HP Universal CMDB

for the Windows and Solaris operating systems

Software Version: 8.01

HP Universal CMDB–Storage Essentials (SE) Integration Guide

Document Release Date: March 2009
Software Release Date: March 2009

Legal Notices

Warranty

The only warranties for HP products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. HP shall not be liable for technical or editorial errors or omissions contained herein.

The information contained herein is subject to change without notice.

Restricted Rights Legend

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

Third-Party Web Sites

HP provides links to external third-party Web sites to help you find supplemental information. Site content and availability may change without notice. HP makes no representations or warranties whatsoever as to site content or availability.

Copyright Notices

Trademark Notices

Adobe® and Acrobat® are trademarks of Adobe Systems Incorporated.

Intel®, Pentium®, and Intel® Xeon™ are trademarks of Intel Corporation in the U.S. and other countries.

Java™ is a US trademark of Sun Microsystems, Inc.

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

Oracle® is a registered US trademark of Oracle Corporation, Redwood City, California.

Unix® is a registered trademark of The Open Group.

Documentation Updates

This guide's title page contains the following identifying information:

- Software Version number, which indicates the software version.
- Document Release Date, which changes each time the document is updated.
- Software Release Date, which indicates the release date of this version of the software.

To check for recent updates, or to verify that you are using the most recent edition of a document, go to:

http://h20230.www2.hp.com/selfsolve/manuals

This site requires that you register for an HP Passport and sign-in. To register for an HP Passport ID, go to:

http://h20229.www2.hp.com/passport-registration.html

Or click the **New users - please register** link on the HP Passport login page.

You will also receive updated or new editions if you subscribe to the appropriate product support service. Contact your HP sales representative for details.

Support

You can visit the HP Software Support web site at:

http://www.hp.com/go/hpsoftwaresupport

This web site provides contact information and details about the products, services, and support that HP Software offers.

HP Software Support Online provides customer self-solve capabilities. It provides a fast and efficient way to access interactive technical support tools needed to manage your business. As a valued support customer, you can benefit by using the HP Software Support web site to:

- Search for knowledge documents of interest
- Submit and track support cases and enhancement requests
- Download software patches
- Manage support contracts
- Look up HP support contacts
- Review information about available services
- Enter into discussions with other software customers
- Research and register for software training

Most of the support areas require that you register as an HP Passport user and sign in. Many also require a support contract.

To find more information about access levels, go to:

http://h20230.www2.hp.com/new_access_levels.jsp

To register for an HP Passport ID, go to:

http://h20229.www2.hp.com/passport-registration.html

Table of Contents

1

Storage Essentials (SE) Integration

This chapter includes the main concepts, tasks, and reference information for SE integration with HP Universal CMDB (UCMDB).

This chapter includes:

Concepts

Tasks

Overview

This chapter describes how to integrate SE with UCMDB. Integration involves synchronizing devices, topology, and hierarchy of a customers' storage infrastructure in the UCMDB database (CMDB). This enables Change Management and Impact Analysis across all business services mapped in UCMDB from a storage point of view.

Integration involves a UCMDB initiated discovery on the SE Oracle database. Synchronized Configuration Items (CIs) include storage arrays, fiber channel switches, hosts (servers), storage fabrics, logical volumes, host bus adapters, storage controllers, and fiber channel ports. The integration also synchronizes physical relationships between various hardware and logical relationships between logical volumes, storage zones, storage fabrics, and hardware devices to enable end-to-end mapping of the storage infrastructure in UCMDB.

Supported Versions

This package was developed and tested on HP Universal UCMDB versions 7.0 and 8.0 with SE version 6.x.

Prerequisites

The minimum VM installation requirements for SE integration are:

- 4 GB memory.
- 50 GB hard drive space.

CI Type Model

The following new Configuration Item (CI) Types in the UCMDB CI Type model represent storage entities in UCMDB:

- **storagearray**. (Storage Array) This CI Type represents a Storage Array with change monitoring on details such as serial number, version, and status. Since a storage array may not have a discoverable IP address, it inherits from the Chassis CI Type under network resources.

- **fcswitch**. (Fiber Channel Switch) A switch falls under the Host CI Type since SE maintains an IP address for each switch. Parameters such as status, state, total/free/available ports, and version are change monitored.
- **fcport**. (Fiber Channel Port) This CI Type has change monitoring enabled on parameters such as state, status, WWN, and trunked state. Since a Fiber Channel Port is a physical port on a switch, it inherits from the Physical Port CI Type under network resources.
- **fchba**. (Fiber Channel HBA) This CI Type has change monitoring enabled on parameters such as state, status, version, firmware version, driver version, WWN, and serial number. A Fiber Channel HBA inherits from the Host Resource CI Type.
- **storageprocessor**. (Storage Processor) This CI Type represents other storage devices such as SCSI controllers, and inherits from the HOST RESOURCE CI Type. A Storage Processor has change monitoring on parameters such as state, status, version, WWN, roles, power management, and serial number.
- **logicalvolume**. (Logical Volume) This CI Type represents volumes on Storage Arrays and hosts with change monitoring on availability, total/free/available space, and storage capabilities.
- **storagefabric**. (Storage Fabric) This CI Type inherits from Network Resource CI Type and represents a storage fabric. This CI Type has no change monitoring enabled.
- **Fiber Channel Connect**. This CI Type represents a fiber channel connection between fiber channel ports.

