White Paper

Implementing EMC SRDF/Star Protected Composite Groups on Open Systems

Applied Technology

Abstract

This white paper documents the design, implementation, and operational considerations for multiple EMC® SRDF®/Star protected composite groups on open systems host platforms using Symmetrix DMX™ arrays with EMC Enginuity™ and EMC Solutions Enabler. This white paper also discusses testing 32 separate Star-protected composite groups in a single Symmetrix DMX array with a single set of Solutions Enabler hosts for management and control. As the Star testing involved several releases of Solutions Enabler (versions 6.4.3 and 6.5) and Enginuity (release 5772 and 5773), some notable differences are highlighted.

June 2009

Part Number h6369

Table of Contents

Executive summary

EMC® SRDF®/Star provides a three-site, disaster recovery (DR) solution. The open systems implementation of SRDF/Star provides an easy-to-use interface with relatively few commands to perform the functions of switching workload sites and managing the environment. Emphasis in this white paper has been placed on the design, implementation, and scalability considerations of SRDF/Star solutions for open systems. The initial design of the solution is critical to the ease of implementation of the solution. Detailed design documentation, while perhaps time-consuming to create, allows a very efficient implementation with reduced risk.

Implementing an SRDF/Star solution in an open systems environment is fairly straightforward. Once the prerequisites are met, setting up the Star environment is accomplished through the Solutions Enabler command symstar. The symstar command simplifies tasks when compared to performing underlying SRDF functions individually. In fact, there is no way to create or manage a Star environment without symstar, as some of the required capabilities for enabling Star are only available to the symstar command.

Scalability of environments is a topic of frequent discussion among open systems customers. While the traditional SRDF/S customer has enjoyed device-level granularity of control, Star solutions are based on composite groups. Recent enhancements to Solutions Enabler version 6.5 improved parallelism of Star operations when managing multiple composite groups concurrently. EMC Engineering continues to make advances in functionality, performance, and scalability.

Introduction to SRDF/Star

SRDF/Star is a data protection and failure recovery solution that covers three geographically dispersed data centers in a triangular topology. This architecture protects business data against a primary site failure or a regional disaster using Concurrent SRDF or Cascaded SRDF capability to mirror the same production data synchronously to one remote site and asynchronously to another remote site:

- The workload site is the primary data center where the production workload is running.
- The synchronous target site is a secondary site usually located in the same region as the workload site. The production data is mirrored to this site using synchronous replication.
- The asynchronous target site is a secondary site in a distant location. The production data is mirrored to this site using asynchronous replication.

In the event of a workload site failure, there is no data loss with the synchronous replication to the regional site. As a result, SRDF/Star provides several major benefits for failure recovery; the following are just a few:

- Users can quickly establish communication and protection between the two remaining remote sites, either of which can become the new workload site. SRDF/Star enables you to incrementally establish an asynchronous session between the two remaining remote sites, thus avoiding a full copy resynchronization and elongating the time it takes to re-enable disaster recovery protection. Incremental resynchronization (replicating only the data differences between the synchronous and asynchronous sites) dramatically reduces the time required to establish remote mirroring and protection for a new workload site following a primary site failure.

- Star allows the coordination of consistency groups to the two remaining remote sites, meaning that devices within a consistency group act in unison to preserve dependent-write consistency of a database that may be distributed across multiple RDF systems. SRDF/Star also enables users to determine which remote target site (synchronous or asynchronous) has the most current data in the event of a rolling disaster that affects the workload site. With a rolling disaster, there is no guarantee that the synchronous site will be more current than the asynchronous site. The capability to display where the most current data is located helps determine which site's data should be used for failure recovery.

Note: SRDF/Star relies on several key underlying technologies. Those technologies are outlined in the section "SRDF/Star supporting products and technology."

Introduction

This white paper examines various aspects of SRDF/Star protected composite groups on open systems. Sections are included on SRDF/Star supporting products and technology, consistency and control, planning and design, implementation, and operational considerations.

Audience

This white paper is intended for technology professionals, system architects, and IT administrators or technical staff interested in open systems SRDF/Star and its implementation and any operational considerations.

Terminology

The following terms are introduced and used in this white paper.

Term	**Description**
RDF-MSC and RDF-ECA	SRDF technologies where data is protected through enhanced SRDF Multi-Session Consistency (MSC) or SRDF Enginuity™ Consistency Assist (ECA) consistency group technology that monitors the data propagation from the source volumes to their corresponding target volumes.
Composite Group (CG)	A group comprised of SRDF devices enabled for remote database consistency that operate in unison to preserve the integrity and dependent write consistency of a database distributed across multiple arrays. Depending on the mode of operation, consistency is maintained with Enginuity Consistency Assist (for synchronous operations), or Multi-Session Consistency (for asynchronous operations), which respects the logical relationships between dependent I/Os.
SDDF Session	A mechanism within Symmetrix used to differentially track changes to volumes in the remote Symmetrix arrays configured in an SRDF/Star environment. This facility allows differential resynchronization between the remote Symmetrix arrays in the event of a loss of the primary workload site array.

SRDF/Star supporting products and technology

The following EMC products and SRDF technologies provide the foundation for consistency protection within an SRDF/Star configuration: SRDF/Synchronous (SRDF/S), SRDF/Asynchronous (SRDF/A), SRDF/Consistency Groups (SRDF/CG), Concurrent SRDF, Cascaded SRDF, and Dynamic SRDF.

SRDF/Synchronous (SRDF/S)

SRDF/S enables synchronous replication of data to maintain a real-time copy of the data on the secondary (R2) device at all times. SRDF/S replication to a nearby regional location (the SRDF/Star synchronous target site) provides consistency protection such that applications are notified that an I/O (or I/O chain) is complete when the remote Symmetrix® array acknowledges that the data has been secured in cache. Synchronous replication ensures 100 percent synchronized mirroring of the R1 devices to the synchronous target site. RDF-ECA, an SRDF technology, provides consistency protection for SYNC-mode devices by performing suspend operations across all SRDF/S devices in a consistency group.

SRDF/Asynchronous (SRDF/A)

SRDF/A is an SRDF replication product that enables users to asynchronously replicate data while maintaining a dependent-write consistent copy of the data on the secondary (R2) device at all times. The point-in-time copy of the data at the remote site is only slightly behind the workload (R1) site. SRDF/A session data is transferred to the asynchronous target site in predefined timed cycles or delta sets, eliminating the redundancy of multiple same-block or track changes being transferred over the link, potentially reducing the required bandwidth.

SRDF/A provides a long-distance replication solution with minimal impact on performance. In the event of a disaster at the workload site, or if RDF links are lost during data transfer, a partial delta set of data can be discarded, preserving consistency on the R2 with a data loss of no more than two SRDF/A cycles. Multi-Session Consistency (MSC) provides consistency protection by coordinating SRDF/A cycle switching and performing cache recovery operations across multiple SRDF/A sessions in a consistency group.

SRDF/Consistency Groups (SRDF/CG)

SRDF/CG ensures the consistency of the data remotely mirrored by the SRDF operations in the event of a rolling disaster. SRDF/CG prevents a rolling disaster from affecting the integrity of the data at the SRDF/Star remote sites. When SRDF/CG detects any write I/O to a volume that cannot communicate with its remote mirror, SRDF/CG suspends the remote mirroring for all volumes defined to the consistency group before completing the intercepted I/O and returning control to the application. In this way, SRDF/CG prevents dependent I/O from reaching its remote mirror in the case where a predecessor I/O only gets as far as the local mirror.

Concurrent SRDF

Concurrent SRDF allows the same source data to be copied concurrently to Symmetrix arrays at two remote locations. As Figure 1 shows, the capability of a concurrent R1 device to have one of its links synchronous and the other asynchronous is supported as an SRDF/Star topology. Additionally, SRDF/Star allows the reconfiguration between concurrent and cascaded modes dynamically.

