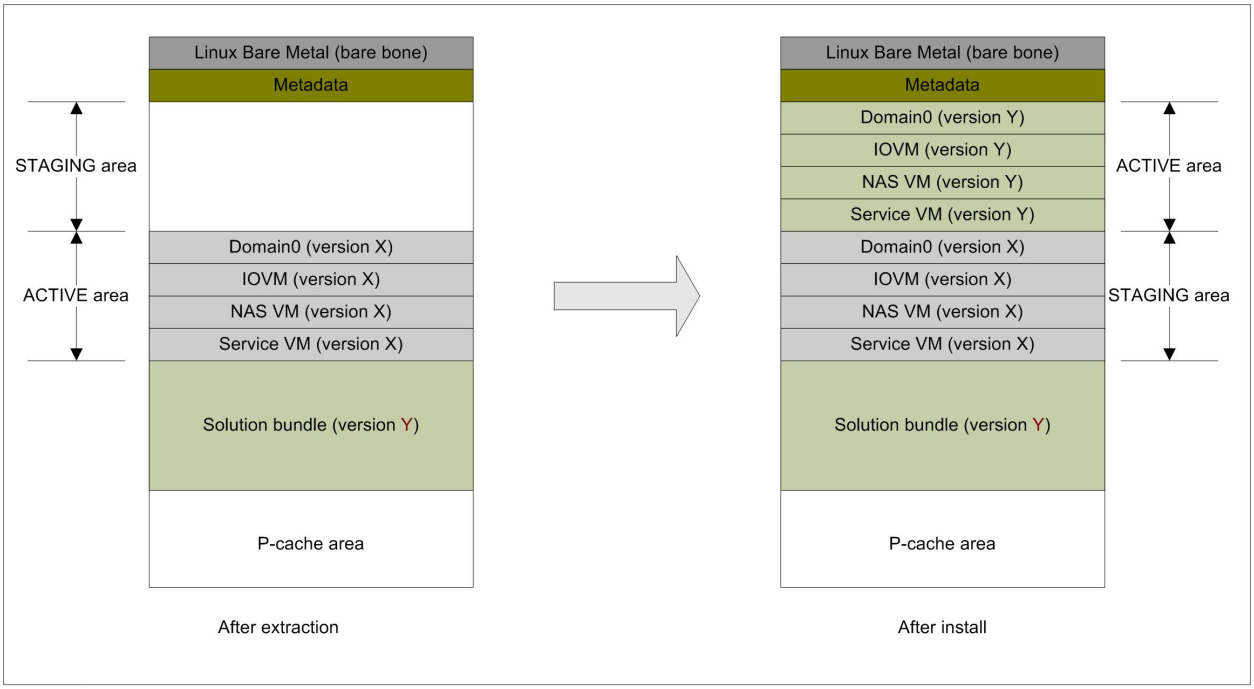
The following steps will be performed during an extraction process:

1. Disable write caching for all volumes in the array. The volumes will disable write caching and sync cache.
2. Transfer all volumes owned by this controller to the Alternate controller.
3. Close the xenbus block backend driver connection. Set state to *XenbusStateClosing*.
4. Trap the xenbus event on the front-end driver in IOVM and dissociate the cache offload device by sending a drive removal event to the interested components.
5. Move the front-end state to *XenbusStateClosed*. This will also ensure that the back-end state for the cache offload device is *XenbusStateClosed*.
6. Domain0 will format the cache offload area and mount a temporary (ext3) file system.
7. The solution bundle for firmware version 'Y' will be copied to the cache offload area and extracted. CRC verification (MD5 checksum) will be performed on the individual packages. Note that the cache offload area at this stage will contain both the solution bundle and the extracted images.
8. All pre-install verifications will be performed at this point. Once the pre-install checks have been performed, it will proceed with the installation steps.

2.2.1.1.7.2.Installing new firmware

The installation process (of the activation phase) will constitute of installing the new firmware for all virtual machines to their specified partitions or VBDs.

Figure 8: Installation Process



1. If the pre-install steps have succeeded, then Domain0 will proceed to create partitions for each virtual machine within the specified region in the flash as depicted in the figure above.
2. Based on the footprint of the individual virtual machines image as located within the cache offload area, LVM backed partitions will be created for Domain0 and all the guest virtual machines.
3. Once the LVM backed partitions have been created, the install script will use the services of the UNIX based disk dump utility 'dd' to move the Xen+Domain0 bootable image to the partition that is designated for Domain0 (see above figure).
4. The current STAGING area will now be marked as the ACTIVE area while the current ACTIVE area becomes the STAGING area.
5. The install scripts will change the grub to boot Domain0 from the newly installed image. The controller will be rebooted.
6. Once Domain0 boot up with the new firmware, it will copy the images for all the guest virtual machines to their designated partitions by using the UNIX based 'dd' utility. It is necessary for Domain0 to reboot first and then install the other guest VMs so that the rpm database on the new Domain0 is updated with rpm data of the newly installed packages (of the virtual machines). If the other VMs were installed prior to Domain0 rebooting with the new FW, then the rpm database on the old Domain0 would have been updated.
7. Once the guest VMs have been installed, Domain0 will spawn the guest VMs from the newly installed FW and a new grub.
8. Once IOVM is in the process of booting up and executing its Start-of-Day with the new firmware, the P-Cache components will discover the cache backup device. However, the device will not be claimed by the components. Instead the lower level component DVC will not send drive ready events to the upper level components in the Applications layer. Instead, DVC will keep an in-memory state of the drive ready events till the commit phase.
9. Each guest VM will notify Domain0 on SOD completion. The partitions on the flash disk with firmware version 'Y' will resemble the diagram in Figure 3. (Note: The firmware version 'Y' has not been committed as yet).
10. Once the IOVM guest VM comes back up, the download management component will initiate the takeover of all volumes from the Alternate controller.

11. The download management component will now indicate to the peer controller to start extracting and installing the new firmware image.

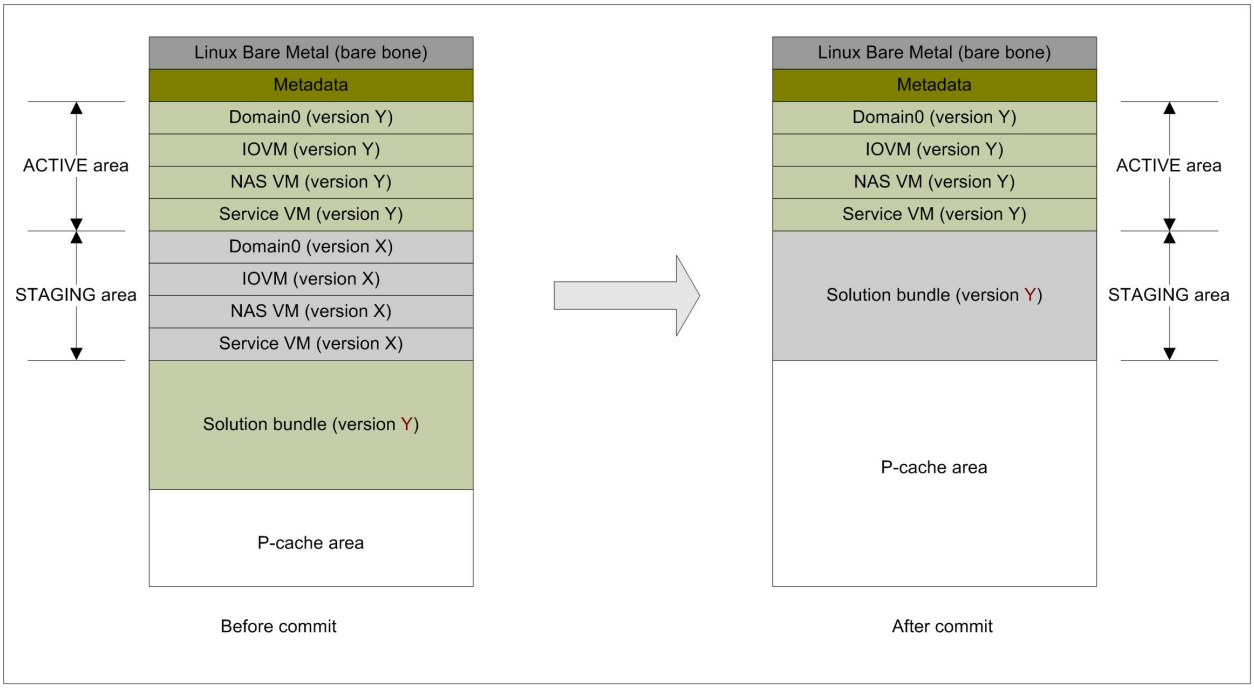
2.2.1.1.7.3.Activation commit

Once both the controllers have completed their installation and all guest VMs have completed SOD processing, the controller "system" SOD is considered complete. The "system" SOD completion is an event which indicates that activation is successful for all VMs running on the array. This is also called as the "commit" phase of the activation. As part of the system SOD completion (commit phase), the following actions are taken:

1. The primary controller will first boot with the new firmware.
2. Once all guest VMs complete their Start-of-Day, the notify Domain of the same. When both controllers come up, the volumes are rebalanced.
3. The primary Domain0 will be responsible for coordinating with the alternate controller to ensure all VMs have completed their Start-of-Day. Once all guest VMs on both the controllers have completed their Start-of-Day, Domain0 on both controllers will repartition their STAGING area to one big partition that will be managed by Domain0.
4. After repartitioning the flash, the solution bundle will be copied from the cache offload area to the new STAGING area. This solution bundle will be used for ACS purposes.
5. The cache offload area can now be claimed by the P-Cache components. The download management component will notify the DVC component to release the drive ready events to the components running within the Applications layer.
6. Write caching will be enabled for all volumes in the array.
7. SYMbol command restriction will now be removed.

