



Get to Know Your Symmetrix®

EMC Proven™ Professional Knowledge Sharing

Alastair Adamson
SAN & Storage Architect
apadamson@gmail.com

Table of Contents

Introduction.....	3
Design.....	4
Simplicity.....	4
Automation.....	5
System and User Considerations.....	5
Infrastructure and Operation.....	5
File Creation.....	5
File Access.....	6
Data Import.....	6
The Spreadsheet.....	6
Server summary.....	6
LUN Map.....	7
FA Port Usage.....	9
Space Usage.....	9
Mirrored Arrays.....	10
Extracting the Data.....	10
Device Information.....	11
Host and Masking Information.....	12
Device Group Information.....	14
Creating the Raw Data Files.....	15
The LUN Map.....	15
The Server Summary.....	16
FA Port Usage.....	16
Remarks.....	16
Importing the Data Files.....	17
Conclusion.....	18

Disclaimer: The views, processes or methodologies published in this compilation are those of the authors. They do not necessarily reflect EMC Corporation's views, processes, or methodologies.

Introduction

As a member of a team responsible for SAN and storage within your company's operations department, it is important to know the many ways that EMC Symmetrix and DMX disk arrays are configured and currently used. These might include the number and size of available HVEs (hypers), fan-in counts on the front-end Fibre Adaptor (FA) ports, which servers are using capacity on the array, and how much capacity they are using. Storage architects will find the same information valuable when designing the layout for new applications or preparing to migrate from one array to another.

While all of this information is available through Solutions Enabler commands, it is more convenient to have the information readily available in a spreadsheet. This is particularly true if you use Symmetrix Remote Data Facility (SRDF[®]) to replicate between sites and want to easily ensure that pairs of arrays are consistent.

This article describes how you can use Solutions Enabler commands in a generalised script to provide data that can be automatically imported into an Excel spreadsheet to provide:

- A detailed LUN map showing protection, type (BCV, R1, ...), use (gate-keeper (GK), SFS, ...), size, server accessibility, meta-LUN, mapping, and device group information per logical volume (LVOL)
- FA port usage per server and fan-in per port
- All servers masked to the array and the capacities they use
- Global capacity per type of usage (system, 2-way mirror, RAID-5, R1, R2, BCV and combinations, DRV) including a count of gatekeepers.

You can answer these questions almost immediately by using the spreadsheet.

- How much unused BCV capacity is left?
- Which FA ports are available for use for the new OS type being rolled out (without using heterogeneous mode)?
- What is the capacity used by a given server?
- Are there enough gatekeepers for my new project or do more need to be created?
- Which FA ports have a high fan-in ratio?

The spreadsheet can also be used to check the consistency of the way the array is being used and answer questions such as:

- Are the hyper/meta-LUNs in the SRDF-paired arrays symmetrically laid out per server?
- Is the use of ASCII WWNs (AWWN) consistent in the VCMDB?
- Are all servers masked to an even number of FA ports?
- Are there LDEVs which are mapped and not masked to any server?
- When a server has been decommissioned, have all traces been removed from the array including any device groups it may have used?

This will ensure compliance to best practices and that the configuration is “clean”.

This article shows how to extract the pertinent information from the arrays via Solutions Enabler SymCLI commands in a script, arrange the data according to the use of Symmetrix and DMX arrays locally, and present the information to the operational team via a spreadsheet. This is achieved via a shell script running on a UNIX server and Excel macros to import the data automatically.

Design

The object of the exercise is to have current, accurate and useful information about the Symmetrix (including DMX) arrays readily available. The information should be simply and automatically created and be updated at will. It should cater to independent arrays and also for pairs of arrays that are used symmetrically, for example to “mirror” the production array at another site.

Simplicity

If the tool is not easy, it will not be used. Scripts generate the necessary SymCLI commands and massage the output into the desired form (csv files) so that we can easily create the raw information. A macro reads the csv files and generates the Excel sheets including summary information not directly available from the raw.

Automation

UNIX system scripts can be run periodically via the standard *cron* utility. Information can be updated once a day or during the night so it is always accurate when the team arrives in the morning. It can also be run manually, since only a single “command” is needed. It could be part of the standard procedure to run this command after each provisioning session, so that it is always 100% up-to-date, though this may be excessive.