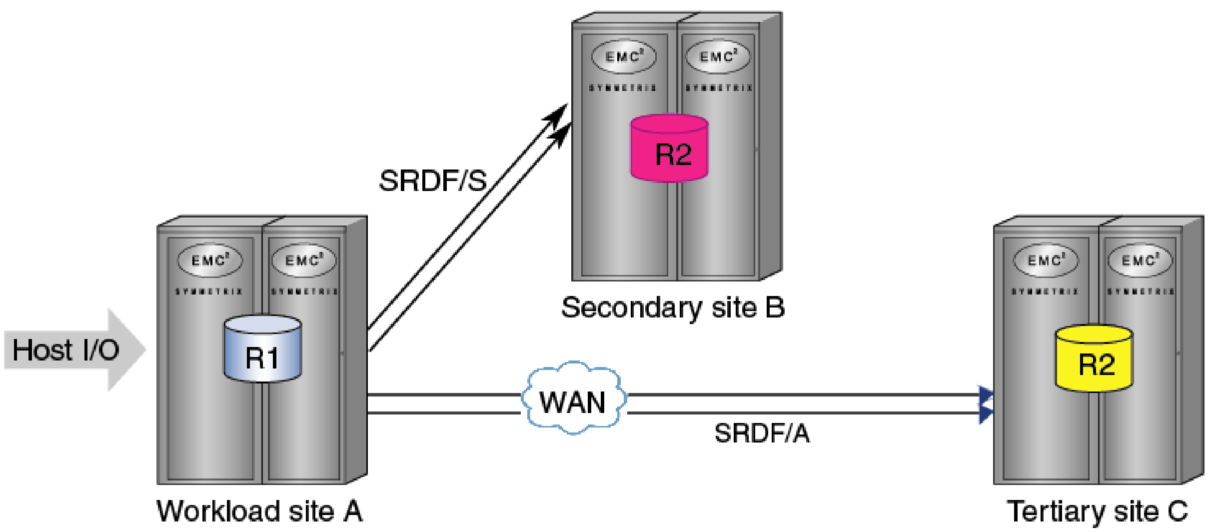

Figure 1. Concurrent SRDF configuration

Cascaded SRDF

Introduced with Enginuity 5773, Cascaded SRDF allows a device to be both a synchronous target (R2) and an asynchronous source (R1) creating an R21 device type. SRDF/Star supports the cascaded topology and allows the dynamic reconfiguration between cascaded and concurrent modes.

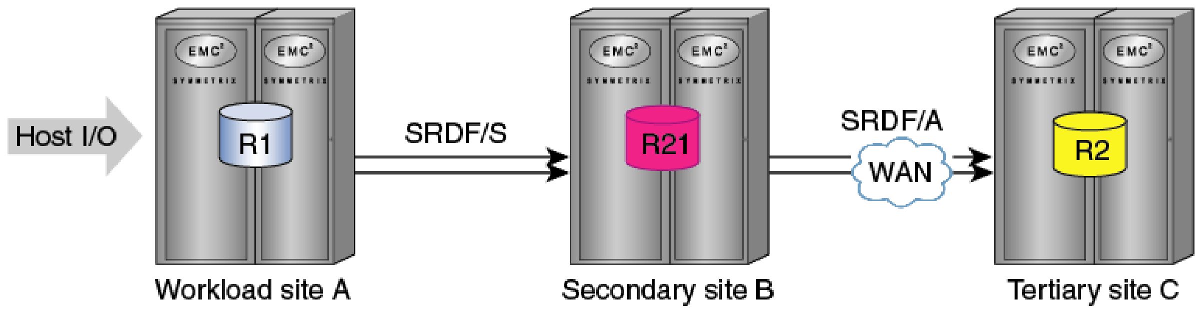

Figure 2. Cascaded SRDF

Dynamic SRDF

Dynamic SRDF is an SRDF configuration option that enables a Symmetrix device to be configured with the dual capability to function as an R1 or an R2 device. By setting a special attribute for the devices they can be easily converted to either an R1 for a source or R2 for target devices. Each Symmetrix array in the SRDF/Star configuration must use dynamic RDF devices. During failure recovery, the R2 devices at either the synchronous target site or the asynchronous target site are dynamically converted to R1 devices to function as the production devices at the new workload site.

SRDF/Star building blocks, components, and definitions

SRDF/Star is a combination of host and Symmetrix software working cooperatively to perform SRDF/A cycle switching, manage SDDF sessions, and perform management and operations for the Star environment. Figure 3 lists the server and Symmetrix DMX™ components needed for SRDF/Star.

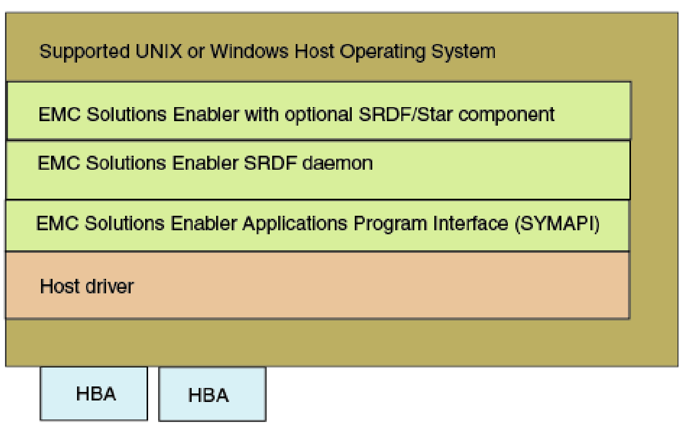

Figure 3. Host components for SRDF/Star

The user interface for SRDF/Star is the Solutions Enabler SYMCLI symstar command. As with any other Solutions Enabler software, the host communicates with the Symmetrix DMX through gatekeepers in the array. The management entity for SRDF/Star is the composite group and SRDF group name within the composite group. Management does not occur at the individual device level in the SRDF/Star environment.

Figure 4 lists the Solutions Enabler definitions for SRDF/Star.

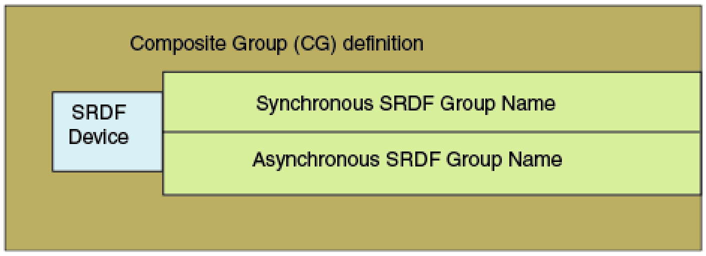

Figure 4. Solutions Enabler definitions for SRDF/Star

SRDF/Star consistency and control

SRDF/S customers have long enjoyed device-level granularity when performing SRDF control operations, such as establish, split, failover, and failback. This device-level operational control affords open systems applications independence with respect to replication. Enterprises are running hundreds of applications on a single Symmetrix DMX array without having to consider grouping applications for replication operations ─ these same customers often create a SYMCLI device group per application (or application-set) and control local (EMC TimeFinder®) and remote (SRDF) replication on the device group or "per application" level. Operational procedures for this paradigm have long been adopted by these enterprises.

With the advent of SRDF/A, the granularity and control of SRDF have shifted to the SRDF-group level rather than the individual device level within the Symmetrix. The number of SRDF groups supported on a single SRDF director is specific to the Enginuity release being deployed. Large-scale consolidated environments often place hundreds of servers and applications into a single Symmetrix array. Such enterprises wishing to deploy SRDF/A, either with or without Concurrent SRDF/S, are potentially faced with a reduced level of application independence. The creation of application-sets based on interdependent data across applications becomes a necessary and important design and implementation consideration for SRDF/A. These application-sets are organized into SRDF groups in SRDF/A implementations. These interrelated applications will replicate as a single consistent image and will restart at a single point in time. This paradigm ensures dependent data produced by one application and consumed by another application will be consistently replicated such that out-of-order data is not replicated.

Consistency options for SRDF

The members of an SRDF/A group managed by Solutions Enabler allow a single member of the group to become unavailable for replication while allowing the other members of the group to continue to replicate. This is not a desired behavior, since it would be possible for order-dependent data to replicate out of order. This is known as tolerance mode:

- Consider tolerance mode "on" as the ability to partially replicate members of an SRDF/A group.
- Consider tolerance mode "off" as a consistency feature requiring all members of an SRDF/A group to replicate, or to suspend replication should one of the members become unavailable for replication.

The SYMCLI symrdf command enable turns off tolerance mode and ensures consistency within the SRDF/A group. As a best practice, EMC recommends enabling this feature when running SRDF/A.

The highest level of consistency is achieved with a composite group (CG) that leverages Enginuity Consistency Assist (ECA) and Multi-Session Consistency (MSC) and ensures all members of a CG are either replicating or not replicating at any given point in time. The CG definition can span SRDF concurrent and cascaded operations (SRDF/S and SRDF/A) and can span multiple Symmetrix DMX arrays. In the case of spanning arrays, MSC is enabled for SRDF/A device groups whereby all devices in the MSC cycle switch at the same time to ensure consistency of data across arrays. CG definitions are provided using Solutions Enabler.