The following diagram depicts the flash drive partitions after a successful commit:

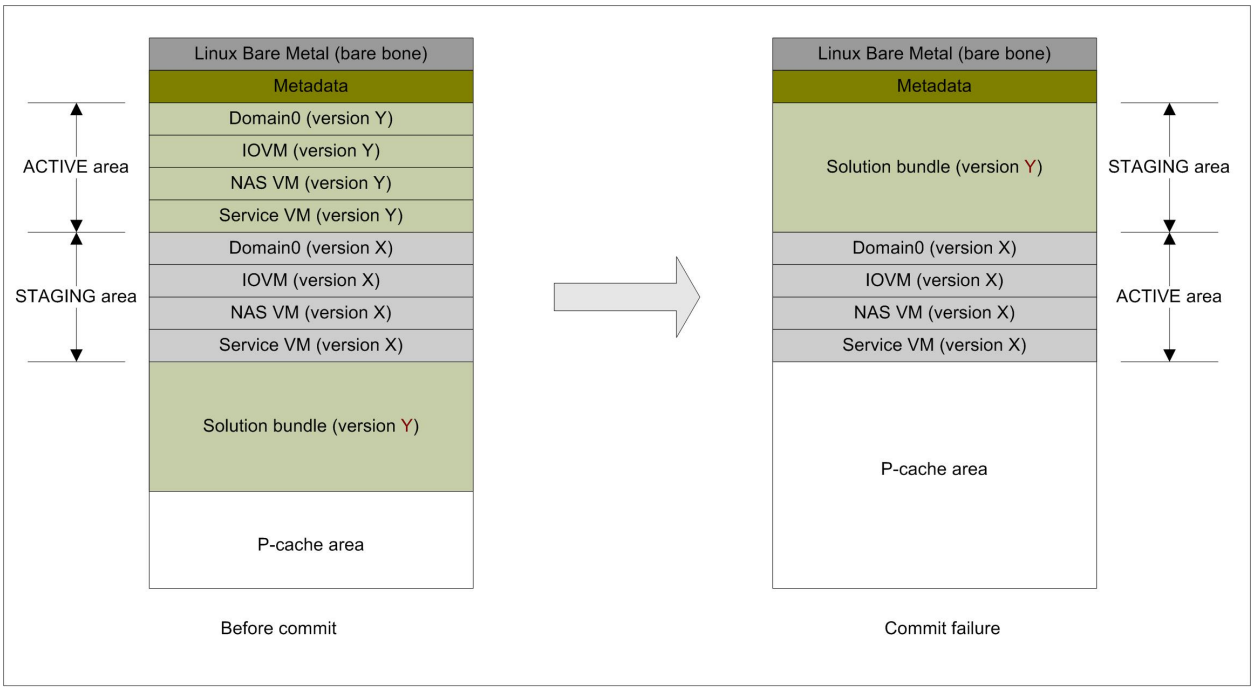
Figure 9: Commit phase during activation



2.2.1.1.7.4.Commit failure

If any of the guest VMs is unable to complete its Start-of-Day operations with the newly installed firmware, the firmware cannot be committed for the array. The firmware will perform a rollback to the previously running version of the firmware. The download management component running on Domain0 of the primary controller will wait for a timeout period from the time Domain0 on the primary controller completes its initialization. If all guest VMs on both the controllers do not complete their initialization within the specified time frame, then the download management component will initiate a rollback to the previously running version of firmware.

Figure 10: Flash partition after rollback



The following actions will be performed with respect to the flash partitioning during a rollback operation:

1. In the scenario that the timeout expires and all guest VMs have not notified the primary Domain0 of SOD completion, the download management component will trigger a rollback operation.
2. The rollback will be performed one controller at a time and hence the volumes will be transferred to the alternate controller when one controller initiates a rollback operation.
3. The new ACTIVE area will again be marked as the STAGING area and the new STAGING area will be marked the ACTIVE.
4. The grub will be modified to boot from the new ACTIVE area which contains firmware version 'X'. The controller is rebooted.
5. Once Domain0 boots from version 'X' of the firmware, it will also spawn the guest VMs from the same version.
6. Once all VMs complete start-of-day with version 'X', the download management component on Domain0 of both the controller will repartition the flash to contain a single partition for the STAGING area.
7. The solution bundle for firmware version 'Y' is now copied back to the STAGING area.

8. The cache offload area will now be reclaimed by the P-Cache components. Write caching will be enabled.
9. SYMbol command restriction will be removed.

2.2.1.1.7.5.Restricting SYMbol commands during activation

User initiated configuration changes are not allowed during activation till the activation successfully completes the commit phase or till the firmware is rolled back to the previous version in case of an activation failure. Therefore, all SYMbol commands which are used for user initiated configuration changes will not be allowed during activation. These commands will return an error indicating that activation is in progress and the command needs to be tried later.

2.2.1.1.7.6.Controllers running different FW versions

The capability of rollback increases the time window when both controllers run different versions of FW during activation. Therefore, all components will need to verify the FW version across both the controllers before making a call to the peer component on the alternate controller.]

All new components introduced with the OSA firmware and beyond will need to check the firmware version before checking-in the first time after the upgrade. All components will need to have a late-checkin facility which will be performed only when the alternate controller boots with the new firmware.

2.2.1.2.Advanced Development Evaluation

There is NO Advanced Development evaluation for this feature.

2.2.1.3.Component Collaboration

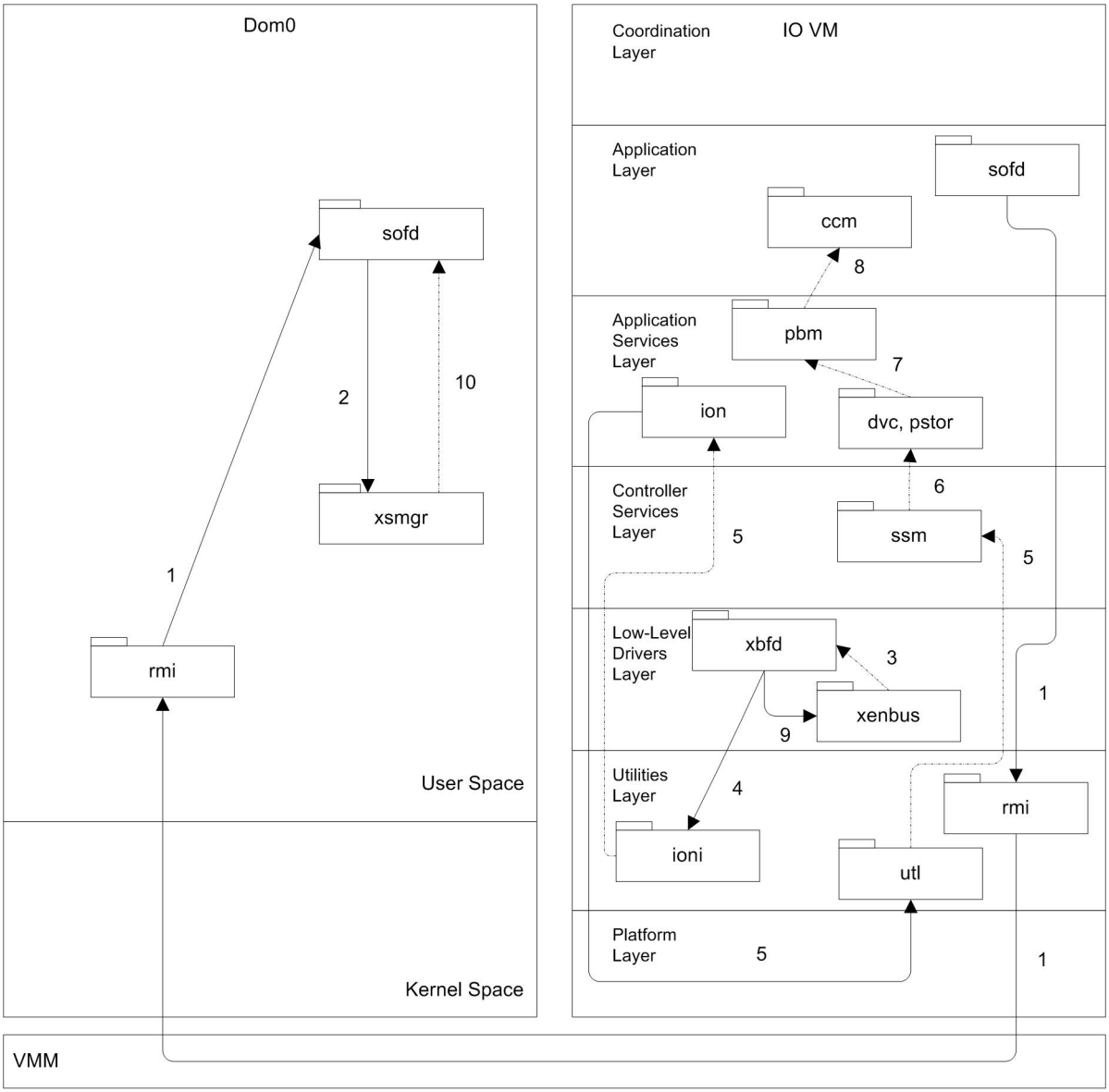
This section specifies the collaboration between components required to implement various aspects of the functional behavior.