The Excel macro that automates the reading and formatting of the data relieves the removes operations team. Either run the maro when the data is created, or run it every time the Excel sheet is opened. No more need for Excel experts.

System and User Considerations

Either UNIX or Perl could have been used for scripting. We chose UNIX due to the ease of using the *shell* (Korn shell in this case) for that purpose, and also for the rich assortment of standard tools available to reorganise the output of the SymCLI commands into the desired raw data form.

Excel was chosen as the “user” interface because a spreadsheet layout allows easy data manipulation. Also, Excel is readily available and often used (hence reducing training requirements), and the choice of a spreadsheet under UNIX would be less portable from one site to another.

Infrastructure and Operation

The tool has three parts. First, the raw data files (*csv*) are created on a UNIX system. Second, the files are made available to an operating system where Excel can be run. Third, the *csv* files are automatically imported into the Excel spreadsheet and the information is made available to users.

File Creation

The raw data files are created on a UNIX system. The only requirement is that Solutions Enabler is installed on the system and that (at least) one gate-keeper (GK) is provisioned from each Symmetrix array from which information is desired. This latter isa requirement to use SymCLI commands against that array. Nothing else is needed as the script only uses standard UNIX commands. The script can be run automatically via *cron* at a frequency compliant with the provisioning procedures in place.

File Access

There are several ways to make the raw files accessible to the operating system where Excel will view the information. This can be either manual or automatic depending largely on the infrastructure.

The common File Transfer Protocol (FTP) can be used to transfer the files from the UNIX system to the system from where Excel will be used. This can be done manually or it can be scripted if supported by the underlying infrastructure.

The files can also be made immediately and automatically available by placing them in a directory accessible from both UNIX and the host environment. For example, they could be put in a NAS share that is accessible via both the NFS and CIFS protocols. The same result could be obtained using a freeware product such as Samba.

Data Import

When the raw csv files are available, the Excel macro only has to know where to find them. The macros do the rest of the work. There is no restriction as to where the Excel file itself resides.

The Spreadsheet

The resultant Excel spreadsheet consists of four pages. Three are generated directly from the raw data files output by the script. These provide information on the capacity used by each server, information on each of the volumes in the Symmetrix, and information on the use of each of the active FA ports. The fourth sheet gives overall figures on the space used by the different types of device in the array and is generated by the Excel macros when the csv files are read in.

Server summary

This sheet has three columns for the host name(s), the total provisioned capacity (in GB) and the space used by gate-keepers (in MB). The first column includes all hosts using the same space. Thus, in the case of a cluster, all nodes will be named. If one node in a cluster has LUNs assigned that are not available on the other node, a separate line will be used. If a Celerra NAS gateway shares the same array, data movers will also be listed.

New York

Hostname	Claimed (GB)	GKs (MB)
ESCON	493.68	0.00
Fred	252.88	3.00
Charlie Mike	33.72	0.00
Jim	0.00	12.00
Ricky	184.38	6.00

From the example sheet above, it would appear that the servers *Charlie* and *Mike* are in a cluster and that *Jim* may well be used for Symmetrix administration since it only has GK's assigned to it.

LUN Map

This is the largest sheet, with one line per LVOL. There are nine columns, as follows:

1. The number of the volume within the array, as a hexadecimal number
2. The RAID type of the volume, or if it is a Vault or DRV volume
3. The special use of the volume, such as SFS, VCMDB, GK, or DRV
4. The name(s) of the host(s) accessing the LUN, with the same considerations as above
5. The capacity in MB
6. Whether it is part of a meta-LUN or not. If it is, the column indicates whether it is the meta-head or a simple member
7. Whether the volume is mapped to one or more FA ports or not
8. The name of the device group in which it appears, if any
9. If the volume is within a device group, the LDEV name within that group

For meta-LUNs, the capacity of the entire meta-LUN is shown against the meta-head member and the other members are shown as having 0 capacity. This is consistent with the output of the `"symdev list"` command.