How It Works

The discovery job runs SQL queries against Oracle materialized views installed and maintained by SE in the Oracle database.

The UCMDB discovery job in this package uses a database CI as the trigger. The job can be scheduled to run. For details, see "Discovery Scheduler Dialog Box" in *Discovery and Dependency Mapping Guide*.

SQL queries executed by the discovery job retrieve detailed information to build CIs and populate UCMDB.

Fiber Channel Switch Details

This package retrieves Fiber Channel Switch details from the mvc_switchsummaryvw and mvc_switchconfigvw views. The discovery retrieves detailed information about Fiber Channel Ports on each switch from the mvc_portsummaryvw view.

A switch inherits from a Host CI Type in UCMDB. Since UCMDB uses the IP address of a host as part of its primary key, this discovery pattern attempts to use an IP address from SE for this purpose. If an IP address is not available, the discovery job attempts to resolve the switch's IP address using a DNS name (also maintained by SE). If neither an IP address nor a DNS name is available, the switch is discarded.

The query below also retrieves details on Storage Domains and Storage Fabrics.

```
SELECT switch.switchid, switch.switchname, switch.cimdomainid, switch.vendor,
switch.description, switch.appiq_last_contacted, switch.ip, switch.dns, switch.wwn,
switch.model, switch.serialnumber, switch.version, switch.switchstatus,
switch.switchstate, switch.switchrole, switch.fabricid, switch.fabricwwn,
switch.fabricname FROM appiq_system.mvc_switchsummaryvw switch WHERE
switch.status<>8
```

```
SELECT switch.availableports, switch.connectedports, switch.totalports FROM
appiq_system.mvc_swtichconfigvw switch WHERE switch.switchid = switchID from
above query
```

```
SELECT port.portid, port.portname, port.domainid, port.description, port.wwn,
port.connected_to_wwn, port.portstate, port.portstatus, port.port_speed,
port.max_speed, port.portnumber, port.scsiport, port.port_symbolic_name,
port.porttype, port.link_technology, port.trunkedstate FROM
appiq_system.mvc_portsummaryvw port WHERE port.containerid = switchID from
above query
```

Results from these queries populate a map as shown below:

Storage Array Details

This package retrieves Storage Array details from the mvc_storagesystemsummaryvw view. The discovery pattern retrieves detailed information on Storage Processors and HBAs from the mvc_storageprocessorsummaryvw and mvc_cardsummaryvw tables respectively.

The SE database may possibly not be able to obtain IP address information on Storage Arrays for a variety of technical and policy related reasons. Since a Storage Array is a host as far as UCMDB is concerned, the discovery assumes that the serial number of a Storage Array is unique and uses this as the primary key. The CI is then manually set as a complete host. If the serial number of a Storage Array is not available, the array is discarded.

Since Fiber Channel Ports may be present on a Storage Array, Storage Processor, or HBA, the discovery pattern uses three separate queries to retrieve Fiber Channel Ports for each Storage Array. Detailed information about Fiber Channel Ports on each array are retrieved from the mvc_portsummaryvw view. Since this view uses a container ID as the key, the discovery pattern queries the view by container ID for each Storage Array, each Storage Processor on a Storage Array, and each HBA on a Storage Array.

Finally, the discovery pattern retrieves detailed information about Logical Volumes on each Storage Array from the mvc_storagevolumesummaryvw view.

```
SELECT array.storagesystemid, array.storagesystemname, array.domainid,
array.vendor, array.description, array.ip, array.model, array.serialnumber, array.version,
array.storagesystemstatus, array.provider_tag FROM
appiq_system.mvc_storagesystemsummaryvw array WHERE array.status<>8
```

SELECT storageProcessor.systemprocessorid, storageProcessor.systemprocessorname, storageProcessor.domainid, storageProcessor.vendor, storageProcessor.description, storageProcessor.ip, storageProcessor.dns, storageProcessor.wwn, storageProcessor.model, storageProcessor.powermanagement, storageProcessor.serialnumber, storageProcessor.version, storageProcessor.processorstatus, storageProcessor.resetcapability, storageProcessor.roles, storageProcessor.providertag FROM appiq_system.mvc_storageprocessorsummaryvw storageProcessor WHERE storageProcessor.status<>8 AND storageProcessor.containerid = storageArrayID from above query