In Concurrent or Cascaded SRDF/S and SRDF/A, each of the SRDF modes will be defined and will run within different SRDF groups. Each of these groups may be defined by a single CG but may be given a different name within the CG. CG operations and control can be as granular as the name within the CG definition, making it possible to automatically control the entire SRDF/S device set and/or the entire SRDF/A device set. This also ensures consistency of data should one of the SRDF modes stop replicating and prevent one mode of replication from stopping the other mode of replication in the event of a CG trip.

SRDF/Star granularity of control

SRDF/Star is a Concurrent or Cascaded SRDF solution based upon CGs for definition and operation. Controlling Star devices is only done at the CG or SRDF group-name within the CG. There is no device-level control available since this is a CG-based solution. The SRDF/A portion of a Star configuration utilizes MSC for cycle switching regardless of the number of source Symmetrix DMX arrays. MSC cycle switching is controlled by the Solutions Enabler SRDF daemon (storrdfd) that must be running for cycle switching to occur. Furthermore, SRDF/Star uses SDDF sessions to track incremental changes between the sites sharing the recovery links. The number of SDDF sessions available for Star is less than the number of SRDF groups supported on a Symmetrix DMX, potentially requiring further consolidation of applications in a smaller number of application-sets. This granularity of control is an important consideration in the design and implementation of SRDF/Star solutions.

SRDF/Star planning and design

Planning and design are by far the most important aspects of an SRDF/Star implementation. Once the design is documented, implementation becomes very straightforward.

The design considerations that follow are applicable to all SRDF/Star configurations. The key is keeping the design as simple as possible. Simplicity is most important when considering deployment of multiple Star-protected CGs on a single array. Whenever possible, EMC recommends keeping the SRDF group numbers, volume numbers, clone relationships, and other components as homogenous as possible across the arrays. It is also important to document the environment in such a way so that it can be easily referenced before, during, and after implementation.

Design considerations

Table 1 lists the major components that were identified and defined during the design of an SRDF/Star solution.

Table 1. Components used in an SRDF/Star solution

Component	Function
Solutions Enabler control hosts	The Solutions Enabler SRDF storrdfd daemon must be running, since it is responsible for cycle switching of the SRDF/A portion of the Star configuration. The daemon is also responsible for maintaining the SDDF sessions, allowing an incremental relationship between the two sites sharing the recovery links. Given the critical nature of this process in the SRDF/Star environment, a redundant pair of Solutions Enabler control hosts should be identified and deployed at each site where the workload could potentially be run. As a best practice, EMC recommends the following: Have dedicated Solutions Enabler hosts in the environment. These hosts can perform other Solutions Enabler commands such as queries and TimeFinder operations, and so on, but should not run any other workload. The hosts do not have to be large — modern 1U servers with access to gatekeepers on one or more arrays are adequate for this function. In this test an Intel 3 GHz system running Solarix X86 was used. When considering a large multiple CG Star implementation, have three Solutions Enabler control hosts so there will be at least two running in the event of a control host failing. Use more than one daemon to manage 32 Star-protected CGs and perform the cycle switching and SDDF session management.
Site names	Each site in a Star configuration must have a unique name. This name should be meaningful to the enterprise and easily identifiable where that site resides. Often it is useful to include application identifying information within the name so the name identifies both physical location and logical function. Note: SiteA, SiteB, and SiteC are common naming conventions used for testing and documentation purposes and are not recommended for a production installation.
Composite Group names	Similar to site names, choosing appropriate CG names is important for ease of use. The CG name is referred to on almost all command lines typed in the management of Star. Defining names that are easy to remember and easy to type is important. Many times the CG names reflect the application sets being protected by the CG. For example, a CG protecting an Oracle database could be oraapps_cg.
SRDF directors and ports	A minimum of four SRDF director ports are recommended for an SRDF/Star implementation. Separating the synchronous ports from the asynchronous ports and having redundant port-pairs is the recommended minimum configuration. For example, the configuration would have a pair of directors identified for SRDF/S and a pair identified for SRDF/A. SRDF configurations are required to start on slice D, port 0 of a director. Each of the three sites must have these four directors and ports identified and reserved. Note: As a best practice, EMC recommends using the same directors and ports at each site to minimize complexity.

<table>
<tr>
<td>SRDF group numbers</td>
<td>

In a single CG Star implementation, each “leg” of the Star configuration requires a unique group number. For a single CG Star, the minimum number of groups on each Symmetrix DMX is three. Although the groups do not have to be the same on each Symmetrix DMX, keeping the groups the same is the best practice.

Additionally, the groups may be either static or dynamic. Figure 5 shows a homogenous group numbering design for a single CG. Solutions Enabler group numbers (decimals) start at one (1). Enginuity native group numbers are internally represented in the Symmetrix DMX as a hexadecimal number starting at 0. Solutions Enabler shows group numbers in decimal and provides the hex number in parenthesis in many of the output displays. In the example that follows, the SYMCLI group number and the Symmetrix DMX internal hex group number are in parentheses. Management and operation of SRDF on the host always require the decimal-based SRDF group.

```
# symcfg -ra all list | more
      Symmetrix ID: 000190103334 (Local)

                 S Y M M E T R I X    R D F    D I R E C T O R S

                                         Remote        Local    Remo
Ident  Symb  Num  Slot  Type       Attr   SymmID         RA Grp   RA
      Status

RF-7D   07D   55     7  RDF-BI-DIR   -   000190103232   4 (03)   4 (
      Online
                                          -   000190103232   5 (04)
                                          -   000190103232  61 (3C)
                                          -   000190103232  42 (29)
                                          -   000190103232  45 (2C)
                                          -   000190103232  62 (3D)
                                          -   000190103232  67 (42)
                                          -   000190103232   4 (03)
```

</td>
</tr>
<tr>
<td>SRDF device pairs</td>
<td>

SRDF/Star requires devices to be DRx devices where the device is either Rdf1 or Rdf2 capable. The initial SRDF pairing relationship is created using the Dynamic SRDF `symrdf createpair` command. Again, for implementation and operational simplicity, keeping the device numbers consistent across sites is recommended. Typically, a pairsfile is created using an editor, and the SRDF pairings for SiteA to SiteB are placed in this file. The pairsfile is a file containing two columns of text that represent the device numbers for SiteA and SiteB in each column.

</td>
</tr>
<tr>
<td>Clone device pairs</td>
<td>

During a resynchronization of SRDF following a link failure or other transient fault condition, the R1 devices incrementally refresh the R2 devices. During this resynchronization, the R2 devices are not guaranteed to be consistent as the order of the tracks moving across the link during the refresh are optimized to minimize resynchronization time.

The resynchronization requires operational intervention to proceed and prior to this a gold copy of the R2 should be made using local replication in the R2 on the Symmetrix DMX. This gold copy provides a point-in-time recovery should a rolling disaster happen during the resynchronization to the R2. Ideally, there will be a clone identified for every DRx device at every site in the configuration.

Note: As a best practice, EMC recommends having similar source/target clone device numbering across all three sites.

</td>
</tr>
<tr>
<td>Create a detailed design document</td>
<td>

Create a detailed design document using a tool most appropriate. For example, a graphical tool is commonly used to depict the high-level and even detailed design and a table or spreadsheet tool is often used to create a written version of the design. While the toolset is not important, the output of the design is critical to the implementation and ongoing operation of the SRDF/Star environment.

</td>
</tr>
</table>

Figure 5 illustrates a single SRDF director for simplification but represents a homogenous configuration across all three sites. An actual design would have two directors for each of the SRDF links defined on each Symmetrix DMX. In our example, there are directors 3D and 14D supporting rdfg 2, directors 4D and 13D supporting rdfg 3, and directors 7D and 10D supporting rdfg 4.