New York (1234)

Vol	Label	Type	Hostname	Cap MB	Meta	Mapped	Device Group	Ldev name
0000	VLT			5120				
0001	VLT			5120				
0002	VLT			5120				
0040	2M	SFS		5756				
0041	2M	SFS		5756				
0042	2M	VCM		3		*		
0043	2M	GK		3		*		
0044	2M	GK		3		*		
0053	2M	GK	Fred	3		*		
0054	2M	GK	Jim	3		*		
0055	2M	GK	Jim	3		*		
0056	2M	GK	Jim	3		*		
0057	2M	GK	Jim	3		*		
0058	2M	GK	Ricky	3		*		
0059	2M	GK	Ricky	3		*		
0066	R1M		ESCON	2078		*		
0067	R2M		ESCON	11619		*		
0068	R2M			11619				
0069	R2M			2078				
006C	R2M			2078				
006D	R1M		ESCON	8631		*		
006E	R1M		ESCON	8631		*		
006F	R1M		ESCON	8631		*		
0070	R5		Fred	34524	M	*	Fred_R1	DEV010
0071	R5		Fred	0	m	*	Fred_R1	DEV011
0072	R5		Fred	0	m	*	Fred_R1	DEV012
0073	R5		Fred	0	m	*	Fred_R1	DEV013

The sample sheet above shows the type of data in this sheet.

FA Port Usage

This page shows all FA ports to which at least one host is masked, each port having its own column. For each of these ports, all the servers masked to that port are shown, one per row. Each host has its own line, so this sheet also gives a complete list of hosts masked to the array. Totals are given, showing the fan-in ratios per port.

In the example below, you can see that the “rule of 17” has been respected and that the ports 8B:0 and 9A:1 have a higher fan-in count than the others. If some sort of operating system (OS) -specific coding is used in the AWWN coding convention, the ports used for a given OS would also be immediately recognisable.

New York FA port usage	3A:0	4A:1	8B:0	8B:1	9A:0	9A:1	13B:0	14B:1
Fred	1							1
Charlie			1			1		
Jim		1					1	
Mike				1	1			
Ricky			1			1		
total	1	1	2	1	1	2	1	1

Space Usage

This sheet is generated directly within Excel from the data in the LUN Map page. It provides the total capacity by type of volume. It shows the space reserved by the array (for the VCMDB, SFS, and Vault (on a DMX-3 or -4)), the space by each RAID type, as standard volumes, BCV, R1, and R2, DRV volumes (for example, for Symmetrix Optimizer), and others. Finally, the number of gate-keepers is given. These figures, apart from the system reserved devices, are given for both masked and unmasked volumes. An example of what this sheet might look like follows.

New York	Masked (GB)	Unmasked (GB)
Inaccessible (System)	651.25	
Accessible :		
RAID 1	809.19	983.50
RAID 5	36719.45	12650.99
BCV	0.00	0.00
BCV RAID 5	0.00	0.00
R1 mirrored	16906.62	0.00
R1 RAID 5	4349.50	0.00
R1 BCV RAID 5	0.00	0.00
R2 mirrored	13.52	19.46
R2 RAID 5	0.00	0.00
R2 BCV RAID 5	0.00	0.00
DRV	0.00	50.57
Other	0.00	0.00
Gatekeepers (number of)	136	271

Mirrored Arrays

It is practical to view two (or more) mirrored arrays side by side. This would be appropriate when the production array on one site is mirrored to another array at a different site. The spreadsheet enables this type of viewing for all four sheets, with user-defined identification of the arrays in question.

Extracting the Data

It is important that correct information is displayed on the spreadsheet. Obtaining data directly from the Symmetrix guarantees the validity of the information and removes the possibility of human error.

The decision to use Solutions Enabler SymCLI commands was straightforward. In fact, only three commands are needed to get all the information we wanted.

- *Symdev* gives information on the volumes defined, their type, and capacity.
- *Symmashdb* provides the link between the devices and the host(s) that can access them.
- *Symdg* shows device group information so that, if used, all the devices linked in a group can be easily identified.

Device Information

Step 1: Individual device information (volumes) within the array is obtained using the *symdev* command. The (partial) output of “*symdev list*” resembles the following.