SELECT hba.cardid, hba.cardname, hba.cardtype, hba.domainid, hba.vendor, hba.description, hba.wwn, hba.model, hba.serialnumber, hba.version, hba.firmware, hba.driverversion FROM appiq_system.mvc_cardsummaryvw hba WHERE hba.status<>8 AND hba.containerid = storageArrayID from above query

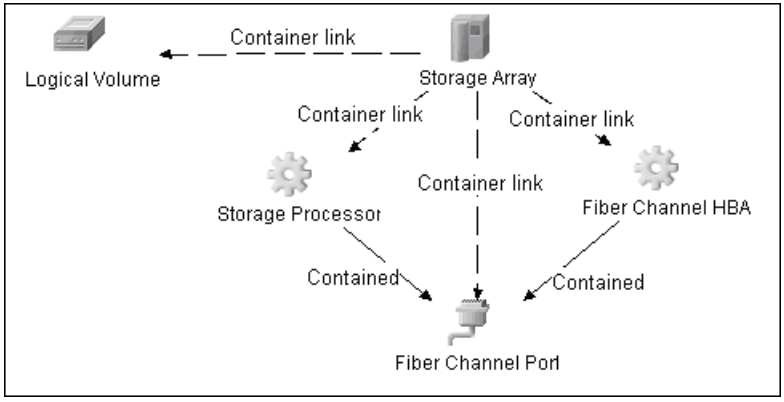
SELECT port.portid, port.portname, port.domainid, port.description, port.wwn, port.connected_to_wwn, port.portstate, port.portstatus, port.port_speed, port.max_speed, port.portnumber, port.scsiport, port.port_symbolic_name, port.porttype, port.link_technology, port.trunkedstate, port.containerid FROM appiq_system.mvc_portsummaryvw port WHERE port.status<>8 AND port.containerid = storageArrayID from above query

SELECT port.portid, port.portname, port.domainid, port.description, port.wwn, port.connected_to_wwn, port.portstate, port.portstatus, port.port_speed, port.max_speed, port.portnumber, port.scsiport, port.port_symbolic_name, port.porttype, port.link_technology, port.trunkedstate, port.containerid FROM appiq_system.mvc_portsummaryvw port WHERE port.status<>8 AND port.containerid IN (SELECT hba.cardid FROM appiq_system.mvc_storagesystemsummaryvw stor, appiq_system.mvc_cardsummaryvw hba WHERE hba.containerid = storageArrayID from above query)

```
SELECT port.portid, port.portname, port.domainid, port.description, port.wwn,
port.connected_to_wwn, port.portstate, port.portstatus, port.port_speed,
port.max_speed, port.portnumber, port.scsiport, port.port_symbolic_name,
port.porttype, port.link_technology, port.trunkedstate, port.containerid FROM
appiq_system.mvc_portsummaryvw port WHERE port.status<>8 AND port.containerid
IN (SELECT storageprocessor.systemprocessorid FROM
appiq_system.mvc_storagesystemsummaryvw stor,
appiq_system.mvc_storageprocessorsummaryvw storageprocessor WHERE
storageprocessor.containerid = storageArrayID from above query)
```

```
SELECT logicalVolume.storagevolumeid, logicalVolume.storagevolumename,
logicalVolume.domainid, logicalVolume.accesstype, logicalVolume.availability,
logicalVolume.statusinfo FROM appiq_system.mvc_storagevolumesummaryvw
logicalVolume WHERE logicalVolume.status<>8 AND logicalVolume.storagesystemid =
storageArrayID from above query
```

Results from these queries populate a map as shown below:

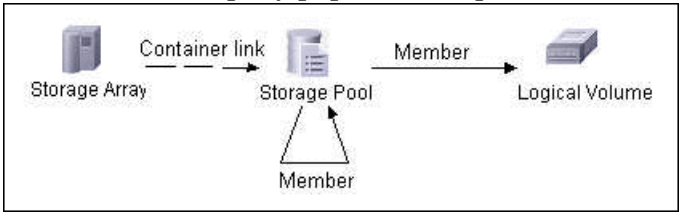


Storage Pool Details

Storage Pool information is also collected from each Storage Array using the query below.

```
SELECT pool.storagepoolid, pool.storagepoolname, pool.storagepooldescription,
pool.parentpoolid, pool.cimpoolid, pool.pooltype, pool.storagecapabilityname,
pool.nosingleptoffailure, pool.defaultnosingleptoffailure, pool.mindataredundancy,
pool.maxdataredundancy, pool.minspindleredundancy, pool.maxspindleredundancy,
pool.default_spindle_redundancy, pool.storagecapabilitycommonname,
pool.storagecapabilitydescription, poolConfig.capacitytype, poolConfig.capacitynum,
poolConfig.exportedmb, poolConfig.unexportedmb, poolConfig.availablemb,
poolConfig.provisionedmb, poolConfig.totalmb FROM
appiq_system.mvc_storgaepoolsummaryvw pool,
appiq_system.mvc_storagepoolconfigvw poolConfig WHERE pool.status <> 8 AND
pool.storagesystemid = storageArrayID from above query AND pool.storagepoolid =
poolConfig.storagepoolid AND poolConfig.collectiontime IN (SELECT
MAX(collectiontime) FROM appiq_system.mvc_storagepoolconfigvw) ORDER BY
pool.parentpoolid DESC
```