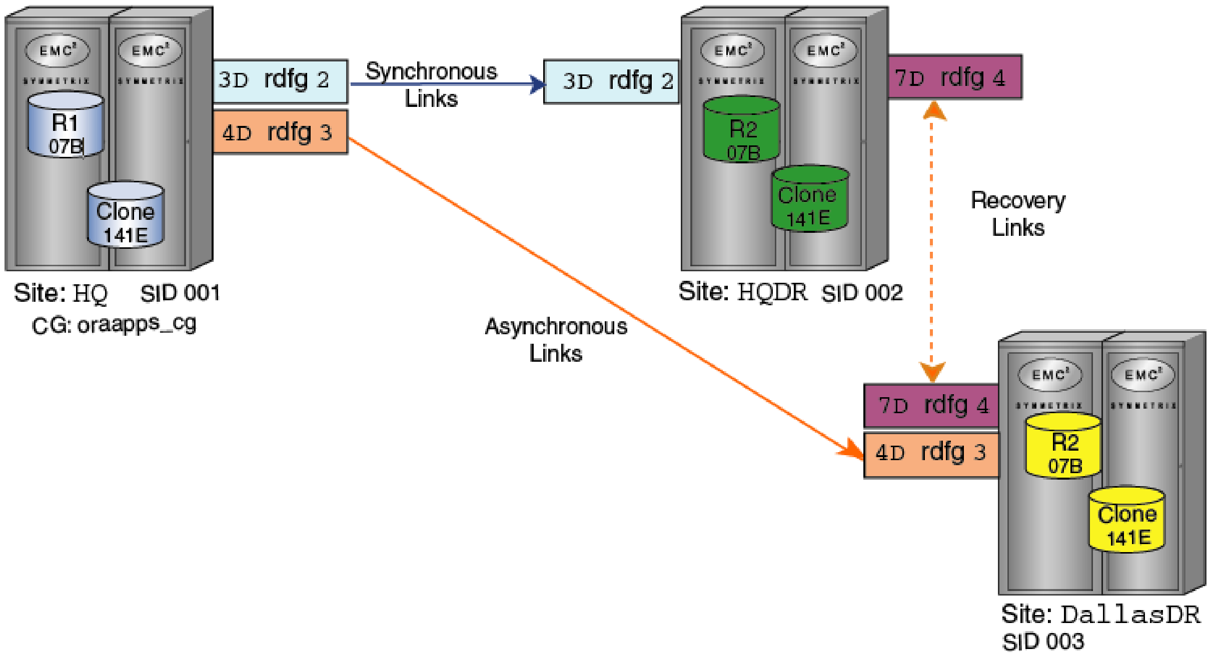

Figure 5. Basic SRDF/Star component design

Figure 6 shows a detailed design diagram of an implementation supporting 32 different Star-protected CGs. The level of detail in this diagram is the output of detailed planning and design. This is a living document that should be updated and maintained throughout the lifecycle of the implementation.

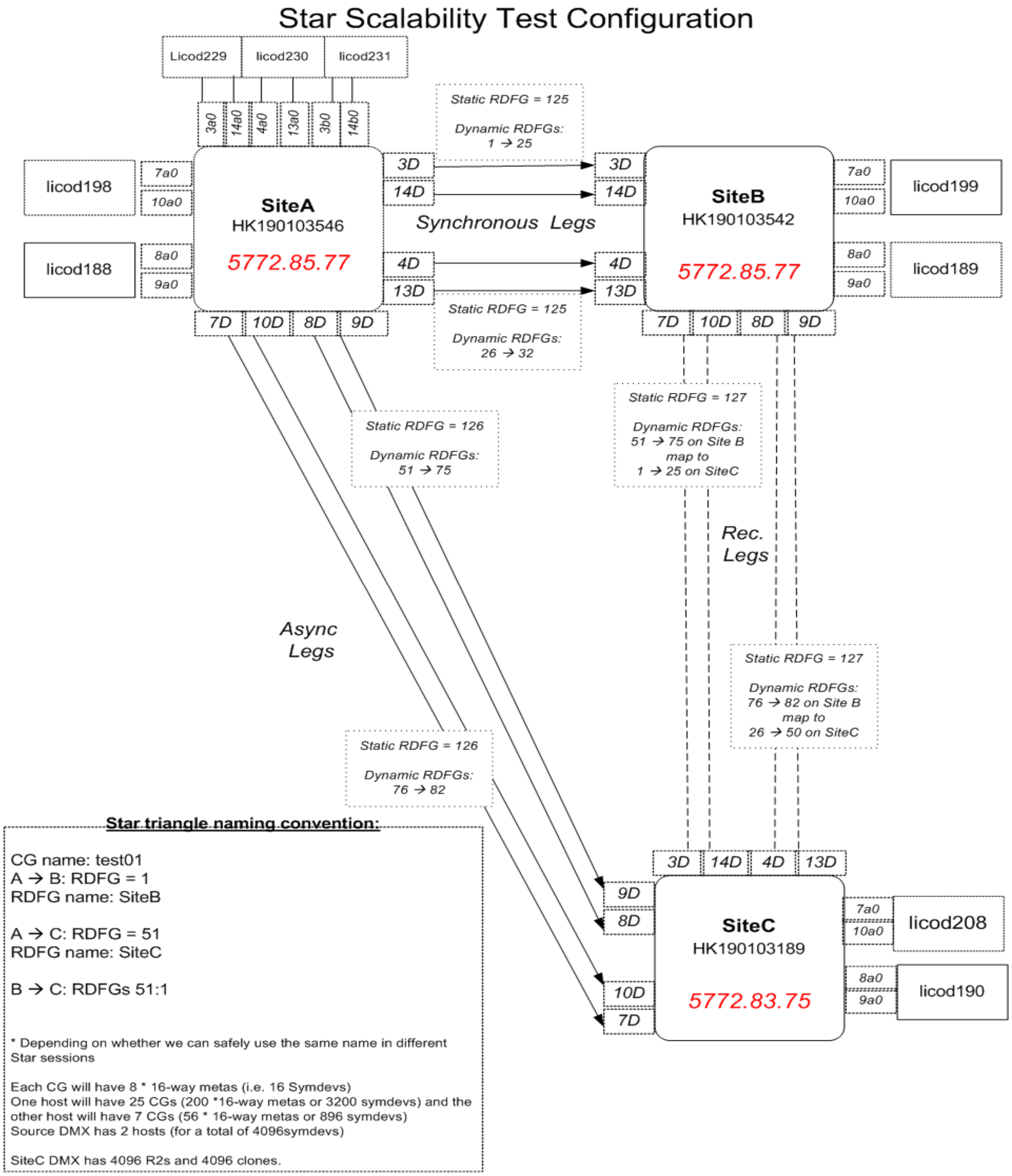

Figure 6. A detailed SRDF/Star design diagram

Implementing SRDF/Star

As previously mentioned, once the planning is complete and the design is documented, implementing SRDF/Star is straightforward. Prior to its implementation, there are several prerequisites that must be met. After meeting the prerequisites, Star implementation can commence and is usually set up in minutes.

Note: For purposes of illustration, the SRDF/Star environment that will be created in the following sections will match the design depicted in Figure 2.

Overview of implementation

The following sections detail the steps required to ensure the SRDF/Star environment has met the prerequisites:

- Check the prerequisites for an SRDF/Star environment.
- Create an SRDF/Star options file.
- Create and establish initial SRDF pairs.
- Create the composite groups and assign names to the synchronous, asynchronous, and recovery links.
- Perform the SRDF/Star setup function.
- Propagate the internal star definitions file to all control hosts at all sites.

Prerequisites for an SRDF/Star environment

There are several checks to be made in the environment to ensure SRDF/Star can be enabled. These pre-requisites are as follows:

- Concurrent SRDF must be Enabled.
- Dynamic SRDF must be Enabled.
- Dynamic_concurrent SRDF must be Enabled.
- Prevent RAs Online Upon Power Up must be Enabled (default).
- Prevent Auto Link Recovery must be Enabled for each Dynamic SRDF group created (default for SYMCLI dynamic groups).
- All devices must be DRX (Rdf1 or Rdf2 Capable).
- An adequate number of gatekeepers must be mapped to the control hosts.
- The SYMAPI options file must have SYMAPI_USE_RDFD=ENABLE (default is DISABLE).
- The SRDF storrdfd daemon must be running.

Checking the prerequisites

Using SYMCLI commands, check the prerequisites for the Star environment. The search utility used in these examples is UNIX grep. For Windows environments, using `findstr` in place of `grep` produces the same output.