2.2.1.3.1.Claiming cache offload device

During *the activation phase*, the cache offload device portion of the iSATA flash drive will be claimed by the download management component (sofd) for performing the necessary extraction and validation of the packages. Therefore, during activation the controller cannot enter a cache backup mode. All volumes owned by the array will disable write caching during this period.

The following collaboration diagram specifies the sequence of operations required to claim the cache backup device

Figure 11: Claim cache offload device during activation



1. [IOVM] During the activation phase, the sofd component running on IOVM initiates the sequence of operations to claim the cache off load device for package extraction and verification. The sofd component invokes a RMI call to notify sofd component running on Domain0 to initiate the claim operation for the device.
2. [Domain0] Once the sofd component running on Domain0 receives the instruction from [IOVM] sofd, it calls an API within the xsmgr component to start claiming the cache offload device. The xsmgr manager component attempts to close the xenbus block back-end driver connection. Sets state to *XenbusStateClosing*.
3. [IOVM] A xenbus event is sent to the xenbus block front end driver (xbfd) running on IOVM.
4. [IOVM] The xbfd driver gets an event that the back-end device is closing. It calls an API provided by the ioni component to indicate that the back-end connection for the cache offload device is down.
5. [IOVM] The *ITN down* event is propagated to the ion component which translates propagates it to the utl component. The *ITN down* event for the cache offload device finally reaches the ssm component.

6. [IOVM]The ssm component creates a *DriveRemoval* object and propagates the *driveRemoval* event to the interested listeners which in this case being the dvc component. The dvc component in turn propagates the event to the pstor component.
7. [IOVM] The pstor component propagates the event to the pbm component.
8. [IOVM] The pbm component notifies the interested listeners which are the ccm component and the ncb component. The ccm component is shown here. The ccm component disables write caching for all volumes since the cache backup device is unavailable.
9. [IOVM] The xbfd component finally sets the front-end state as *XenbusStateClosed*. This sets the back-end state as *XenbusStateClosed*.
10. [Domain0] The xsmgr manager running in Domain0 informs the [Domain0] sofd that the cache offload device has been dissociated. The sofd component will now claim this device. It will format the cache offload area and mount a temporary (ext3) file system on this device. Please refer to section 2.2.1.1.7.1 for details.

2.2.1.3.2.Firmware download - primary controller

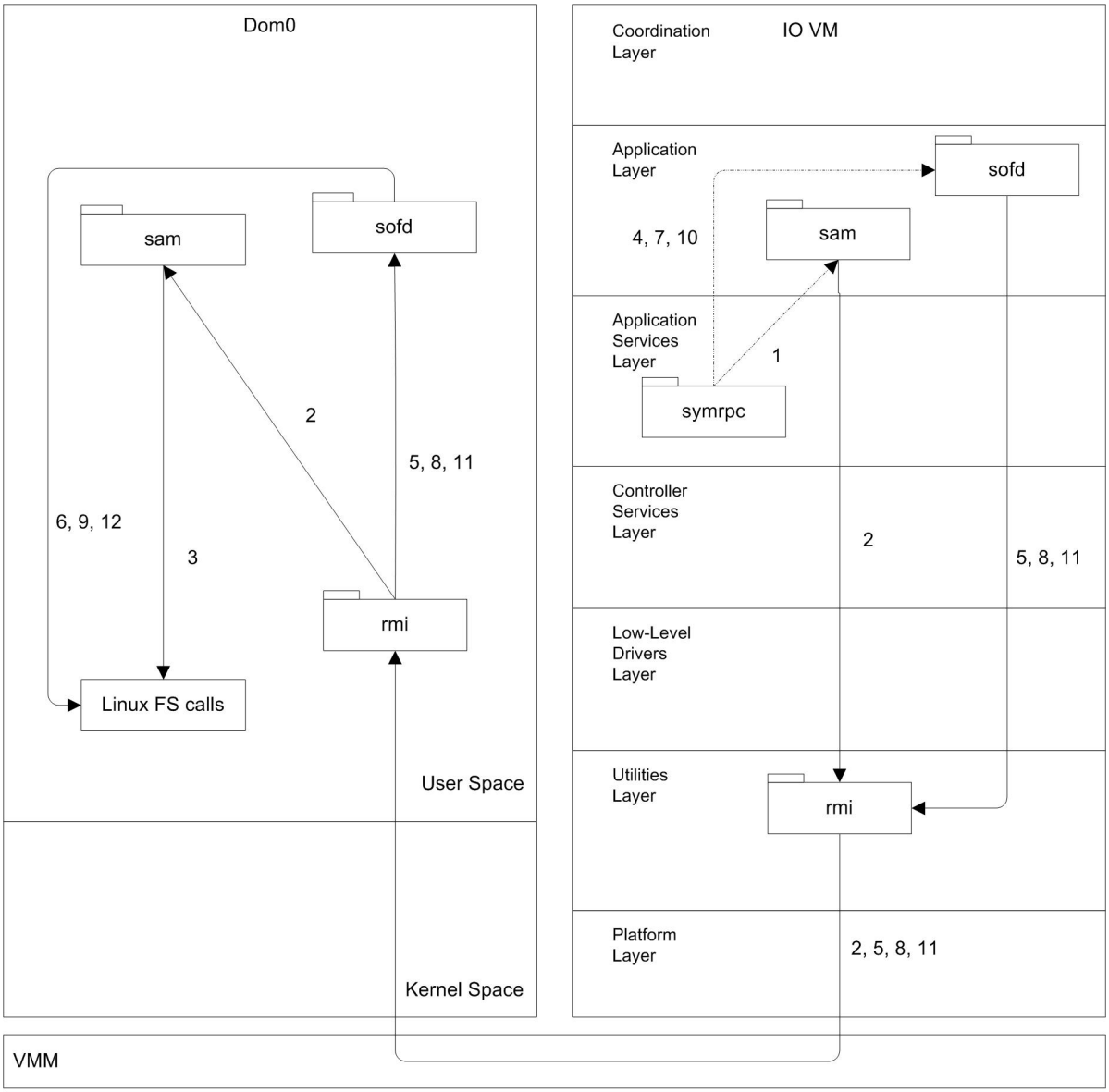
The first step as part of the download operation will be to retrieve the solution bundle for the currently running firmware to the management host. This saved bundle will be restored back to the controller in case the upgrade operation fails. The solution bundle will be stored within a known location within the management host.

Once the existing solution bundle has been retrieved the management client starts the actual download of the new solution bundle to the STAGING area.

The management client will set up a ssh session for the actual retrieve and download operations. It will get the necessary authentication information from the firmware and also all necessary configuration information for the download operation.

The following collaboration diagram specifies the sequence of operations for a download operation on the primary controller:

Figure 12: Firmware download to primary controller



1. [IOVM] The management client calls the *getDownloadConfiguration* SYMbol command. The call reaches SYMbol RPC server running on IOVM. The SYMbol server calls the registered handler for this routine which will be handled by the sam component.
2. [IOVM] The sam component invokes a RMI call to get the download configuration information from the same component running on Domain0.
3. [Domain0] The sam component on Domain0 makes necessary file system API calls to retrieve the configuration information. The configuration information includes ssh authentication keys, ssh port number and also the full download path on Domain0 file system.
4. [IOVM] The management client invokes the *saveFirmwareBundleToHost* API call. The SYMbol RPC server receives the call and invokes the handler registered by the sofd component.
5. [IOVM] The sofd component performs necessary checks as identified in section 2.2.1.1.4. After performing the checks, it invokes a RMI call to mark the necessary state information within Domain0. The call reaches the sofd component running on Domain0.

6. [Domain0] The sofd component on Domain0 marks the necessary state information within the Domain0 file system. Once the call returns, the management client will set up the ssh session and copy the solution bundle to the known location. Steps 4 - 6 are repeated for the *saveFirmwareBundleToHostComplete* SYMbol API.
7. [IOVM] The management client invokes the *startOSAFirmwareDownload*API call. The SYMbol RPC server receives the call and invokes the handler registered by the sofd component.
8. [IOVM] The sofd component invokes a RMI call to mark the necessary state information within Domain0. The call reaches the sofd component running on Domain0.
9. [Domain0] The sofd component on Domain0 marks the necessary state information within the Domain0 file system.
10. [IOVM] The management client invokes the *OSAFirmwareDownloadComplete*API call. The SYMbol RPC server receives the call and invokes the handler registered by the sofd component.
11. [IOVM] The sofd component invokes a RMI call to mark the necessary state information within Domain0. The call reaches the sofd component running on Domain0.
12. [Domain0] The sofd component on Domain0 marks the necessary state information within the Domain0 file system.