Results from this query populate a map as shown below:



Host Details

This package retrieves Host details from the mvc_hostsummaryvw view. The discovery pattern retrieves detailed information on HBAs from the mvc_cardsummaryvw view.

SE maintains information on Operating Systems, CPUs, Memory, IP address, and DNS name on each host. UCMDB uses this information to create host CIs of type UNIX or Windows, and adds CPU and Memory CIs for each host as available.

Since UCMDB uses the IP address of a host as part of its primary key, this discovery pattern attempts to use the IP address from SE for this purpose. If an IP address is not available, the pattern then attempts to resolve the hosts IP address using a DNS name. If neither an IP address nor a DNS name is available, the discovery pattern ignores the host.

Similar to Storage Arrays, a host may have Fiber Channel Ports directly associated with itself or on HBAs on the host. The discovery pattern uses three separate queries to retrieve Fiber Channel Ports for each host. The job retrieves detailed information about Fiber Channel Ports on each host from the mvc_portsummaryvw view. Since this view uses a ContainerID as the key, the discovery pattern queries the view by container ID for each host, and each HBA on a host.

Finally, the discovery pattern retrieves detailed information about Logical Volumes on each host from the mvc_hostvolumesummaryvw and mvc_hostcapacityvw view. The mvc_hostcapacityvw view maintains capacity information for each volume over multiple instances in time, and the discovery pattern uses only the latest available information.

```
SELECT host.hostid, host.hostname, host.domainid, host.vendor, host.description,
host.ip, host.dns, host.model, host.version, host.os, host.totalphysicalmem,
host.numberprocessor FROM appiq_system.mvc_hostsummaryvw host WHERE
host.status<>8
```

```
SELECT hba.cardid, hba.cardname, hba.cardtype, hba.domainid, hba.vendor,
hba.description, hba.wwn, hba.model, hba.serialnumber, hba.version, hba.firmware,
hba.driverversion FROM appiq_system.mvc_cardsummaryvw hba WHERE
hba.status<>8 AND hba.containerid = hostID from above query
```

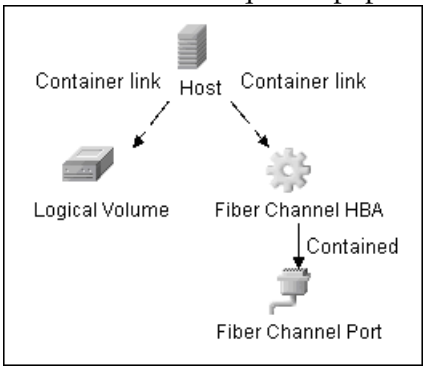
```
SELECT port.portid, port.portname, port.domainid, port.description, port.wwn,
port.connected_to_wwn, port.portstate, port.portstatus, port.port_speed,
port.max_speed, port.portnumber, port.scsiport, port.port_symbolic_name,
port.porttype, port.link_technology, port.trunkedstate, port.containerid FROM
appiq_system.mvc_portsummaryvw port WHERE port.status<>8 AND port.containerid
= hostID from above query
```

SELECT port.portid, port.portname, port.domainid, port.description, port.wwn, port.connected_to_wwn, port.portstate, port.portstatus, port.port_speed, port.max_speed, port.portnumber, port.scsiport, port.port_symbolic_name, port.porttype, port.link_technology, port.trunkedstate, port.containerid FROM appiq_system.mvc_portsummaryvw port WHERE port.status<>8 AND port.containerid IN (SELECT hba.cardid FROM appiq_system.mvc_storagesystemsummaryvw stor, appiq_system.mvc_cardsummaryvw hba WHERE hba.containerid = hostID from above query)

SELECT logicalVolume.logicalvolumeid, logicalVolume.logicalvolumename, logicalVolume.domainid, logicalVolume.description, logicalVolume.deviceid, logicalVolume.filesystemtype, logicalVolume.share_name FROM appiq_system.mvc_hostvolumesummaryvw logicalVolume WHERE logicalVolume.status<>8 AND logicalVolume.hostid = hostID from above query

SELECT logicalVolume.total, logicalVolume.used, logicalVolume.free FROM appiq_system.mvc_hostcapacityvw logicalVolume WHERE LOWER(logicalVolume.capacitytype) = \'raw\' AND logicalVolume.volumeid = logicalVolumeID from above query AND logicalVolume.timestamp IN (SELECT MAX(lv.timestamp) FROM appiq_system.mvc_hostcapacityvw lv WHERE LOWER(lv.capacitytype) = \'raw\' AND lv.volumeid = logicalVolumeID from above query).