- Check the SRDF configuration parameters allowing Concurrent, Dynamic and Concurrent Dynamic SRDF.

```
symcfg list –v –sid 3189 | grep RDF
 Switched RDF Configuration State     : Enabled
 Concurrent RDF Configuration State   : Enabled
 Dynamic RDF Configuration State      : Enabled
 Concurrent Dynamic RDF Configuration : Enabled
 RDF Data Mobility Configuration State: Disabled
 SRDF/A Maximum Host Throttle (Secs)  : 0
 SRDF/A Maximum Cache Usage (Percent) : 94
```

- Check the group-specific configuration to prevent the RAs from coming to the online state during a power-up and to prevent auto link recovery should the links become suspended.

```
symcfg list –v –rdfg 51
Symmetrix ID : 000190103189

                S Y M M E T R I X   R D F   G R O U P S

    Local             Remote                 Group              RDFA Info
--------------  ---------------------  -----------------------  ------------------
           LL                                     Flags  Dir  Flags  Cycle
 RA-Grp  (sec)   RA-Grp  SymmID        T    Name     LPDS  Cfg  CSRM   time    Pri
```

```
--------------  ---------------------  -----------------------  -----  -----  ---
 51 (32)    10   51 (32) 000190103546  D DYN51       XX.N  F-S  -AM-      30   33


 Group Flags       :
     Prevent Auto (L)ink Recovery      : X = Enabled, . = Disabled
     Prevent RAs Online Upon (P)ower On: X = Enabled, . = Disabled
     Link (D)omino                     : X = Enabled, . = Disabled
     (S)TAR mode                       : N = Normal, R = Recovery, . = OFF
  RDFA Flags        :
     (C)onsistency : X = Enabled, . = Disabled, - = N/A
     (S)tatus      : A = Active, I = Inactive, - = N/A
     (R)DFA Mode   : S = Single-session, M = MSC, - = N/A
     (M)sc Cleanup : C = MSC Cleanup required, - = N/A
```

The flags under the Group heading are defined in the Group Flags section at the bottom of the output. The "L" flag indicates Prevent Auto (L)ink Recovery and the X under the flag heading indicates it is enabled. Similarly, the P indicates Prevent RAs Online Upon (P)ower On and the X indicates this feature is enabled. In this example, both prerequisites have been met.

Once the SRDF configuration prerequisites have been met, the devices must be checked to ensure the devices are both Rdf1 and Rdf2 capable Dynamic SRDF devices:

```
symdev list -dynamic -rdf1 -rdf2
Symmetrix ID: 000190103334

        Device Name          Directors                  Device
--------------------------- ------------- -------------------------------------
                                                                            Cap
Sym  Physical               SA :P DA :IT  Config        Attribute    Sts   (MB)
--------------------------- ------------- -------------------------------------

0140 Not Visible            ***:* 15A:DD  RDF1+Mir      N/Grp'd      RW    2707
0141 Not Visible            ***:* 02C:CD  RDF1+Mir      N/Grp'd      RW    2707
0142 Not Visible            ***:* 15B:CD  RDF1+Mir      N/Grp'd      RW    2707
0143 Not Visible            ***:* 02D:DD  RDF1+Mir      N/Grp'd      RW    2707
0144 Not Visible            ***:* 15A:C4  RDF1+Mir      N/Grp'd      RW    2707
0145 Not Visible            ***:* 02C:D4  RDF1+Mir      N/Grp'd      RW    2707
```

The `–dynamic –rdf1 –rdf2` qualifiers will display only devices that are Dynamic RDF and RDF1 and RDF2 capable devices.

For further examination, use the following command:

```
symdev show 140 |  grep Capability
   Dynamic RDF Capability   : RDF1_OR_RDF2_Capable
```

The number of gatekeepers required to be visible to the control hosts is dependent upon Solutions Enabler and Enginuity release levels. With the introduction of Solutions Enabler 7.0 and Enginuity 5874 there has been reduction in the number of GKs required for deployment of a Star implementation.

- Solutions Enabler 7.0 and Enginuity 5874:
  A 2:1 ratio of CGs to gatekeepers is recommended.

- Solutions Enabler 6.5 (or earlier) and Enginuity 5773 (or earlier):

The sum of three plus the number of CGs being Star protected plus the number of other concurrent SYMCLI commands that may be running. The formula is:
3 + #CGs + simultaneous SYMCLI commands

Gatekeepers can be shown using the following command:

```
symdev list –cap 6
```

Check the options file to ensure the SRDF daemon is enabled. On UNIX hosts, the options file is located in /var/symapi/config. On Windows hosts, the options file is in c:\program files\emc\symapi\config.

```
grep RDFD /var/symapi/config/options
# Parameter:                         SYMAPI_USE_RDFD
#SYMAPI_USE_RDFD                   = DISABLE
```

Uncomment this option in the options file and set it to ENABLE using an editor such as vi.

```
grep RDFD /var/symapi/config/options
# Parameter:                         SYMAPI_USE_RDFD
SYMAPI_USE_RDFD                    = ENABLE
```

Check if the SRDF daemon is running:

```
stordeamon list –running

Available Daemons  ('[*]': Currently Running):
[*] storapid              EMC Solutions Enabler Base Daemon
[*] storwatchd            EMC Solutions Enabler Watchdog Daemon
```

Start the SRDF daemon:

```
stordaemon start storrdfd
  Waiting for daemon to start.  This may take several seconds.
stordaemon list -running
```

Available daemons are as follows ('[*]': Currently Running):

```
[*] storapid              EMC Solutions Enabler Base Daemon
[*] storrdfd              EMC Solutions Enabler RDF Daemon
[*] storwatchd            EMC Solutions Enabler Watchdog Daemon
```

Configuring SRDF/Star

After all SRDF/Star prerequisites have been met, the SRDF/Star components can be defined to set up and configure Star.

1. Create the Star options file. This file defines the site names for each of the three sites and optionally controls whether Cascaded SRDF mode may be used as a Star topology. The file is a plain text file and can be created with any editor. There are essentially three lines to enter into this file that specify the three site names for the configuration. There is an optional line to allow a Cascaded SRDF configuration. The three lines in the options file are as follows:

   ```
   SYMCLI_STAR_WORKLOAD_SITE_NAME    = <Wname>
   SYMCLI_STAR_SYNCTARGET_SITE_NAME  = <Sname>
   SYMCLI_STAR_ASYNCTARGET_SITE_NAME = <Aname>
   ```

   Where <Wname> is the primary workload site name, <Sname> is the synchronous secondary site name, and <Aname> is the Asynchronous secondary site name. Figure 2 represents the design for this configuration and the parameters would be set to the following:

   ```
   SYMCLI_STAR_WORKLOAD_SITE_NAME    = HQ
   SYMCLI_STAR_SYNCTARGET_SITE_NAME  = HQDR
   ```

```
SYMCLI_STAR_ASYNCTARGET_SITE_NAME = DallasDR
```

The optional parameter:

```
SYMCLI_STAR_ALLOW_CASCADED_CONFIGURATION = YES
```

This allows Star to operate in either Concurrent or Cascaded SRDF mode. This parameter can be set dynamically after Star has been initialized and set up. The site name parameters are static and must be present at initial setup time. If the optional parameter is set after the initial setup, run the following command to activate the new option in the file:

```
symstar setup –options <filename> -reload_options
```

2. Once the options file is created, create the dynamic SRDF groups. Dynamic SRDF groups are not a requirement for Star and may be statically defined by EMC Services. However, in this example, dynamic groups are created according to the simplified design diagram.

   Create the synchronous HQ->HQDR SRDF group:

   ```
   symrdf addgrp –label HQSYNC –rdfg 2 –sid 001 –dir 3D –remote_rdfg 2 –remote_sid 002
   –remote_dir 3D
   ```

   Create the asynchronous HQ->DallasDR SRDF group:

   ```
   symrdf addgrp –label HQASYNC –rdfg 3 –sid 001 –dir 4D –remote_rdfg 3 –remote_sid 003
   –remote_dir 4D
   ```

   Create the recovery HQDR->DallasDR SRDF group:

   ```
   symrdf addgrp –label HQDRDALLAS –rdfg 4 –sid 002 –dir 7D –remote_rdfg 4
   –remote_sid 003 –remote_dir 7D
   ```

3. Create the pairsfile. Now that the SRDF groups are defined, the dynamic SRDF pairs are created using a pairs file. The pairs file for this example has the following line in it. The first column is the R1 device; the second column is the R2 device.