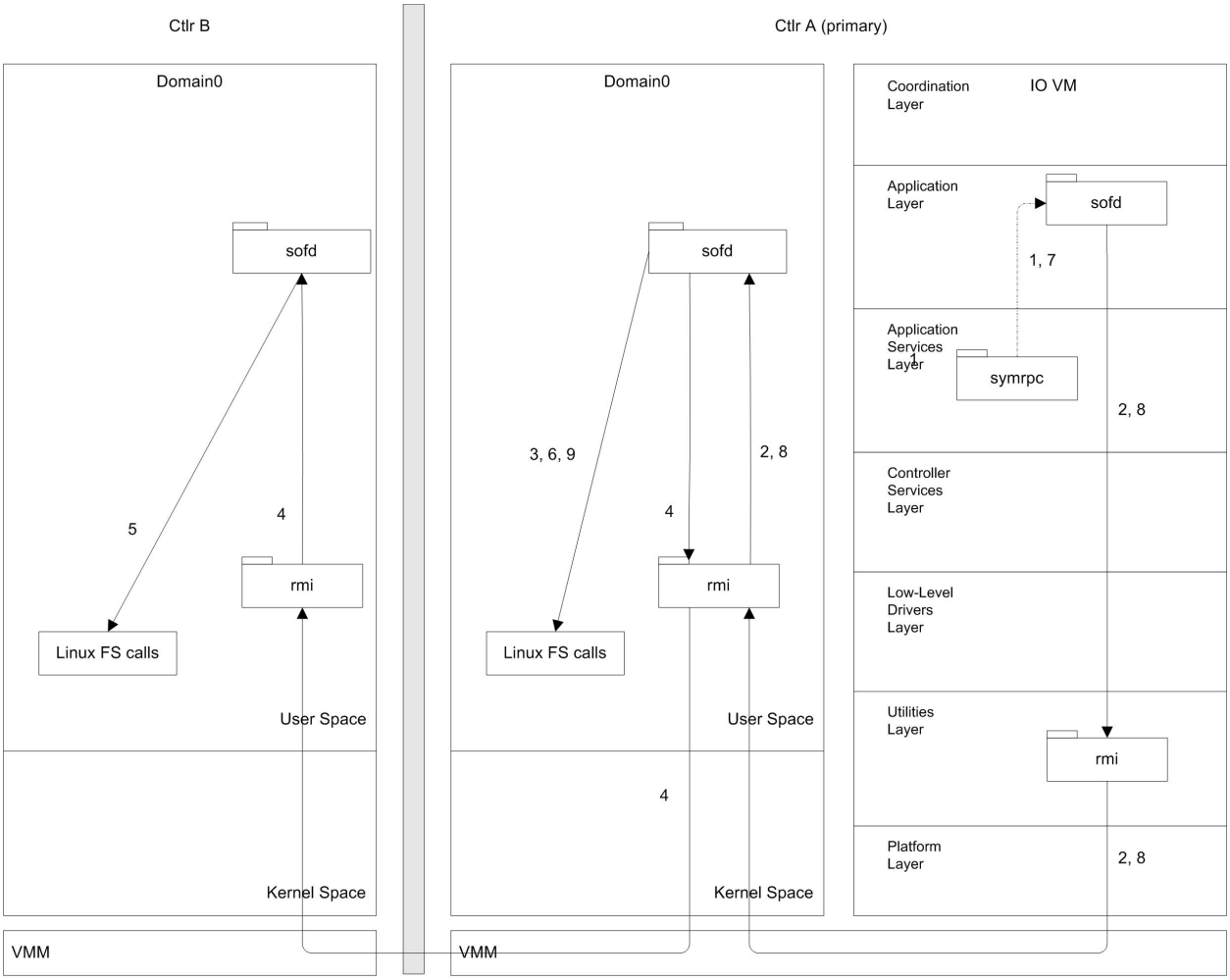
2.2.1.3.3.Firmware download - peering to alternate controller

As soon as the management client invokes the *OSAFirmwareDownloadComplete* SYMbol API, the sofd component marks the download status within the Domain0 file system. It also starts peering the solution bundle to the alternate controller. The management client will return the call to the user only when the peering of the solution bundle to the alternate controller is complete. Else, the download operation is failed and the newly downloaded solution bundle to the primary controller is discarded.

Therefore, as soon as the *OSAFirmwareDownloadComplete* call returns, the management client will invoke the *getSolutionBundlePeeringStatus* SYMbol API. This API will not return until the solution bundle has been copied to the know location within the alternate controller's Domain0.

The following collaboration diagram specifies the sequence of operations for a download operation on the primary controller:

Figure 13: Peering the solution bundle



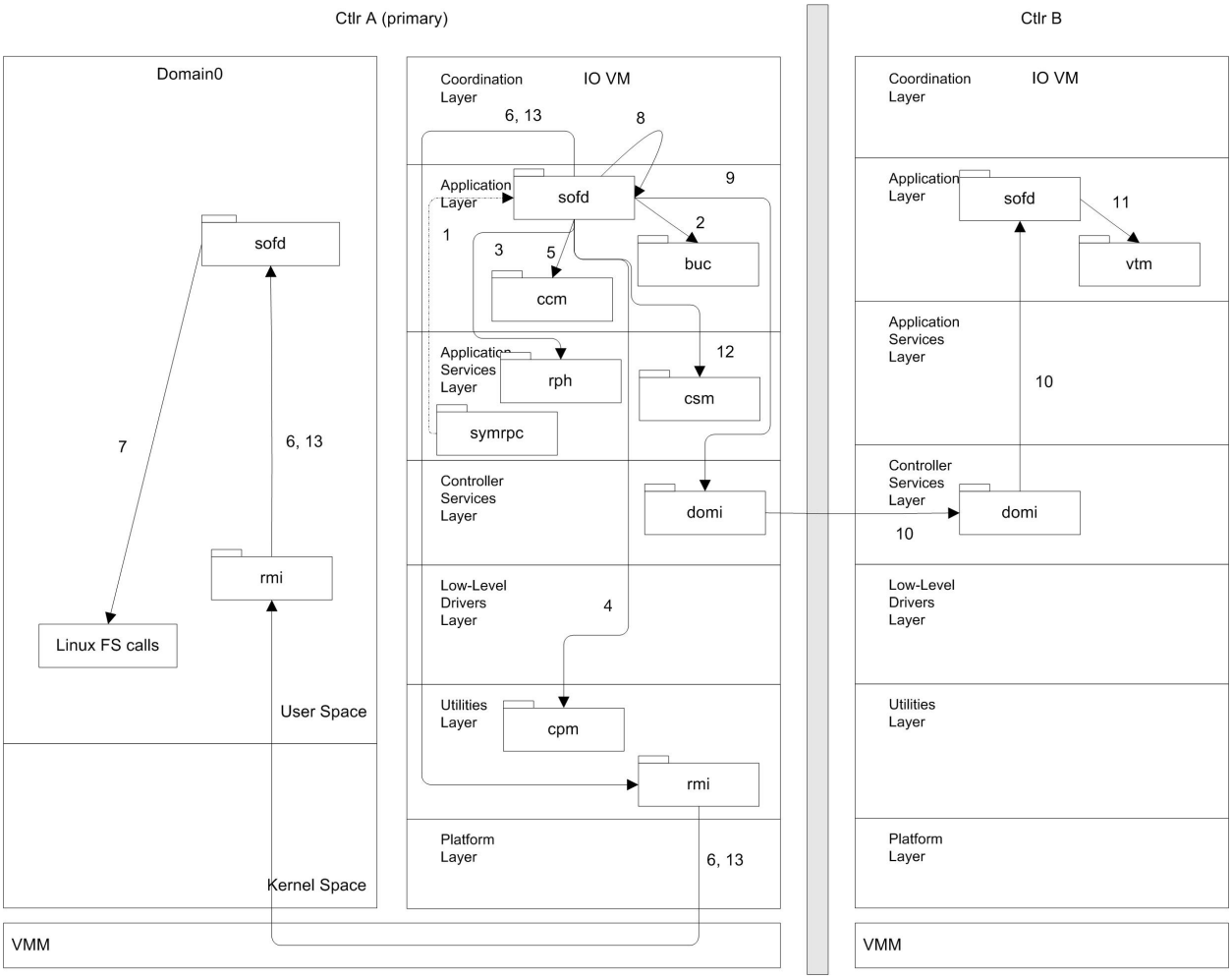
1. [Ctlr A, IOVM] The management client invokes the *OSAFirmwareDownloadComplete*API call. The SYMbol RPC server receives the call and invokes the handler registered by the sofd component.
2. [Ctlr A, IOVM] The sofd component invokes a RMI call to mark the necessary state information within Domain0. The call reaches the sofd component running on Domain0.
3. [Ctlr A, Domain0] The sofd component on Domain0 marks the necessary state information within the Domain0 file system.
4. [Ctlr A, Domain0] The sofd component initiates a RMI call for [Domain0] sofd on Controller B indicating that it will initiate a copy of the firmware solution bundle.
5. [Ctlr B, Domain0] The sofd component marks the status in the Domain0 file system of Controller B.
6. [Ctlr A, Domain0] The sofd component then initiates a 'ssh' session for a 'scp operation to a known location within the controller B flash. Once the 'scp' operation is complete [Domain0] sofd on Controller A writes the state information to the Domain file system.
7. [Ctlr A, IOVM] The management client will wait on the *getSolutionBundlePeeringStatus* SYMbol API call. The call reaches the sofd component.
8. [Ctlr A, IOVM] The sofd component on IOVM invokes a RMI call so that the sofd component on Domain0 can be invoked.
9. [Ctlr A, Domain0] The call will check the status of the peering of the solution bundle. Once the peering completes and the peering status has been updated in step 6, the call will indicate that the peering of the solution bundle to the alternate controller is complete.