Symmetrix ID: 000190100097

Device Name		Directors		Device			
Sym	Physical	SA	:P DA :IT	Config	Attribute	Sts	Cap (MB)
04AA	Not Visible	03A:0	01A:C2	2-Way Mir	Grp'd	(M) RW	17261
04AE	Not Visible	03A:0	01D:C3	2-Way Mir	Grp'd	(m) RW	-
04B2	/dev/rdisk/c5t0d4s2	03A:0	16C:D2	2-Way Mir	Grp'd	(M) RW	17261
04B6	/dev/rdisk/c5t0d5s2	03A:0	16C:C3	2-Way BCV Mir	Grp'd	RW	17261
04BA	/dev/rdisk/c5t0d6s2	03A:0	01A:C4	RAID-5	Grp'd	(M) RW	17261
04BB	/dev/rdisk/c5t0d7s2	03A:0	01A:D5	RAID-5	Grp'd	(m) RW	17261
04C6	Not Visible	???:?	01A:C4	Unprotected	N/Grp'd	RW	516
0E2A	/dev/rdisk/c5t0d0s2	03A:0	15D:CD	Unprotected	N/Grp'd	RW	6
0E2B	/dev/rdisk/c5t0d1s2	03A:0	15D:CD	Unprotected	N/Grp'd	RW	6

The listing is in an easy-to-read tabular form. However, it presents some problems when being parsed by a script. For example, the “Physical” column contains either the name of the disk device as seen by the operating system, if the LUN is visible to the current host, or “Not Visible”. The first is one “word” and the second is two “words”. The “Config” column also has a variable number of “words”. Further, the contents of this column can be quite long and an abbreviated version is preferred in the spreadsheet, such as using “2M” instead of “2-Way Mir”.

To solve this, the script first massages the output from the *symdev* command, ensuring each column has only one “word” and also changing the contents of the “Config” column to shorter names, as they will appear in the spreadsheet.

Another problem remains; the number of columns, after this massaging, is still variable. If the volume is part of a meta-LUN or the Symmetric File System (SFS), a ninth column will be present. This is useful spreadsheet information, so the script has to check for the number of columns before knowing how to parse the line.

Thus, the *symdev* command gives us the following information that will appear in the spreadsheet.

- The volume number (“Sym” column)
- Whether or not the volume is mapped to a FA port (the “SA:P” column will contain “???:?” if it is not mapped)
- The type and use of the volume (“Attribute” column)
- If it is part of a meta-LUN or not (“(M) for the head of a meta-LUN, “(m)” for a member, or nothing if not relevant)
- The capacity of the device (note that no value (“-”) is shown for meta-LUN members; the capacity of the entire meta-LUN is shown against the head of the meta-LUN).

Host and Masking Information

Step 2: Next, we need to know the host(s) to which each of the devices has been masked, if any. This information is held in the VCMDDB and can be obtained via the *symmaskdb* command. The (truncated) output of “*symmaskdb list db*” could appear as below.

```

Symmetrix ID          : 000000006196
Last updated at      : 04:58:00 PM on Tue Mar 25,2004

Director Identification : FA-2A
Director Port         : 1

      User-generated
Identifier  Type  Node Name      Port Name      Devices
-----
10000000c9238053  Fibre  api145         i@1f,4000,@2  0040:0043
10000000c924e04a  Fibre  HOST.23.65.70  10000000c924e04a  00BC:00BF
                                                00C3:00C6
                                                00D4

Director Identification : FA-2B
Director Port         : 1

      User-generated
Identifier  Type  Node Name      Port Name      Devices
-----
10000000c9238053  Fibre  api145         i@1f,4000,@2  None

```

The output of the `symmaskdb` command is more difficult to parse than the `symdev` command, but it provides the information we seek. The WWN of the HBA is shown in the “Identifier” column. The ASCII WWN (AWWN) is shown in the “Node Name” column. If no AWWN has been assigned to a HBA, the WWN itself will appear in the spreadsheet. It is a best practice to always use AWWN’s when masking as this clearly identifies the host in question.

People don’t tend to know their WWN’s! AWWN’s can be assigned automatically using the command `symmask discover hba`. While information about the FA ports and devices are also present, we will use another command to get this information, in a form that is more directly exploitable, as described later.

The output above is used to generate a list of correspondences between WWN’s and AWWN’s, using the first and third columns only.