Results from these queries populate a map as shown below:

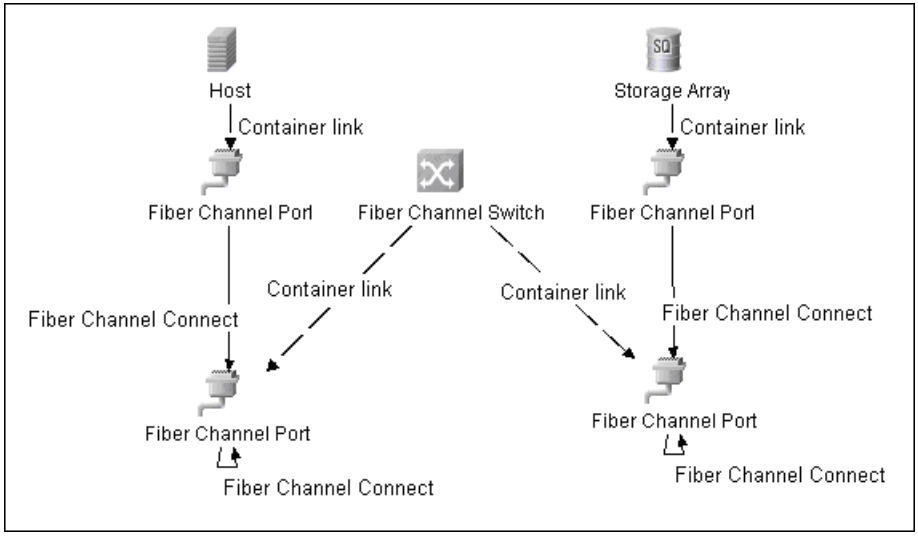


SAN Topology

SAN Topology consists of the Fiber Channel network topology and includes (fiber channel) connections between Fiber Channel Switches, Hosts, and Storage Arrays. SE maintains a list of WWN's each Fiber Channel Port connects to, and this package uses this list of WWN's to establish Fiber Channel Connection links.

```
SELECT port.wwn, port.connected_to_wwn FROM appiq_system.mvc_portsummaryvw
port WHERE port.wwn IS NOT NULL AND port.connected_to_wwn IS NOT NULL
```

Results from these queries populate a map as shown below:

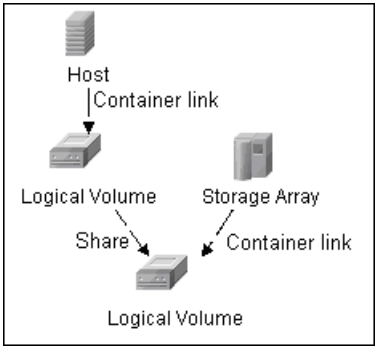


Storage Topology

Storage topology consists of relationships between Logical Volumes on a host and Logical Volumes on a Storage Array. This package uses multiple tables to identify this relationship as shown in the query below. This view is a simple summary of all of the above information.

```
SELECT DISTINCT hv.logicalvolumeid, sv.storagevolumeid FROM
appiq_system.mvc_hostsummaryvw h, appiq_system.mvc_pathvw ap,
appiq_system.mvc_subpathvw p, appiq_system.mvc_diskdrivesummaryvw ds,
appiq_system.mvc_hostvolumesummaryvw hv,
appiq_system.mvc_storgaepoolsummaryvw sp,
appiq_system.mvc_storagevolumesummaryvw sv,
appiq_system.mvc_storagevolumeports vp, appiq_system.mvc_protocolcontrollervw pc
WHERE ap.hostid=h.hostid AND ap.logicalvolumeid<>0 AND ap.ismountednum=1
AND p.pathid=ap.pathid AND p.diskdriveid=ds.diskdriveid AND
sv.storagevolumeid=p.storagevolumeid AND sp.storagepoolid=sv.poolid AND
sv.storagevolumeid=vp.storage_volume_id AND vp.port_id=pc.id AND
hv.logicalvolumeid=ap.logicalvolumeid
```

Deployment



The integration includes two UCMDB packages named Storage_Basic.zip and SE_Discovery.zip. The Storage_Basic.zip package contains the new CI Type definitions, views, reports, and correlation rules. This package is common to all Storage Management integration solutions.

The SE_Discovery.zip package contains the trigger TQL for SE Discovery, the discovery script, discovery pattern and the discovery job.

The UCMDB discovery job in this package uses the HP SE database CI as the trigger, and can be included in UCMDB's Spiral Discovery schedule. This discovery pattern has no parameters and requires no configuration.