2.2.1.3.4.Firmware activation - primary controller

The activation process is triggered by the *activateOSAFirmware* SYMbol command. The firmware is first moved to the cache offload area, extracted and verified and then installed into the various LVM backed partitions for each virtual machine. Domain0 will be the first virtual machine which gets installed and boots with the new firmware. Once domain0 boots up, it installs the firmware for the other VMs and spawns the virtual machines.

The following collaboration diagram specifies the sequence of operations for a download operation on the primary controller:

Figure 14: Firmware activation - primary controller



1. [Ctlr A, IOVM] The management client invokes the *activateOSAFirmware* API call. The SYMbol RPC server receives the call and invokes the handler registered by the sofd component.
2. [Ctlr A, IOVM] The sofd component invokes an API provided by the buc component to ensure that no cache restoration operation is taking place. If a cache restoration is occurring, the command is returned with an error.
3. [Ctlr A, IOVM] The sofd component registers with the rph component to restrict the SYMbol calls that

are allowed during the activation phase.

4. [Ctlr A, IOVM] The sofd component writes the first checkpoint during activation using the service of the cpm component. Note that checkpoints will be written throughout the entire activation phase at different logical points.
5. [Ctlr A, IOVM] The sofd component calls an API provided by the ccm component to disable write caching for all volumes within the array.
6. [Ctlr A, IOVM] The sofd component invokes a RMI call to notify to [Domain0] to verify that the staged firmware solution bundle is valid.
7. [Ctlr A, Domain0] The sofd component on Domain0 checks the validity of the staged firmware solution bundle. It also checks if the alternate controller has a valid staged solution bundle.
8. [Ctlr A, IOVM] The sofd component initiates the claim of the cache offload device for activation. It initiates the steps as identified in section 2.2.1.3.1. Further to that [Domain0] sofd copies the solution bundle to the cache offload area, extracts the individual images from it and verifies them for CRC (MD5 checksum) as specified in section 2.2.1.1.7.1.
9. [Ctlr A, IOVM] The sofd component initiates the transfer of all volumes owned by this controller to the alternate. It invokes a domi call. The sofd component waits till the alternate has finished taking over all the volumes.
10. [Ctlr B, IOVM] The domi call reaches the sofd component running on IOVM of the alternate controller.
11. [Ctlr B, IOVM] The sofd component calls an API within the vtm component to takeover all the volumes owned by Ctlr A.
12. [Ctlr A, IOVM] The sofd component takes the Sofd SYMbol lock so that the volumes are not explicitly failed back.
13. [Ctlr A, IOVM] The sofd component invokes a RMI call to the sofd component running on Domain0. The call reaches [Domain0] sofd. [Domain0] sofd performs all the steps as identified in section 2.2.1.1.7.2.

Architecture Note: The activation phase prior to the commit for Controller B (secondary controller) also takes the same sequence of operations as the primary controller (from step 8 above).

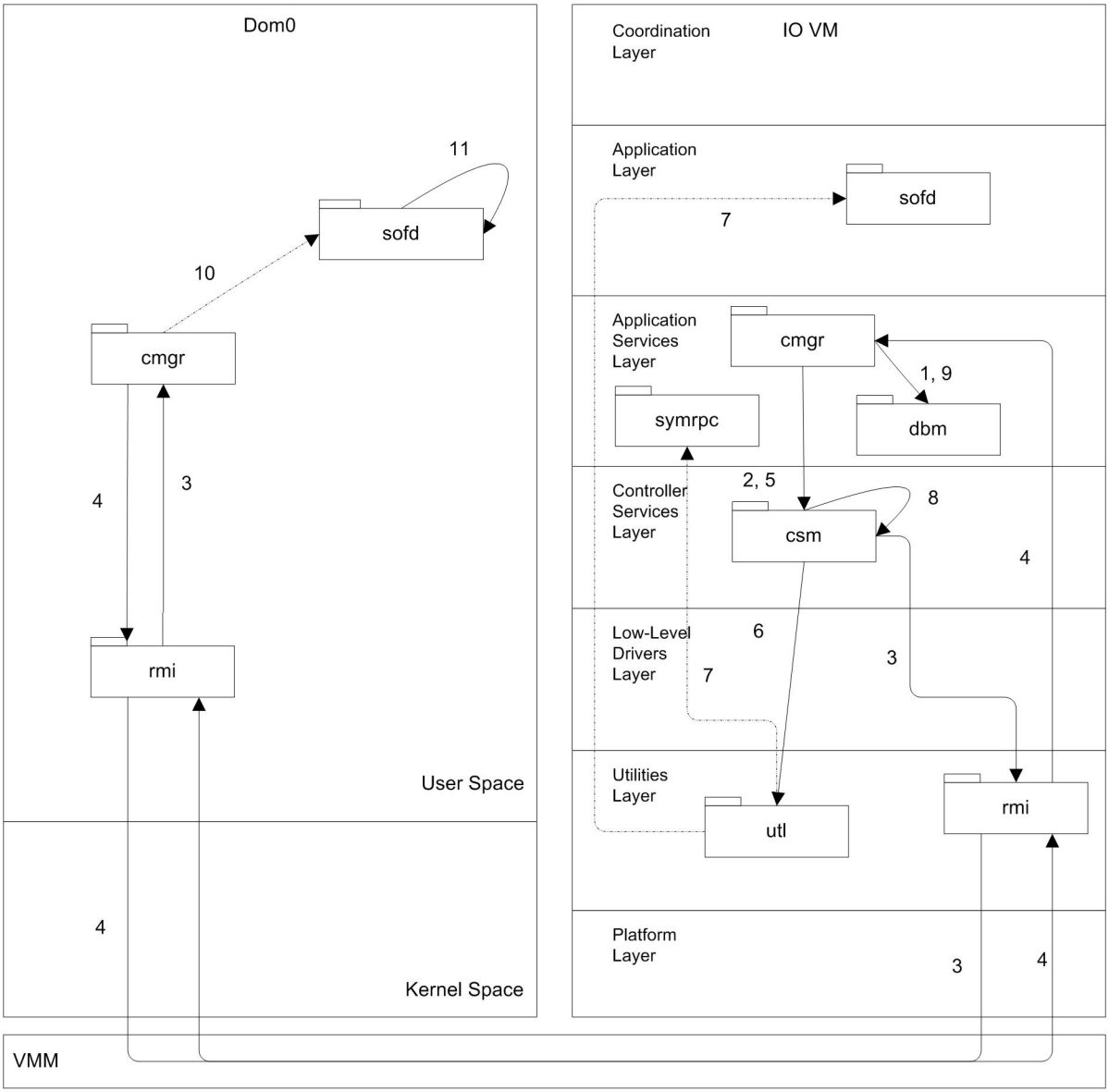
2.2.1.3.5.Activation commit

Domain0 will be the first VM to boot with the new firmware. Once it boots from the new firmware, it installs the other guest VMs and launches them. Please refer to section 2.2.1.1.7.2. Each guest VM will then boot up and execute its Start-of-Day sequence with the new firmware. The guest VMs will not allow external access either through a management client (e.g. SYMbol) or through a host IO initiator till the activation has reached a commit point. Therefore, all VMs will enable management access and host IO access only when "System SOD" is complete. This will be true only on the first reboot after an upgrade. Each VM, therefore, will be responsible for the intelligence required to understand the first reboot after upgrade and take appropriate action.

This section will identify the sequence of events when IOVM on the controller reboots with the new firmware to the completion of "System SOD" when the activation will reach the "commit" phase. The controller will release IOs and provide access to SYNbol server only when system SOD is complete. If system SOD cannot be completed then the firmware will attempt a rollback. Hence IO release event and SYMbol access will not be provided till the system SOD is complete.

The following collaboration diagram will specify the sequence of activation commit with respect the events for the primary controller during activation:

Figure 15: Activation commit



1. [IOVM] When the controller reboots with the new firmware, the cmgr component running on IOVM reads the controller record from dbm. It identifies that the current firmware version has changed. It also identifies that the controller is native. Hence a firmware upgrade has occurred.
2. [IOVM] The cmgr component calls an API provided by the csm component to let csm know of the first reboot after FW upgrade.
3. [IOVM] On SOD completion, the csm::SODComplete will be invoked. The csm component will identify that it is a first reboot after upgrade and hence will not send the ControllerSODComplete event at this stage. Instead it will invoke a RMI call to notify the cmgr component running on IOVM that the IOVM SOD is complete.
4. [Domain0] The cmgr component will wait for SOD complete from all guest VMs which include the guest VMs from the alternate controller. Once all the guest VMs have completed SOD and notified [Domain0] cmgr, it will invoke a RMI call to [IOVM] cmgr to indicate that all guest VMs have successfully complete their SOD [process and hence the activation phase can be committed.