Step 3: To get real masking information, the command `symmaskdb list assign` is used, with a list of devices indicated using the `-devs` option. The output, shown below, will be used to provide the name(s) of the host(s) to which LUNs are masked, in the LUN map, and also to get the FA port usage information.

```
Symmetrix ID : 000087431126

Device  Identifier          Type  Dir:P
-----  -
0470    50060b000009a9e2    FIBRE FA-3A:0
        50060b0000070536    FIBRE FA-3A:0
        50060b000009b126    FIBRE FA-14A:1
        50060b0000070614    FIBRE FA-14A:1
0476    50060b000009a9e2    FIBRE FA-3A:0
        50060b0000070536    FIBRE FA-3A:0
        50060b000009b126    FIBRE FA-14A:1
        50060b0000070614    FIBRE FA-14A:1
```

The only problem is that the output contains WWN’s and we prefer to use host names (AWWN’s) where possible. This output is edited to replace each WWN by the corresponding AWWN as identified from the `symmaskdb list db` command’s output (from step 2).

Step 4: Masking, however, is not used for devices provisioned over ESCON links. The host information should not appear blank in the spreadsheet as such volumes would be interpreted as being unused, which is not the case. Since there is no masking, the “name” of the host using the ESCON link cannot be identified, but the spreadsheet can at least indicate the LUNs that are available using this protocol. We use the command “`symdev list - escon`” to get a list of these volumes and simply note the device numbers concerned.

Device Group Information

Step 5: The use of device groups is the last piece of information required before we can create the csv files. We generate this data using the `symdg` command. Device group names over 18 characters long are truncated (characters 18 to the end are replaced with a “*”) with the “`symdg list`” command so the “-v” option must also be used. This produces far more output than desired but full group names are shown. The sample output has been severely cut showing only the basic structure and the lines of interest to the script (in bold type).

Group Name: Host451_Mirror

```
Group Type           : RDF1
Valid                : Yes
Symmetrix ID        : 000187900267
...
```

```
Number of STD Devices in Group      : 4
Number of Associated GK's           : 0
Number of Locally-associated BCV's  : 0
Number of Locally-associated VDEV's : 0
Number of Remotely-associated BCV's (STD RDF) : 4
Number of Remotely-associated BCV's (BCV RDF) : 0
Number of Remotely-assoc'd RBCV's (RBCV RDF) : 0
```

Standard (STD) Devices (4):

```
{
-----
LdevName          PdevName          Sym          Cap
Dev  Att.  Sts          (MB)
-----
DEV001          N/A              0002       RW         1031
DEV002          N/A              0003       RW         1031
...
}
```

BCV Devices Remotely-associated (STD RDF) (4):

```
{
-----
LdevName          PdevName          Sym          Cap
Dev  Att.  Sts          (MB)
-----
RBCV001        N/A              0048       NR         1031
RBCV002        N/A              0049       NR         1031
...
}
```

```

Device Group RDF Information
{
  RDF Type                : R1
  RDF (RA) Group Number   : 1 (A)
  Remote Symmetrix ID     : 000187908326
  ...
}

```

A list is made from this output including the device number, group name, and LDEV name (the name of the device as it is known within the device group). This is fairly straightforward except for the remote (R2) devices, which want to be linked with the “Remote Symmetrix ID” but this field comes further down the output than the list of R2 devices concerned.

To get around this, the output from the command is read twice. The first time, a list of device group names is generated for valid groups only, with both the local and remote Symmetrix IDs (if it exists). During the second pass, the list of device numbers for each device group is generated - along with the group and LDEV names - into a file named with the appropriate (local or remote) Symmetrix ID.

Creating the Raw Data Files

With all the information at hand, as generated above, three raw data (csv) files are created per disk array.

The LUN Map

The information from the “`symdev list`” command (Step 1) immediately gives the device number, the device type and usage (apart from gate-keepers), capacity, meta-LUN, and mapping information. The device is deemed to be a gate-keeper if its size is less than 10 MB. Normally, GK’s use 6 cylinders (about 3 MB). To obtain the hostname using each volume (if any), we first look in the output from the “`symmaskdb list assign`” command (Step 3) which now contains host names rather than WWN’s. There may be more than one host accessing the device, as in the case of clustered servers.