To initiate discovery, activate DDM to discover the SE Oracle database using the following sequence of discovery patterns:

- IP address of the Oracle database server: Class C IPs by ICMP or by Range IPs (under Network – Basic).
- Oracle database server: Host Connection by SNMP or WMI or Shell (under Network – Basic).
- Oracle server instance: Oracle Connection by SQL (under Database – Oracle).
- Edit Discovery TQL (query) HPSE_OracleDB and set a node condition on the SID such that the TQL results only contain the HP SE database.
- Storage infrastructure: SE Integration by SQL (under Integration – Storage Essentials).

Note:

- For the Oracle server instance, it is recommended that you use the Oracle username **REPORT_USER**, since this user has privileges necessary to run SQL queries on the appiq_system tables.
- Since this discovery pattern queries Oracle Materialized Views, there is a chance that the views are being refreshed while the discovery job is running, resulting in the display of an error message.

Mapping

The integration synchronizes physical relationships between various hardware and logical relationships between logical volumes, storage fabrics, and hardware devices to enable end-to-end mapping of the storage infrastructure in UCMDB.

This section describes basic views, correlation rules, and reports.

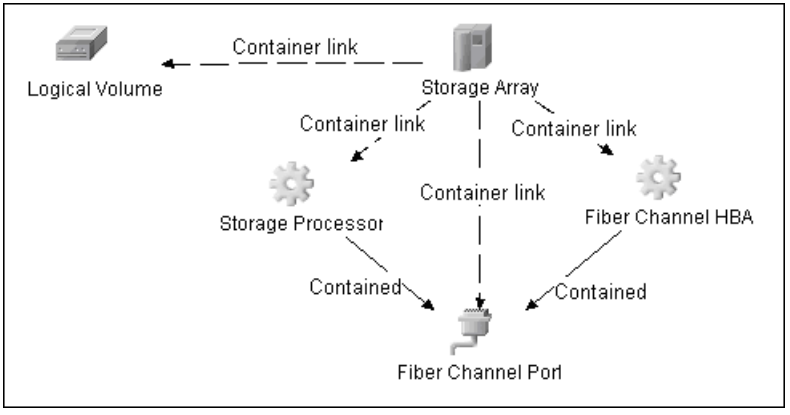
Views

This package contains views that display common storage topologies. These are basic views that can be customized to suit the integrated SE applications.

Storage Array Details

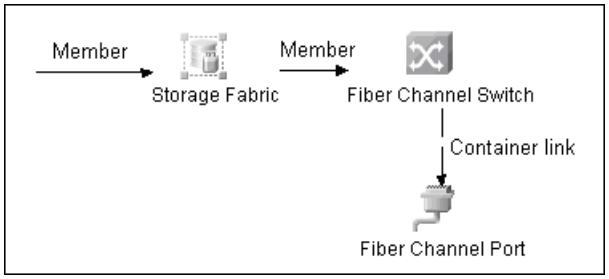
This view shows a Storage Array and all its components including Logical Volumes, HBAs, Storage Processors, and Fiber Channel Ports. The view shows each component under its container Storage Array and groups Logical Volumes by CI Type for clarity.

Storage Array does not require all components in this view to be functional. Container links stemming from the Storage Array have a cardinality of zero-to-many. The view may show Storage Arrays whether even where there are no Logical Volumes or Storage Processors.



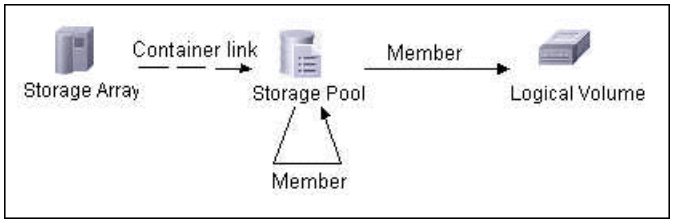
FC Switch Details

This view shows a Fiber Channel Switch and all connected Fiber Channel Ports. The view shows Fiber Channel Ports under its container Switch for clarity.



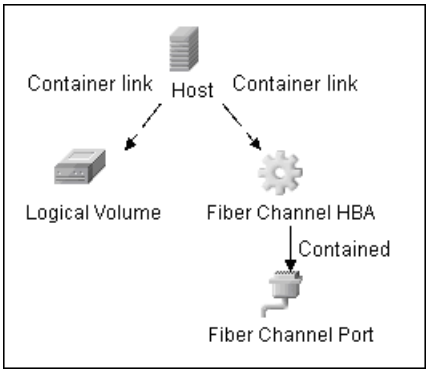
Storage Pool Details

This view shows Storage Pools with associated Storage Arrays and Logical Volumes.