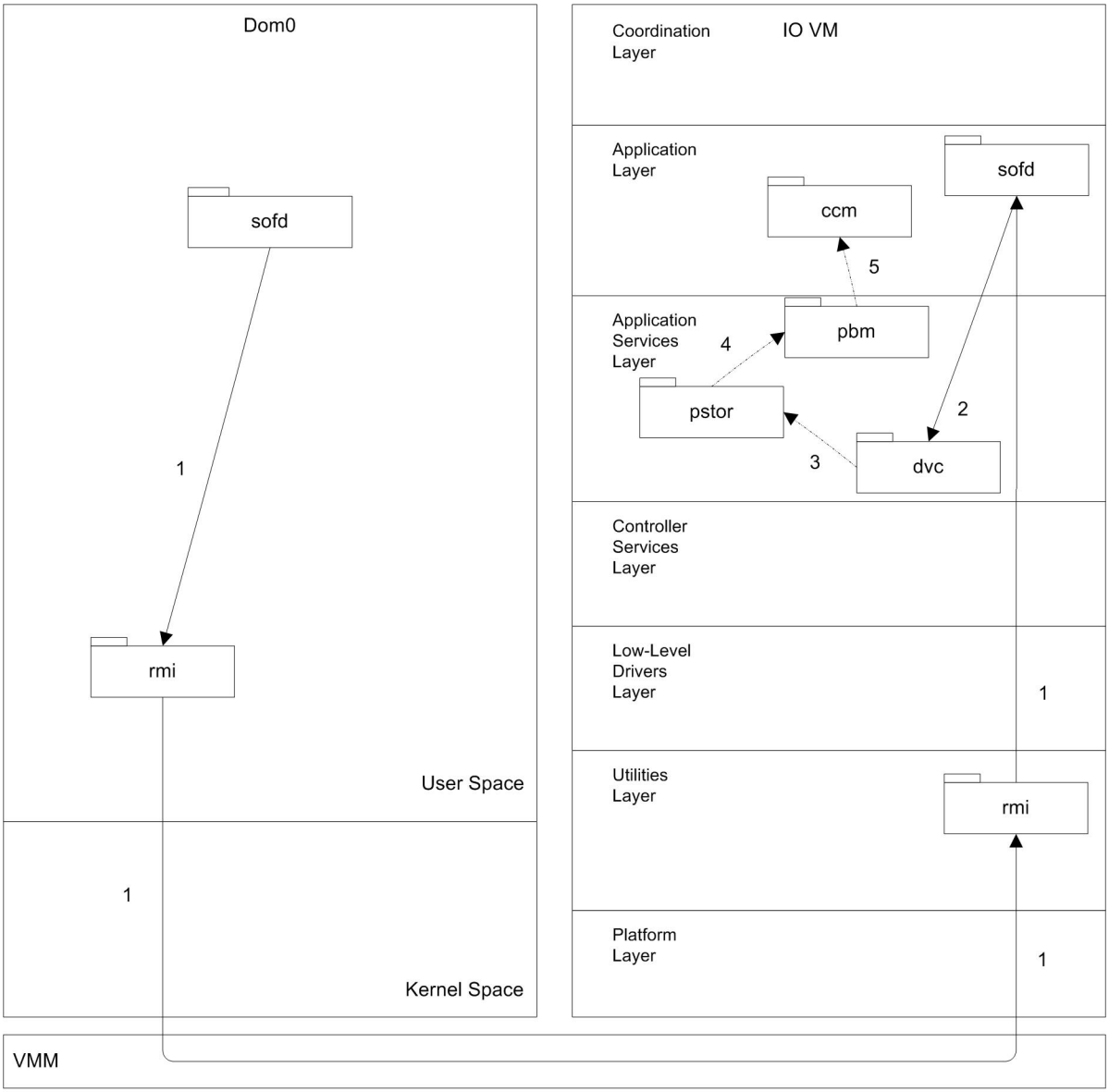
5. [IOVM] The cmgr component on IOVM will call an API provided by the csm component to indicate that "System SOD" is complete.
6. [IOVM] The csm complete will call an API provided by utl to indicate that System SOD is complete. This will be a new utl Event called SystemSODCompleteEvent.
7. [IOVM] The utl component will in turn notify registered listeners that System SOD is complete. The registered listeners in this case being the symbol rpc server and the sofd component. The symrpc component will allow management access through SYMbol and the sofd component will begin the reclaim of the cache offload device.
8. [IOVM] The csm component will then release the ControllerIOsReleasedEvent. Interested listeners will be notified.
9. [IOVM] The cmgr component will finally update the controller record in dbm with the new firmware version.
10. [Domain0] The cmgr component in Domain0 will similarly notify all guest VMs that System SOD is complete. Finally it will notify the sofd component that System SOD is complete and all event notifications can be performed.
11. [Domain0] The sofd component will now perform steps 3 and 4 as identified in section 2.2.1.1.7.3.

2.2.1.3.6.Reclaim cache offload device

The IOs will be released as part of the system SOD complete event. However, the volumes will still be in write through (WT) mode till the cache offload device has been reclaimed by the p-cache components. The cache offload region will be reclaimed as soon as the sofd component running in Domain0 has completed the movement of the firmware solution bundle to the new STAGING area as part of the commit phase.

Note that the cache offload region will be visible to the lower level drivers in IOVM and also the dvc component in IOVM as soon as the controller boots with the new firmware. However, the dvc component will identify from a NVSRAM record that the cache offload region was used for firmware upgrade purposes. Hence, the dvc component will not release the *driveReady* event to the upper layers till system SOD has been completed. When system SOD is completed, dvc will be informed by the sofd component. The dvc component will now release the *driveReady* events for the cache offload device. The ccm component in the Application layer will be notified and it will enable write caching for all volumes.

Figure 16: Reclaim cache offload device



1. [Domain0] The sofd component will move the solution bundle from the cache offload area to the new STAGING area as identified in section 2.2.1.1.7.3. After moving the solution bundle, the sofd component will invoke a rmi call to notify [IOVM] sofd that the solution bundle has been moved and the cache offload device can be reclaimed.
2. [IOVM] The sofd component running on IOVM will call an API provided by the dvc component to let know that the cache offload device has been released and hence the *driveReady* event can also be released to the upper layers.
3. [IOVM] The dvc component releases the *driveReady* event for the cache offload device. It notifies the registered listener pstor.
4. [IOVM] The pstor component processes the *driveReady* event. It performs re-initialization of the metadata wherever required and notifies the pbm component.
5. [IOVM] The pbm component in turn releases the *cacheBackupDeviceAvailable* event to the interested listeners, ccm and ncb. The ccm component on receiving the event enables write caching for all

volumes.

2.2.2.Core Assets

This section specifies the core assets affected by the feature. Each core asset is specified as its own subsection below. For those core assets that are components, diagrams may be used to illustrate new dependencies being added to an existing component or a diagram may also be used to illustrate how any new component(s) fit into the existing component layer(s) as well as their dependencies.

Any new core assets or updates to the overall CFW component dependency model will result in corresponding updates to the Controller Firmware Architecture Specification.

Also specified in this section will be which components will contain new variation points and a brief description why and how the variation points are being implemented.

The **Foundations 2** core asset team is responsible for coordinating implementation of this feature and submitting the Feature Request CR if one hasn't already been submitted.

2.2.2.1.Firmware Architecture

2.2.2.1.1.nvcfg - NVSRAM configuration

A new NVSRAM variable will be required by the DVC component to update when the driveReady events will need to be notified to the upper layer p-cache components. The variable will be defined as:

- dvcDriveReadyEventCache

This will be a Boolean variable. It will be initialized to 0 by default. The default value will indicate that the drive ready events will not be cached, but propagated. If the value is set to 1, the drive ready events will be cached by the dvc component.

The existing BDVirt region in NVSRAM can be used to store this Boolean variable.

This variable will not be updated via download of NCF.

This variable will not be backed up to DACstore nor will it be peered.

2.2.2.1.2.VariationMgmt - Variation Management Tools

2.2.2.1.2.1.Gears Variable

The Hypervisor AAD Virtual Machine Management element introduces a Boolean variable **HW_OpenStorageArchitecture**. The components within this AAD element will use the same Boolean either as a fully key or half key as required.

2.2.2.1.2.2.Recipe Process

For the individual VM build, recipe changes will be required to map exported header files and also to import header files into the project repository. For the solution bundle environment, recipe changes will also be required to import VM binaries into the repository.

Assets that created export directories in the project will need corresponding directories in the product repository.

The following table specifies the generic recipe processing extensions required to support the [Domain0]

sofd component:

Component	Element Mapping in CM Synergy Project
sofd	Foundations2_domain0/sofd (user level source)

The following table specifies the generic recipe processing extensions required to support the [Domain0] sam component:

Component	Element Mapping in CM Synergy Project
sam	Foundations2_domain0/sam (user level source)

2.2.2.1.3.Product Analysis

2.2.2.1.3.1.Solution bundle header

The solution bundle will contain a bundle header in the format specified within the Firmware Architecture Specification for firmware download, document number 349-1005040. The following additional information will need to be part of the bundle header:

- The CRC information (MD5 checksum) for the solution bundle payload
- The solution bundle size
- OSA compatibility for the bundle

2.2.2.1.3.2.Build tools

Appropriate build tools will need to be developed so that the solution bundle can be generated. The solution bundle will consist of a payload which includes an archive of all the rpm packages for the VMs.

The deliverable for the development team to Firmware architecture will be the bootable image as specified in Figure 3. The firmware architecture team will be responsible for wrapping/packaging the bootable image files in rpm format.

The rpms for each individual VM will contain the necessary pre-install, install and post-install scripts.