   ```
   cat pairsfile.txt
   007B  007B
   ```

   - Prior to Solutions Enabler version 6.5, the pairs were typically created using the option `–invalidate` flag rather than `-establish`. The typical sequence was to create the pairs and invalidate the R2, set the mode of the pair to adaptive copy disk, and perform an establish.
   - Beginning with Solutions Enabler 6.5, a single command performs these three steps.
     The SRDF pairs must be defined for both the synchronous and asynchronous secondary targets:

     ```
     symrdf createpair –f pairsfile –sid 001 –rdfg 2 –type rdf1 -establish
     –rdf_mode acp_disk

     symrdf createpair –f pairsfile –sid 001 –rdfg 3 –type rdf1 -establish
     –rdf_mode acp_disk
     ```

4. Create the concurrent dynamic SRDF relationship. The following commands create the concurrent dynamic SRDF relationship between the three sites, sets the mode to adaptive copy disk, and performs a full establish of the pairs. The next step is to define the composite group (CG) and provide names for the connections between the sites.

   ```
   symcg create oraaaps_cg –type RDF1 –rdf_consistency
   symcg –cg oraapps_cg –sid 001 –rdfg 2 addall dev
   ```

The first two commands above define a name (used by Star) for the SRDF group. The syntax following `–rdfg` is the SID followed by the SRDF group number. The second two commands define the recovery links between HQDR and DallasDR using similar SID:group syntax.

At this point, the environment is ready to set up Star. The setup is done using the symstar utility. Subsequent Star management and operations are also done using `symstar` command rather than other SYMCLI commands such as `symrdf`.

```
symstar setup –cg oraapps_cg –options oraapps_options.txt
```

The `symstar setup` creates a Star internal definitions file that contains information specific to the site at which the CG was created and the setup was run. Prior to Solutions Enabler 7.0, this file must be propagated to all Solutions Enabler control hosts at all sites that will participate in the management and operation of Star. This is done by copying the internal definition file to those hosts. The filename is the name of the CG. With Solutions Enabler 7.0, there is no need to copy this file as it is globally visible to hosts connected to the Symmetrix arrays if configured as above.

```
ftp host2
ftp> bin
ftp> put /var/symapi/config/STAR/def/oraapps_cg
```

Once the file is copied, the CG definition for the other management hosts must be built to reflect the role of the site for which the CG is being defined. The CG definition for HQ is an RDF1 CG. The CG definition for HQDR is different. Furthermore, the CG does not exist on the other hosts. This step creates the CG on the other Solutions Enabler control hosts. The command should be run on a control host and the site name where that control host exists is the site name passed to the command. In this example, a host at the HQDR site is being used. A host at the DallasDR site would have passed the site name of DallasDR. A second host at HQ (provided GNS is not used) would provide HQ as the site name.

```
symstar –cg oraapps_cg buildcg –site HQDR
```

Star is now set up. A Star query of the CG shows both sites connected since the SRDF createpair performed the establish.

```
Site Name                                      : HQ

Workload Site                                  : HQ
1st Target Site                                : HQDR
2nd Target Site                                : DallasDR

Composite Group Name                           : oraapps_cg
Composite Group Type                           : RDF1

Workload Data Image Consistent                 : Yes
System State:
  {
  1st_Target_Site                              : Connected
  2nd_Target_Site                              : Connected
  STAR                                         : Unprotected
  }

Last Action Performed                          : Setup
Last Action Status                             : Successful
Last Action Timestamp                          : 06/05/2008_16:52:47

STAR Information:
  {
  STAR Consistency Capable                     : Yes
  STAR Consistency Mode                        : STAR
  Synchronous Target Site                      : HQDR
  Asynchronous Target Site                     : DallasDR
  Differential Resync Available                : No
  R2 Recoverable                               : Yes
```

```
  Asynchronous Target Site Data most Current  : No
  }

1st Target Site Information:
  {
  Source Site Name                            : HQ
  Target Site Name                            : HQDR
  RDF Consistency Capability                  : SYNC
  RDF Consistency Mode                        : SYNC
Site Data Image Consistent                  : Yes

    Source Site                     Target Site
  ------------------------------ -- ------------------------------ - ------------
            ST                    LI            ST                    M
        RD   A                     N Rem    RD   A                    O
  Symm   F   T   R1 Inv   R2 Inv  K Symm   F   T   R1 Inv   R2 Inv D RDF Pair
  ID     G   E   Tracks   Tracks  S ID     G   E   Tracks   Tracks E STATE
  ----- --- -- -------- -------- -- ----- --- -- -------- -------- - ------------

  02000 202 RW        0        0 RW 03232 202 WD        0        0 S Synchronized

   Totals:  -- -------- -------- --           -- -------- -------- - ------------
            RW        0        0 RW           WD        0        0 S Synchronized
  }
```

The second target site information is as follows:

```
  {
  Source Site Name                            : HQ
  Target Site Name                            : DallasDR
  RDF Consistency Capability                  : MSC
  RDF Consistency Mode                        : MSC
  Site Data Image Consistent                  : Yes

            Source Site                     Target Site
  ------------------------------ -- ------------------------------ - ------------
            ST                    LI            ST                    M
        RD   A                     N Rem    RD   A                    O
  Symm   F   T   R1 Inv   R2 Inv  K Symm   F   T   R1 Inv   R2 Inv D RDF Pair
  ID     G   E   Tracks   Tracks  S ID     G   E   Tracks   Tracks E STATE
  ----- --- -- -------- -------- -- ----- --- -- -------- -------- - ------------

  02000 203 RW        0        0 RW 03334 203 WD        0        0 A Consistent

   Totals:  -- -------- -------- --           -- -------- -------- - ------------
            RW        0        0 RW           WD        0        0 A Consistent
  }
```

Operational considerations

SRDF/Star operational paradigm

SRDF/Star offers a different operational paradigm than SRDF; Star is an application that provides a simple way to manage the three-site DR solution. Star performs the SRDF operations rather than a user needing to type and execute all the underlying SRDF commands required to fail over from site to site and all the other complexities surrounding a no-data-loss three-site solution.

Consider a Star configured as Concurrent SRDF performing a planned "failover" from the primary workload site (HQ) to the synchronous secondary site (HQDR). To accomplish this, the application(s) must be shut down, the links must be suspended and the secondary site has to swap from an R2 to an R1. But this swap is special in that it is only swapping the personality of the R2 to an R1 and leaving the original R1 alone (for a moment). This site must also create an incremental R1-R2 relationship with the asynchronous secondary site. Finally, the site must also swap the personality of the original workload site to be an R2 to enable it to retain an incremental relationship with that original workload site.

The SRDF commands to perform the above are non-trivial and, in fact, the functionality in SYMCLI/SYMAPI™ required to perform such a "failover" is only available using SRDF/Star – the symstar command. "Failover" is an SRDF term. In Star, the act of reconfiguring in such a way that allows applications to move from site to site is called "switching the workload site."

The above sequence of SRDF tasks is accomplished very easily with SRDF/Star. First, Star ensures all three sites have equivalent data using the halt command. Following this, the switch command is issued to perform all the underlying SRDF functions allowing the HQDR site to become the primary workload site and the other two remaining sites to become the synchronous secondary and asynchronous secondary sites. "Appendix A: Detailed outputs from SRDF/Star commands" has detailed output from these commands.

On the HQ control server:
```
symstar –cg oraapps_cg halt –noprompt
```

On the HQDR control server:
```
symstar –cg oraapps_cg switch –site HQDR –noprompt
```

At this point, the applications at HQDR may be brought online. Protection of the HQDR site is accomplished with the `connect`, `protect`, and `enable` sequence of `symstar` commands. Switching the workload back to the original primary site is accomplished in the same manner as the switch described above, but the command is run back on the original primary site.

SRDF/Star command overview

The `symstar` command is used to control and manage SRDF/Star configurations. Table 2 lists the common symstar commands in the operation of Star.