If this produces no result, the list of devices from the “`symdev list -escon`” command (Step 4) is checked to see if the device is available via an ESCON link. If there is still no result, the device is not available to any host and thus the hostname field in the spreadsheet will be empty. If the device number is found in the device group information file for the Symmetrix we are currently dealing with (produced in Step 5), the device group and LDEV names are taken from there and used to fill the corresponding fields in the csv file.

The Server Summary

We examine each line in the LUN map file that we just created to produce the summary of the capacity available to each host. For each line that has a host name, we get a complete list of the hosts accessing the array, and sum the capacities of the volumes accessible to that host for both GKs and other devices. This latter distinction is made because GK capacity is reported in MB whereas other device capacity is shown in GB.

FA Port Usage

All information needed to produce this table comes from the masking data generated in Step 3. However, as opposed to all the other pages in the spreadsheet, the number of columns in this table is not known in advance. There is one column for each FA port to which at least one host is masked. ESCON and SRDF ports are not shown here. There is one row for each host masked to any port on the array.

From the output generated in Step 3 above, a list is generated with one line per unique hostname/FA-port pairing. From this list, the FA-port names are added as column headers to the output csv file. Finally, for each host in the list, a row is created with a “1” in the column corresponding to the FA-port(s) to which it is masked.

Remarks

The device group information does not “belong” to any one array. The group information used by the script is in fact generated first, and only once independent of the number of arrays for which the csv files will be created. The three csv files described above are then generated for each nominated Symmetrix (via the command line arguments or a pre-defined list within the script).

The output of the SymCLI commands differs slightly depending on the version of Solutions Enabler being used. The script has been used with versions 5 and 6 (various sub-levels) and on Symmetrix 5, DMX-2 and DMX-3 arrays and currently contains no version-specific differentiation.

Importing the Data Files

When the *csv* files are available on the system from where Excel will be used, an Excel macro called *readFiles* will read them and display the information in four separate sheets. There are no pathnames built into the macros; the *csv* files are simply expected to be in the same directory as the Excel file itself. (This could easily be changed to either an explicit path or a path relative to the directory where the Excel file resides.) The *csv* files created by the UNIX script have names of the form *<info>_<SID>.csv* where *info* differentiates between the three files created and *SID* is the Symmetrix array in question. Since the filename format is fixed, the only information required to read in the files is the *SID* of the array and a site-specific information string used in table headers. Note that this is only useful when the files from two or more arrays are shown side by side. This information is given as arguments to the *importData* macro which is called from the *readFiles* macro, once per array. Everything else is automatic and needs no change.

The main macro clears all the sheets since there may be fewer rows or columns than in the initial state when the Excel file is first opened. It then reads in the *csv* files using the *QueryTables.Add* method. The FA port sheet is a bit different from the others as it gives fan-in totals below each of the (port) columns and draws some lines. Finally, the Space sheet is created by summing the capacities per type of device (RAID, BCV, R[12], system) from the information in the LUN Map sheet. It provides sums for both masked and unmasked devices and also shows used and unused gate-keeper counts.

The reading of the *csv* files can easily be made automatic when Excel opens the spreadsheet. Simply have the *readFiles* macro called from the inbuilt *Workbook_Open* subroutine called from the standard *ThisWorkbook* object. In this case, it would also be advisable to set *ActiveWorkbook.Saved* to "True" when the workbook is closed, as there is no point in saving contents that will be over-written the next time the spreadsheet is opened.

Conclusion

A fairly simple shell script (around 80 lines) and a few Excel macros (about 200 lines) are sufficient to provide a useful tool for Symmetrix array administration that can be used to help in daily operational tasks, check the cleanliness and consistency of the setup, and answer the boss's queries almost instantaneously. Moreover, all of this is achieved using just three SymCLI commands and a bit of data reorganization.

Biography

Coming from the UNIX world, I have installed a dual-site SAN with full disaster recovery using BCVs and SRDF, migrated the data centres (and SAN!) from Paris to London with minimal downtime and migrated from a single tier infrastructure to newer arrays with three tiers based on CLARiiON and DMX arrays. I have participated in defining SAN administration procedures and written several tools to aid in administration and automate tasks to ensure up-to-date and correct information is available to manage the SANs. I have been involved in SANs in both administrative and management roles over the last eight years. Currently I serve as SAN and Storage Architect for a large European institution.