The install scripts will be responsible for the following:

- CRC (MD5 checksum) validation for the package
- Locating the offset of the VBD for the particular VM within the iSATA flash drive
- Create a LVM backed partition for the virtual machine with appropriate label which will indicate the VM nomenclature
- Mount the partition such that it can be accessed by the currently installed Domain0
- Copy the bootable image file for the VM to the LVM backed partition by using the UNIX utility 'dd'

The post-install script for the Domain0 rpm will be responsible for:

- Changing the grub to boot from the newly installed firmware/software

2.2.2.1.3.3.rpm spec file

A rpm spec file will need to be written. This spec file will indicate the install scripts to be run, the setup commands to be run and specify where within the iSATA drives the respective rpm will be installed. The spec file will be a preamble to the install process for the individual virtual machines.

2.2.2.2.Foundations 1

2.2.2.2.1.Meldb - Major Event Log Database

2.2.2.2.1.1.Controller Firmware Rollback Start

The asset that owns this new MEL event is [IOVM] sofd, section 2.2.2.3.8.

MEL Data Name	MEL Data Content
Event Name	Controller Firmware Rollback Start
Event Group	Staged Online Firmware Download events
Event Priority	Informational
Log Group	System
Event Category	Failure
Event Component Type	None
Sense Key	None
ASC/ASCQ	None
Event Specific Data	Firmware version of new STAGED firmware

2.2.2.2.1.2.Controller Firmware Rollback Complete

The asset that owns this new MEL event is [IOVM] sofd, section 2.2.2.3.8.

MEL Data Name	MEL Data Content
Event Name	Controller Firmware Rollback Complete
Event Group	Staged Online Firmware Download events
Event Priority	Critical
Log Group	System
Event Category	Failure
Event Component Type	None
Sense Key	None
ASC/ASCQ	None
Event Specific Data	Firmware version of new STAGED firmware

2.2.2.2.1.3.Controller Firmware not ACS capable

The asset that owns this new MEL event is [IOVM] sofd, section 2.2.2.3.8.

MEL Data Name	MEL Data Content
Event Name	Controller Firmware not ACS capable
Event Group	Staged Online Firmware Download events
Event Priority	Critical
Log Group	System
Event Category	Failure
Event Component Type	None
Sense Key	None
ASC/ASCQ	None
Event Specific Data	Firmware version of current RUNNING firmware

2.2.2.2.1.4.Solution bundle restore complete

The asset that owns this new MEL event is [IOVM] sofd, section 2.2.2.3.8.

MEL Data Name	MEL Data Content
Event Name	Solution Bundle restore complete
Event Group	Staged Online Firmware Download events
Event Priority	Informational
Log Group	System
Event Category	Failure
Event Component Type	None
Sense Key	None
ASC/ASCQ	None
Event Specific Data	Firmware version of current RUNNING firmware

2.2.2.2.2.SYMbol API

The SYMbol API changes related to System Firmware Upgrade are summarized in the Serviceability AAD, System Firmware Upgrade element, document number 45971-00. Details of these changes are in or will be included in the SYMbol Specification – Internal Master, document number 349-1051890.

The SYMbol API is represented in an XML data structure, from which both the SYMbol specification and the public API file SYMbolAPI.x are generated. The SYMbolAPI asset does not directly contain this XML content, but rather it contains a generated SYMbolAPI.x file for each of the currently supported feature

sets on the RAIDCore trunk. The Gears api.target variable is used to select the appropriate SYMbolAPI.x file for the feature set associated with the product being built.

The variation mechanism used to determine the generated content of the feature-set-specific SYMbolAPI.x files is outside the scope of this FAM. However, this AAD element does specify the Gears variables defined in the feature model that allow the firmware assets to vary as needed to adapt to the contents of the feature-set-specific SYMbolAPI.x selected for the product being built. These variables are defined in section 2.2.2.1.2 and their use is noted throughout the FAM as appropriate.

The SYMbol API set will be extended to support the following new SYMbol APIs:

- getDownloadConfiguration

The *getDownloadConfiguration* SYMbol API will be invoked by the management client as the initial step for a firmware download or restore operation. This API will return the following information:

1. ssh authentication keys
2. ssh port number
3. full download path

This API will be serviced by the sam (Storage Array Manager) component. The management client will set up the ssh session with in Domain0 by using the above information.

- startOSAFirmwareDownload

The *startOSAFirmwareDownload* SYMbol API will be invoked by the management client prior to starting the actual download of the new solution bundle. This API will be serviced by the sofd component. This API will be used by the controller firmware to mark appropriate download states within the controller flash.

- OSAFirmwareDownloadComplete

The *OSAFirmwareDownloadComplete* SYMbol API will be invoked by the management client after the 'scp' operation to download the solution bundle has completed. This will notify the controller firmware that the actual download for the solution bundle is complete and the controller firmware can now mark its states and start peering the solution bundle to the Alternate controller. This API will be serviced by the sofd component.

- getSolutionBundlePeeringStatus

The *getSolutionBundlePeeringStatus* SYMbol API will be invoked by the management client to get the status of the peering operation of the solution bundle to the Alternate controller. The user call to download the solution bundle to the flash will return only after this call indicates a completion status to the client. This call will be serviced by the sofd component.

- activateOSAFirmware

The *activateOSAFirmware* SYMbol API will be invoked by the management client to activate the currently staged firmware solution bundle. This command will actually install the new firmware within the controller and then reboot the controller from the newly installed firmware. This command will be implemented by the sofd component.

- saveFirmwareBundleToHost

The *saveFirmwareBundleToHost* SYMbol API is invoked by the management client just prior to the actual download of the new solution bundle. This command will indicate to the controller firmware that the solution bundle for the currently running firmware solution bundle is being retrieved by the management host to a known location within the management host. This command will be serviced by the sofd component.

- saveFirmwareBundleToHostComplete

The saveFimwareBundleToHostComplete SYMbol API is invoked by the management client to indicate to the controller firmware that the currently running firmware solution bundle has been saved to a known location within the management host. The controller firmware will mark appropriate status upon completion. This API is serviced by the sofd component.

- restorePreviousOSAFirmware

The *restorePreviousOSAFirmware* will be invoked by the management client in case the newly staged firmware fails to activate. In such a scenario, the client will restore the previously running OSA solution bundle which was retrieved using the *saveFirmwareBundleToHost* command. This API can also be invoked if a spare controller running the same version of firmware as the incumbent is inserted to the array between the time window of a download and activation. In such a scenario, the staged firmware solution bundle will be discarded and a critical event logged. The recovery action will be to use the above command and restore the previously running firmware solution bundle for ACS purposes. This API will be serviced by the sofd component.

The following SYMbol APIs will not be used for OSA based firmware:

- loadControllerFirmware
- activateStagedControllerFirmware

2.2.2.3.Foundations 2 (Coordinating Asset Team)

2.2.2.3.1.[Domain0] sod - Start of Day

The Domain0 start-of-day sequence will start the sofd component after starting the cmgr component. The sofd component will reside within the Linux user space.

2.2.2.3.2.[Domain0] cmgr - Controller Manager

The cmgr component running in Domain0 will implement an API which can be invoked by using a RMI call from the other guest VMs. From the IOVM perspective, the csm component running within IOVM will invoke an API provided by [Domain0] cmgr to let it know that the IOVM SOD is complete after the firmware upgrade.

Once all guest VMs have notified [Domain0] cmgr that their Start-of-Day sequence is complete, the cmgr component will invoke a RMI call to indicate that the system SOD is complete and the respective VM can release IOs if necessary. From the IOVM perspective, the [Domain0] cmgr will invoke a RMI call to indicate to [IOVM] cmgr that system SOD is complete. The [Domain0] cmgr component will therefore, be a RMI client.

The cmgr changes for this feature will vary and will be managed by the Feature Model HW_OpenStorageArchitecture.

2.2.2.3.3.[IOVM] cmgr - Controller Manager

The [IOVM] cmgr will be enhanced to determine the instance of the first SOD after a firmware upgrade. The cmgr controller record has information about the current and expected firmware versions. The cmgr code also determines whether the controller is native or foreign by comparing the serial number with the alternate controller. Therefore, if the controller is native and the current and expect firmware versions do not match, then it can be considered the first reboot after upgrade. In such a scenario, the cmgr component will call an API provided by the csm component to indicate to csm that this is the first reboot

after a firmware upgrade and hence the controller SOD complete event need not be sent to interested listeners.

The cmgr component in IOVM will provide an API which the pstor component can call to determine the first SOD after a firmware upgrade. In such a scenario, the pstor component will initialize the pstor metadata region.

The [IOVM] cmgr will provide an API which [Domain0] cmgr can call to notify of the system SOD completion. The [IOVM] cmgr will therefore be a RMI client.

When [Domain0] cmgr notifies [IOVM] cmgr of the system SOD completion event, the cmgr component running on Domain0 invokes an API provided by the csm component to release the system SOD completion event to interested listeners. As part of the system SOD complete, the cmgr component will update the firmware solution version number to dbm.

The cmgr changes for this feature will vary and will be managed by the Feature Model HW_OpenStorageArchitecture.

2.2.2.3.4.[IOVM] csm - Controller Services Manager

The csm component wil provide an API which the [IOVM] cmgr component can call to indicate the first SOD after a firmware upgrade. The csm component will mark an in-memory data structure to identify this state. When the csm sod complete routine is invoked as part of IOVM SOD complete, it will first check with the in-memory data structure. If the data structure indicates that this is the first SOD after upgrade, the csm component will not send the utl event ControllerSODCompleteEvent. Instead it will invoke an API provided by the [Domain0] cmgr component (through RMI) to notify [Domian0] cmgr of IOVM SOD completion. Therefore, the csm component will be a RMI client.