Table 2. symstar commands

`symstar`	Command and paramter/environment variable description
`setup`	**Builds the internal Star definitions file. Also used to clear Star indicators for repurposing devices into non-Star groups.**
`buildcg`	**Builds the internal Star definition file and the CG on management hosts.**
`show`	**Displays information about the Site relationships including SID and RDFG.**
`query`	**Displays the state of the Star environment from the host on which it is run.**

`verify`	**Validates the targeted site is in the state provided to the command.**
`connect`	**Initiates SRDF data flow on the links in Adaptive Copy Disk mode.**
`protect`	**Enables CG protection for the target site (name within the CG).**
`enable`	**Activates Star protection and SDDF sessions enabling differential resynchronization between any of the three sites.**
`disable`	**Deactivates Star protection and SDDF sessions, leaving the state Protected.**
`unprotect`	**Disables CG protection, leaving the site connected.**
`disconnect`	**Suspends replication to the targeted site.**
`halt`	**Write disables all devices in the CG and ensures all three sites are equivalent.**
`switch`	**Moves the workload site to the designated site. Performed at the site where the switch is targeted.** **The option `-keep_data` designates which data is most current and remotely restores that data to the targeted site of the switch.**
`reset`	**Recovers from transient fault of the SRDF links between sites. The option `-reconfigure` allows a dynamic reconfiguration between concurrent and cascaded topologies.**
`isolate`	**Suspends replication to the targeted site and Read Write enables the volumes, allowing maintenance or other activities at that site.**

SRDF/Star considerations for deploying multiple CG environments

SRDF/Star is designed to meet the needs of enterprise BC/DR requirements. The solution can scale a single CG across multiple Symmetrix DMX systems at any of the three sites. The solution can also scale within a Symmetrix DMX providing multiple CGs to be Star protected on a single Symmetrix DMX. This solution appeals to many open systems customers desiring a finer level of granularity over a single CG. While there are potential advantages to running multiple CGs, there are additional management and operational complexities to be considered.

Each CG in a multiple CG environment will undergo the entire design and implementation process described in this white paper independently of each other. There will be different RA groups for each CG, different names for those groups, separate Star internal definitions files ─ essentially each Star protect CG is built and managed independently as entirely autonomic entities. This in and of itself adds complexity to the management environment. Figure 3 is an illustration of a multi-CG environment and shows the complexity of such a design.

EMC has extensively tested multiple CGs running on a single Symmetrix DMX replicating to single DMX systems in the secondary sites. Operational considerations of such a complex environment include gatekeeper availability, SRDF daemon availability, responding to multiple simultaneous events (such as loss of links), responding to multiple simultaneous disasters (such as loss of a site), performing planned switches of an entire site, and more.

When implementing multiple CG Star environments, it is important to consider programmatic scripting control of the environment. This implies entities within the environment such as CG names will be defined in such a way that allows a loop to operate on each CG serially. Whether the name is appended with a numeral such as 1,2,3, or if the CG names are listed in a file and the file is parsed, the programmatic approach becomes important for the management of these large and complex environments. It is not practical to manage independent CGs without such scripts in place. Ideally a set of scripts is written to support each of the Star functions allowing an individual CG as an operand, a subset of CGs as operands, or the entire set of CGs as an operand. Scripting alone can be a significant amount of work relative to the setup of the environment.

Gatekeeper considerations

The number of gatekeepers needed for multiple CG environments has decreased in recent releases of Solutions Enabler 7 and Enginuity 5874. General guidelines for the numbers of GKs required for parallel management of multiple CG environments have been provided in this paper. For the latest information pertaining to gatekeeper requirements and deployment recommendations, please consult EMC.

SRDF daemon considerations

As previously mentioned, a minimum of two SRDF daemons must exist at each site where the workload may be switched. This redundant pair of daemons is required to continue operations critical to Star should a host running one of the daemons become unavailable or be taken out of service for maintenance.

Parallel operation consideration

As stated previously, responding to multiple failures across multiple CGs can be a challenge without proper scripting. Even with scripting, decisions about running multiple concurrent Star commands against multiple Star CGs must be considered. In the event the primary workload site is lost, the assumption is made that all CGs will switch the workload to a secondary site. There are several possible choices to perform the switching of the CGs to the secondary site:

- Execute symstar commands serially on a single control host for each CG.
- Execute symstar commands serially on a single host but balance the commands across multiple hosts.
- Execute symstar commands in parallel on a single host or multiple hosts.

In all cases, EMC recommends deploying the latest Solutions Enabler version available for the most optimized parallel execution environment possible. There are significant enhancements to the internal operations of Star in Solutions Enabler 7.0 that make parallel execution of sysmtar commands scale nearly linearly with the number of CGs in many cases.

Parallel operations using Solutions Enabler 6.4

EMC recommends running Star commands serially in all cases in Solutions Enabler 6.4 environments. This approach delivers the shortest RTO for any given CG and may provide the shortest RTO for all CGs. Given a serialized approach is recommended, prioritization of the order of switching the CGs must be considered. Which CG is to be switched first? Which is last? What is the order between the first and the last? These questions represent a sampling of the complexities beyond the technical details involved when considering multiple CGs in an environment.

Parallel operations using Solutions Enabler 6.5

There are significant improvements in Solutions Enabler 6.5 with respect to the implementation and efficiency of Star. These improvements allow more parallelism of Star commands running on a single Solutions Enabler host. The average performance gain running parallel connect, protect, enable commands in 6.5 relative to 6.4 is nearly 3X – meaning parallel connect, protect, enable sequences for multiple CGs on a single host runs nearly three times faster in Solutions Enabler 6.5. However, the individual connect, protect, enable commands run serially do not show vast improvement. The performance gains in running parallel operations come primarily from enhanced locking mechanisms between the SRDF daemon and the Symmetrix. It is possible to reduce overall RTO using Solutions Enabler 6.5 for all CGs combined running Star commands in parallel. These commands include the switch, halt, and connect, protect, enable. However, running commands in parallel increases the RTO of individual CGs. For example, if the serial runtime to perform a connect, protect, enable sequence of a single CG is 7 minutes, running multiple CGs in serial would simply be a multiplication of 7 minutes times the number of CGs being protected. Running parallel connect, protect, enable sequences provides an elapsed time of 7 minutes times the number of CGs divided by 2.5 or 3. However, in this paradigm an individual CG may take up to the entire duration of the parallel sequence to become protected itself.

Parallel operations using Solutions Enabler 7.0

Further enhancements to the parallel execution of the management of Star protected CGs have been included in version 7.0 of Solutions Enabler. The enhancements are such that there is excellent scalability in parallel execution of commands relative to the execution of a single command. This scalability is near linear when CPU and other resources exist.

Multiple Consistency Group summary

The flexibility provided by SRDF/Star is a consideration when choosing the number of CGs to deploy. Among the choices are application failover granularity, management complexity, scripting for serial and parallel execution, and RTO. Once there is more than a single CG in the environment, the choices and decisions to be made increase significantly. In all cases and all environments, a single CG typically provides the shortest RTO possible, yet there is no single solution that is recommended for any environment. The discussions and decisions about CG and application dependencies become paramount to the design phase of the Star solution and subsequently drive the implementation of the solution.

Conclusion

SRDF/Star is a scalable business continuance/disaster recovery solution that provides the flexibility to protect data across multiple arrays in a single CG and multiple applications across many CGs.
This flexibility can be leveraged to accommodate even the most complex environments yet the principles of design, implementation, and operations remain common to all SRDF/Star solutions. There is no universal blueprint for the number of CGs in an environment — that decision first depends on application data dependencies and often involves granularity and complexity of SRDF/Star operations. The detailed design of the SRDF/Star environment is the key to the successful deployment of an SRDF/Star solution

The Solutions Enabler `symstar` command is a robust application that simplifies the management and operation of SRDF/Star. The paradigm in this environment is to always use `Symstar` and not use underlying SRDF and CG operational commands. Once the design is complete, the implementation using Solutions Enabler simply becomes a matter of creating RA Groups, SRDF pairs, and CG(s), and running the `symstar setup` command. Once implemented, the management and operation of SRDF/Star are again simplified with the remainder of the `symstar` commands.

References

The following documents are available on EMC Powerlink® and provide additional information on SRDF/Star.

- *EMC Solutions Enabler Using SYMCLI to Implement SRDF/Star Version 6.2 Technical Note*
- *EMC Solutions Enabler Symmetrix SRDF Family CLI Product Guide*
- *EMC Symmetrix Remote Data Facility (SRDF) Product Guide*
- *EMC SRDF Host Component for z/OS Product Guide*
- *ResourcePak Base for z/OS Product Guide*
- *EMC SRDF/A and SRDF/A Multi-Session Consistency on Open Systems and Windows TechBook*

Appendix A: Detailed outputs from SRDF/Star commands

The sections that follow contain detailed outputs from the SRDF/Star commands.