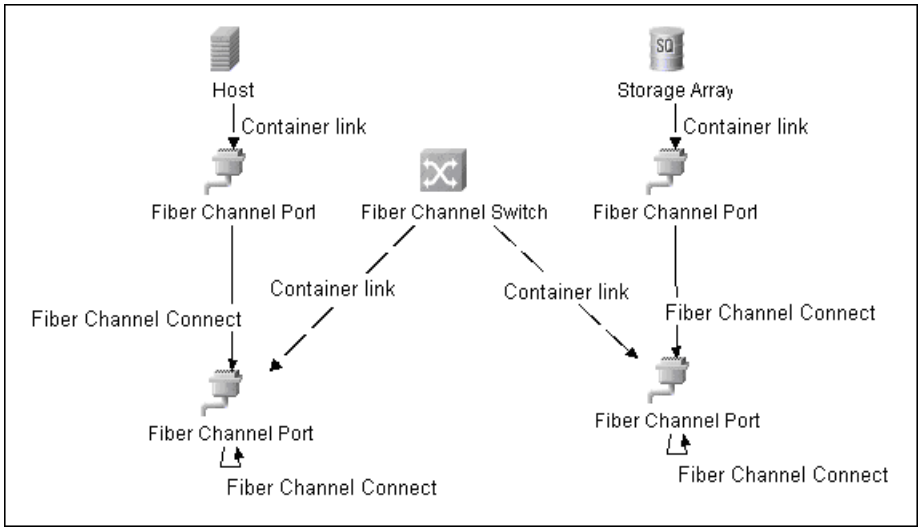
Host Details

This view shows only Hosts that contain a Fiber Channel HBA or a Logical Volume. This keeps this view storage-specific and prevents Hosts discovered by other UCMDB discovery jobs from appearing. The view shows Logical Volumes and Fiber Channel HBAs under its container Host and groups them by CI Type for clarity.



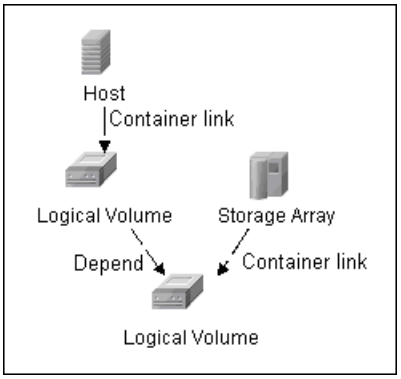
SAN Topology

This view maps physical connections between Storage Arrays, Fiber Channel Switches and Hosts. The view shows Fiber Channel Ports below their respective containers, and groups the Fiber Channel Connect relationship CI Type to prevent multiple relationships between the same nodes from appearing on the top layer.



Storage Topology

This view maps logical dependencies between Logical Volumes on Hosts and Logical Volumes on Storage Arrays. There is no folding in this view.



Correlation Rules

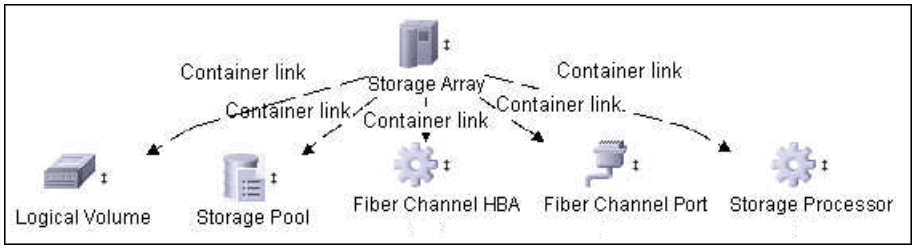
This package contains basic correlation rules to enable the Impact Analysis and Root Cause Analysis features in UCMDB. These correlation rules are templates for more complex rules that you can define based on business needs.

All correlation rules fully propagate both Change and Operation events. For details on Impact Analysis, see Chapter 6, “Correlation Manager Overview”.

Correlation events are not propagated to Fiber Channel Ports for performance reasons, since there can be many. For details, see “Storage Array Devices to Storage Array” on page 25, “Host Devices to Host” on page 25, “Logical Volume to Logical Volume” on page 26, and “FC Switch Devices to FC Switch” on page 26.

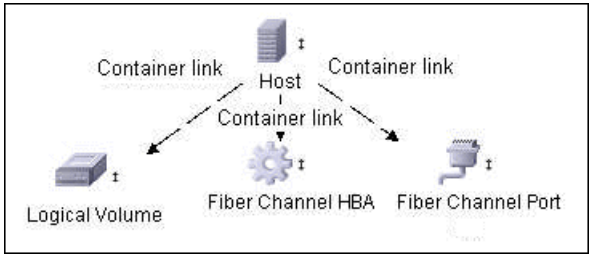
Storage Array Devices to Storage Array

This correlation rule propagates events between Logical Volumes, Storage Processors, Fiber Channel HBAs, and Storage Arrays.