The csm component will provide an API which the [IOVM] cmgr component can call to indicate it to release the system SOD complete event. A new event SystemSODCompleteEvent will be defined. When the csm API is invoked by the cmgr component, the SystemSODCompleteEvent will be notified to interested listeners. The potential listeners will be symrpc and sofd.

The csm changes for this feature will vary and will be managed by the Feature Model HW_OpenStorageArchitecture.

2.2.2.3.5.[Domain0] sam - Storage Array Manager

[Domain0] sam will be responsible for maintaining the download configuration data like the ssh authentication keys, the ssh port number on Domain0 and the full destination path on the Domain0 file system. The data will be maintained within the Domain0 file system within the metadata region so that this is retained across firmware upgrades. The data will be read off the metadata region whenever the SYMbol API invoked through IOVM reaches Domain0.

[Domain0] sam will be a client of RMI as it will communicate with [IOVM] sam using RMI.

The sam changes for this feature will vary and will be managed by the Feature Model HW_OpenStorageArchitecture.

2.2.2.3.6.[IOVM] sam - Storage Array Manager

The [IOVM] sam component will implement the handler for the following SYMbol API:

- getDownloadConfiguration

It will need to register the handler for the above API with SYMbol. The above API will be invoked to get

the download configuration data

The sam component running on IOVM will coordinate with the sam component running on Domain0 to fetch the configuration data. It will invoke a RMI call to fetch the required data from [Domain0] sam. Therefore, the same component will be a RMI client.

The sam changes for this feature will vary and will be managed by the Feature Model HW_OpenStorageArchitecture.

2.2.2.3.7.[Domain0] sofd - Staged Online Firmware Download

The [Domain0] sofd will perform the main functions of the download and activation. The [Domian0] sofd will be responsible for marking the different states in the Domain0 file system about the download and the activation process. [Domain0] sofd will coordinate with [IOVM] sofd regarding the retrieval of the solution bundle to host, downloading the new solution bundle and also restore the previous solution bundle in case of an activation failure. [Domain0] sofd will therefore be a RMI client.

Once the new solution bundle has been downloaded to the STAGING area, the management client will invoke the OSA firmware download complete SYMbol API. As soon as the call reaches [IOVM] sofd, it notifies [Domain0] sofd that the download is complete within the primary controller. [Domain0] sofd will then initiate the transfer of the solution bundle to the alternate controller using a ssh connection with Domain0 of the alternate controller. It will monitor the status of the transfer to the alternate controller. Whenever the client invokes the Get Solution Bundle Peering Status SYMbol call, [IOVM] sofd communicated with [Domain0] sofd to return the transfer status to the management client. The call indicates transfer completion when the 'scp' operation to the alternate controller is complete.

[Domain0] sofd will be responsible for the actual install process of the new firmware. It will first initiate claiming of the cache offload device by communicating with the xsmgr component. Once the cache offload device has been claimed, it will move the solution bundle from the STAGING area to the cache offload area. Further to that it will perform all steps as identified in section 2.2.1.1.7.1, section 2.2.1.1.7.2 and section 2.2.1.1.7.3.

[Domain0] sofd will be responsible for managing the ACTIVE and the STAGING area within the iSATA flash drive for the firmware images. It will mark the regions appropriately during the activation phase.

As part of the controller reboot during activation, Domain0 will first boot off the new firmware. It will then attempt to install the other VMs and then launch the VMs. The sofd component running in Domain0 will need to determine the Domain0 boot after the upgrade. Therefore, all activation states will be marked within the Domain Linux file system within the metadata region as identified in the section mentioned. This region will be mounted by the active Domain0 and the metadata will be read from this partition.

As part of system SOD complete, the cmgr component running within Domain0 will notify sofd that the system SOD phase is complete. The sofd component will then move the new solution bundle to the new STAGING area. It will then invoke a RMI call to [IOVM] sofd to notify the p-cache components to reclaim the cache offload device as identified in section 2.2.1.3.6.

[Domain0] sofd will wait for a specified timeout period for system SOD to complete. If system SOD does not complete within that specified period, the sofd component will initiate the rollback to the previous running version of the firmware. It will initiate the rollback one controller at a time. It will perform all the steps as identified in section 2.2.1.1.7.4. It will coordinate with [IOVM] sofd using RMI calls for reclaim of the cache offload device (as stated above) and also for event notification when the activation fails.

The sofd changes for this feature will vary and will be managed by the Feature Model HW_OpenStorageArchitecture.

Architecture Note: The timeout period for system SOD can be set as 10 minutes to start with. This may

need to be tuned further. The timeout starts from the time Domain0 on the primary controller boots off the new firmware and completes its initialization process till the point where the guest VMs on the primary controller are also launched with the new firmware.

2.2.2.3.8.[IOVM] sofd - Staged Online Firmware Download

The sofd component running within IOVM will need to implement the handlers for the following SYMbol APIs:

- startOSAFirmwareDownload
- OSAFirmwareDownloadComplete
- getSolutionBundlePeeringStatus
- activateOSAFirmware
- saveFirmwareBundleToHost
- saveFirmwareBundleToHostComplete
- restorePreviousOSAFirmware

It will need to register the handler for the above APIs with SYMbol. These APIs would be invoked at various stages during the firmware download and activation operations.

As part of the download process, the SYMbol commands will reach [IOVM] sofd. However, the actual download operations will be performed by Domain0 as part of the 'scp' operation. However, [IOVM] sofd will invoke RMI calls to communicate with [Domain0] sofd so that [Domain0] sofd can maintain the download states at various stages of the download. Therefore, [IOVM] sofd will be a RMI client.

As part of the activation step, the sofd component will call an API provided by the dvc component to indicate to the dvc component that it will initiate claiming the cache offload device for activation. It will then communicate through RMI to indicate to [Domain0] sofd to close the back-end connection for the cache offload device.

The sofd component running within IOVM will use the services of the rph component (by registering with rph) to ensure that all active SYMbol commands which can be used to change array configuration are not allowed during activation.

[IOVM] sofd will also call an API provided by the ccm component to disable write caching for all volumes within the array. This will be done prior to claiming the cache backup device for activation.

Once the above 3 operations are performed the sofd component will proceed to transfer the volumes, take the SYMbol lock and then indicate to [Domain0] sofd via RMI to proceed with the actual installation and activation steps during the activation phase.

[IOVM] sofd will also register for the new System SOD Complete Event which is emitted by the csm component. As part of the system SOD complete event, it understands that IOVM has completed all SOD completed notifications. Further to this, when [Domain0] sofd indicates that via RMI that it has received notification from [Domain0] cmgr on system SOD completion, the sofd component running within IOVM calls and API provided by the dvc component so that dvc can release the *driveReady* events for the cache offload device to the upper layer components.

The sofd changes for this feature will vary and will be managed by the Feature Model HW_OpenStorageArchitecture.

2.2.2.4.Hypervisor

2.2.2.4.1.xsmgr - Xenstore Manager

The xsmgr component will provide an API which the sofd component running on Domain0 can call to

initiate closing the cache offload device backend connection by marking the state as *XenbusStateClosing*. This step is necessary to send the appropriate event to the xbfd driver running on IOVM.

2.2.2.5.IO Interfaces 1

2.2.2.5.1.xbfd - Xenbus block front-end driver

The xbfd driver running on IOVM will receive an event when the back-end connection for the cache offload device is closed. The xbfd driver will need to implement an interface that the cache backup device is no longer accessible. This will result in a ITN down event to ioni which will be propagated to the upper layers (interested components).

2.2.2.6.Platforms

2.2.2.6.1.BCM - Board Configuration Module

The BCM component will need to provide get/set routines for the NVSRAM variable defined in section 2.2.2.1.1.

2.2.2.7.Volume IO Services

2.2.2.7.1.dvc - Drive Virtualization Component

The dvc component will need to provide an API which will be called by the sofd component prior to claiming the cache offload device for activation. When this API is called, the dvc component will store the drive ready event caching variable in NVSRAM. If this variable is set, the dvc component will not propagate the drive ready events for the cache offload device to the upper layers. Instead, it will cache the drive ready events. When the NVSRAM variable is reset, it will again propagate the drive ready events to the upper layers.

The dvc component implements the drive removal implement. This method will need to propagate the appropriate drive removal event with the appropriate values to the upper layers,, in this case the pstor component.

The dvc implementation for caching the drive ready events will vary and will be managed by the Feature Model HW_OpenStorageArchitecture.

2.2.2.7.2.pbm - Persistent Backup Manager

The pbm component implements the drive removal event. It needs to ensure that a drive removal events translates into the cache backup device unavailable event for ccm and ncb so that ccm can disable write caching for all volumes.

2.2.2.7.3.pstor - Persistent Store

The pstor component implements the drive removal implement. This method will need to propagate the appropriate drive removal event with the appropriate values to the upper layers, in this case the pbm component. All of the pstor cache will need to be disabled when the disabled when all of the devices are removed (by closing the back-end connection from Domain0.

When the drive ready events are propagated the first time after a firmware upgrade, the pstor component will need to ensure that the pstor metadata region is re-initialized. The pstor component will call an API

provided by the cmgr component to determine that it is the first reboot after a firmware upgrade. In such a scenario, the pstor component will initialize the pstor metadata region.

The pstor implementation to initialize the metadata region during the first reboot after a firmware upgrade will vary and will be managed by the Feature Model HW_OpenStorageArchitecture.