Planned switch to HQDR

```
# symstar -cg oraapps_cg halt -nop

A STAR Halt operation is
in progress for composite group oraapps_cg. Please wait...

Halt ............................................................ Started
SA Write Disable Devs SID:000190102000 .......................... Started
SA Write Disable Devs SID:000190102000 .......................... Done
Not Ready Devs .................................................. Not needed
Drain remote data - checkpoint .................................. Started
Drain remote data - checkpoint .................................. Done
Suspend SID:000190102000 RDFG:203 ............................... Started
Suspend SID:000190102000 RDFG:203 ............................... Done
Suspend SID:000190102000 RDFG:202 ............................... Started
Suspend SID:000190102000 RDFG:202 ............................... Done
Disable STAR protection ......................................... Started
Disable STAR protection ......................................... Done
Deactivate STAR target SID:000190103232 ......................... Started
Deactivate STAR target SID:000190103232 ......................... Done
Deactivate STAR target .......................................... Not needed
Setting Star data consistency indicators ........................ Started
Setting Star data consistency indicators ........................ Done
Disable RDF Consistency protection .............................. Started
Disable RDF Consistency protection .............................. Done
Setting Star data consistency indicators ........................ Started
Setting Star data consistency indicators ........................ Done
Disable RDF Consistency protection .............................. Started
Disable RDF Consistency protection .............................. Done
Set Mode Adaptive Copy SID:000190102000 RDFG:202 ................ Started
Set Mode Adaptive Copy SID:000190102000 RDFG:202 ................ Done
Set Mode Adaptive Copy SID:000190102000 RDFG:203 ................ Started
Set Mode Adaptive Copy SID:000190102000 RDFG:203 ................ Done
Halt ............................................................ Done
#
# symstar -cg oraapps_cg  switch -site HQDR -nop

A STAR Switch operation is
in progress for composite group oraapps_cg. Please wait...

Switch_HQDR ..................................................... Started
Half Swap SID:000190103232 RDFG:202 ............................. Started
Half Swap SID:000190103232 RDFG:202 ............................. Done
Half Delete SID:000190103334 RDFG:203 ........................... Started
Half Delete SID:000190103334 RDFG:203 ........................... Done
Createpair SID:000190103232 RDFG:214 ............................ Started
Createpair SID:000190103232 RDFG:214 ............................ Done
Assign Name:DallasDR, Recovery RDFG:203 to SID:000190103232 RDFG:214 Started
Assign Name:DallasDR, Recovery RDFG:203 to SID:000190103232 RDFG:214 Done
```

```
Set Mode Adaptive Copy ............................................ Started
Set Mode Adaptive Copy ............................................ Done
SA RW Enable Devs ................................................. Not needed
Ready Devs SID:000190103232 ....................................... Started
Ready Devs SID:000190103232 ....................................... Done

SA RW Enable Devs SID:000190102000 ................................ Started
SA RW Enable Devs SID:000190102000 ................................ Done
Ready Devs ........................................................ Not needed
Half Delete SID:000190102000 RDFG:203 ............................. Started
Half Delete SID:000190102000 RDFG:203 ............................. Done
Half Swap SID:000190102000 RDFG:202 ............................... Started
Half Swap SID:000190102000 RDFG:202 ............................... Done
Switch_HQDR ....................................................... Done
```

Protecting HQDR after a planned switch

```
symstar -cg oraapps_cg connect –site HQ –nop

A STAR Connect operation is
in progress for composite group oraapps_cg. Please wait...

Connect_HQ ........................................................ Started
Setting Star data consistency indicators .......................... Started
Setting Star data consistency indicators .......................... Done
Disable RDF Consistency protection ................................ Started
Disable RDF Consistency protection ................................ Done
Set Mode Adaptive Copy ............................................ Not needed
Establish ......................................................... Started
Establish ......................................................... Done
SA RW Enable Devs ................................................. Not needed
Ready Devs ........................................................ Not needed
Connect_HQ ........................................................ Done

symstar -cg oraapps_cg connect –site DallasDR –nop

A STAR Connect operation is
in progress for composite group oraapps_cg. Please wait...

Connect_DallasDR .................................................. Started
Setting Star data consistency indicators .......................... Started
Setting Star data consistency indicators .......................... Done
Disable RDF Consistency protection ................................ Started
Disable RDF Consistency protection ................................ Done
Set Mode Adaptive Copy ............................................ Not needed
Establish ......................................................... Started
Establish ......................................................... Done
SA RW Enable Devs ................................................. Not needed
Ready Devs ........................................................ Not needed
Connect_DallasDR .................................................. Done

symstar -cg oraapps_cg protect –site HQ –nop

A STAR Protect operation is
in progress for composite group oraapps_cg. Please wait...
```

```
Protect_HQ ....................................................... Started
Wait for invalid tracks to reach 30000 ........................... Started
Wait for invalid tracks to reach 30000 ........................... Done
Set Mode Sync .................................................... Started
Set Mode Sync .................................................... Done
Enable RDF Consistency protection ................................ Started
Enable RDF Consistency protection ................................ Done
Wait until RDF consistency is achieved ........................... Started
Setting Star data consistency indicators ......................... Started
Setting Star data consistency indicators ......................... Done
Protect_HQ ....................................................... Done

symstar -cg oraapps_cg protect –site HQ –nop

A STAR Protect operation is
in progress for composite group oraapps_cg. Please wait...

Protect_DallasDR ................................................. Started
Wait for invalid tracks to reach 30000 ........................... Started
Wait for invalid tracks to reach 30000 ........................... Done
Set Mode Async ................................................... Started
Set Mode Async ................................................... Done
Enable RDF Consistency protection ................................ Started
Enable RDF Consistency protection ................................ Done
Wait until RDF consistency is achieved ........................... Started
Wait until RDF consistency is achieved ........................... In Progress
Wait until RDF consistency is achieved ........................... Done
Setting Star data consistency indicators ......................... Started
Setting Star data consistency indicators ......................... Done
Protect_DallasDR ................................................. Done

symstar -cg oraapps_cg enable -nop

A STAR Enable operation is
in progress for composite group oraapps_cg. Please wait...

Enable ........................................................... Started
Create STAR target ............................................... Not needed
Create STAR target ............................................... Not needed
Deactivate STAR target ........................................... Not needed
Deactivate STAR target SID:000190102000 .......................... Started
Deactivate STAR target SID:000190102000 .......................... Done
Clear STAR target SID:000190102000 ............................... Started
Clear STAR target SID:000190102000 ............................... Done
Clear STAR target ................................................ Not needed
Setting Star data consistency indicators ......................... Started
Setting Star data consistency indicators ......................... Done
Activate STAR target ............................................. Not needed
Enable STAR protection ........................................... Started
Enable STAR protection ........................................... Done
Wait until R2 Recoverable is achieved ............................ Started
Wait until R2 Recoverable is achieved ............................ In Progress
Wait until R2 Recoverable is achieved ............................ In Progress
Wait until R2 Recoverable is achieved ............................ Done
Enable ........................................................... Done
```

Appendix B: Runtimes for selected operations

Selected operations	Implementation	Solutions Enabler 6.4	Solutions Enabler 6.5
Switch to C, Keep B	Single session	Suspend: 6s Switch to C: 53s (average of several runs) Total: 59s	Disconnect –trip –site SiteC: 35s Switch to C: 59s Total: 1m 34s
	32 sessions sequentially	Suspend and Switch: Total: 31m 28s	Disconnect & switch: Total: 50m 8s
	32 in parallel on one host	Total: 30m 29s	Disconnect –trip –site SiteC: 1m 8 Switch to C: 6m 3s but some failed Total: 7m 13s
Connect, Protect & Enable at Site A	Single session	6m 51s	6m 35s
	32 sessions sequentially	~3h 39m 12s	~3h 30m 40s
	32 in parallel on Symm via 16 in parallel on each of two hosts	38m (Individual runtimes ranged between 25m and 38m. Average was 32m 40s.)	Not tested
	32 in parallel on one host	26/03/08: 1h 01m 02s (Individual runtimes ranged between 50m and 114m. Average was 1h 1m 2s.) (27/3/08 1h 8m 23s)	27m
Site A Halt (from Star enabled state)	Single session	3m 23s	2m 27 - 2m 57s
	32 sessions sequentially [16 sequentially]	~1h 48m 16s [16 seq: 00:54:08]	~1h 18m 24s - 1h 34m 24s
	32 in parallel on Symm via 16 in parallel on each of two hosts	54m 08s (1*16) 61m 53s (1*16)	5m 14s (1*15) 5m 39s (1*16)
	32 in parallel on one host	Not tested due to anticipated runtime	13m 52s