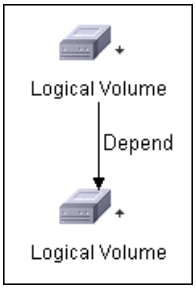
Host Devices to Host

This correlation rule propagates events between Fiber Channel HBAs, and Hosts, and Logical Volumes on the Host.



Logical Volume to Logical Volume

This correlation rule propagates events on a Logical Volume contained in a Storage Array to the dependent Logical Volume on the Host.

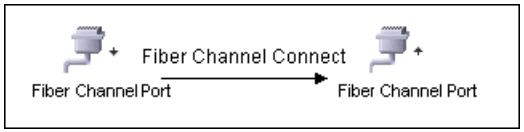


FC Switch Devices to FC Switch

This correlation rule propagates events from a Fiber Channel Port to and from a Switch. The event is propagated to the associated Storage Fabric as well.

FC Port to FC Port

This rule propagates events on a Fiber Channel Port to another connected Channel Port.



In an example scenario, consider an HBA on a Storage Array going down. The event propagates from the HBA to the Storage Array and the Logical Volumes on the Array because of the Storage Devices to Storage Array rule. The correlation event on the Logical Volume then propagates to other dependent Logical Volumes through the Logical Volume to Logical Volume rule. Hosts using those dependent Logical volumes see the event next because of the Host Devices to Host rule. Depending on business needs, you can define correlation rules to propagate events from these hosts to applications, Business Services, Lines of business, etc. This enables end-to-end mapping and impact analysis using UCMDB.

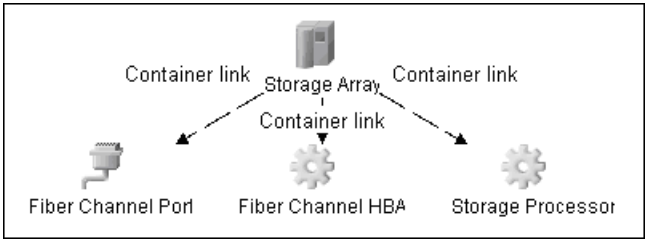
Reports

This package contains basic reports that can be customized to suit the integrated SE applications.

In addition to the system reports, Change Monitoring and Asset Data parameters are set on each CI Type in this package, to enable Change and Asset Reports in UCMDB. For details see “Storage Array Configuration” on page 28, “Host Configuration” on page 28, “Storage Array Dependency” on page 29, and “Host Storage Dependency” on page 29.

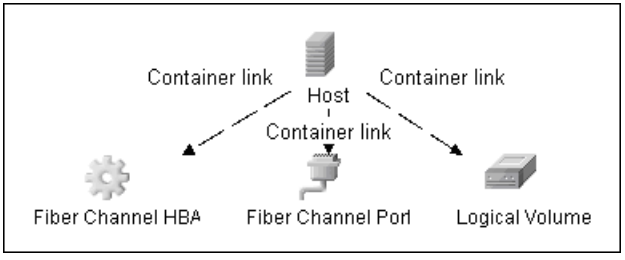
Storage Array Configuration

This report shows detailed information on Storage Arrays and its sub-components including Fiber Channel Ports, Fiber Channel Arrays, and Storage Processors. The report lists Storage Arrays with sub-components as children of the Array.



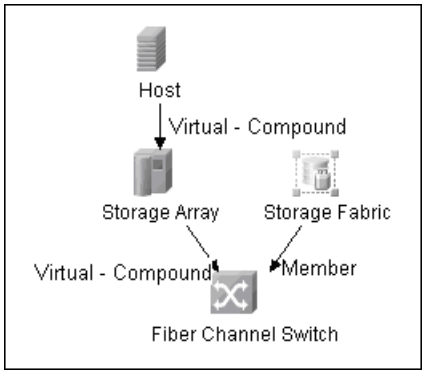
Host Configuration

This report shows detailed information on Hosts that contain one or more Fiber Channel HBAs, Fiber Channel Ports, or Logical volumes. The report lists Hosts with sub-components as children of the Host.



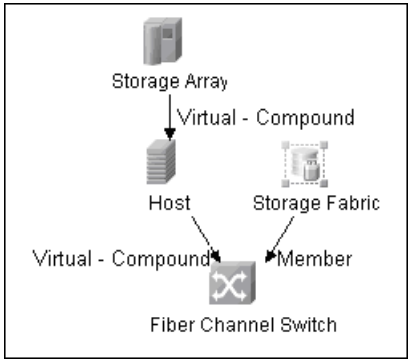
Storage Array Dependency

This report is an attempt to map dependencies on a Storage Array. The report also displays information on Switches connected to it.



Host Storage Dependency

This report shows detailed information on storage infrastructure dependencies of a Host. The report lists Hosts and dependent components.



Storage Pool Configuration

This report shows detailed information on Storage Pool configuration.