

Solaris Volume Manager : Descriptive Names

Descriptive Names

- An enhancement in SVM to remove the naming restrictions on metadevices and hotspare pools.
- Also known as friendly names.

Current Name Definition

- Metadevices
 - ◆ dXXX where XXX is an integer between 0 and 8191
- Hotspare Pools
 - ◆ HspYYY where YYY is a 3 digit integer between 000 and 999

Descriptive Names

- Metadevice and hotspare pool names extended to made up of alphanumeric characters plus the characters '-', '_', and '.'. Names must begin with a letter. Names are limited to 256 characters.
- Metadevice and hotspare pool names will share a name space. This means that it will not be possible to have a hotspare pool named 'B' and a metadevice of the same name within a single set.
- Additional constraint that 'none', 'all' and 'mddb' are not allowed as valid descriptive names.
 - ◆ The reason for this is that they are used in metaparam and metahs and it would not be possible to disambiguate.

Device Name Space

- /dev/md/[r]dsk/My_Metadevice
- /dev/md/set1/[r]dsk/AccountingMirror
- Hotspare pools cannot be opened or manipulated by the end user so it is not necessary to create entries in the device name space.

md.tab changes

- md.tab(4) can be used by metainit(1m) and metadb(1m) to create metadbs, metadevices, and hot spare pools
- Descriptive names can now be used in md.tab files.
- One restriction is that it is still necessary to specify metadbs with the 'mddbXXX' syntax.
 - ◆ Without this requirement it is not always possible to differentiate between a hot spare pool and a metadb

Downgrade Issues

- Use of descriptive names creates changes in the metadb that are incompatible with previous versions of SVM.
- To prevent possible corruption in the case of a user downgrade the metadb records are versioned. During metadb snarf if an incompatible record version is encountered then the snarf will fail and SVM will not start.
- In the unlikely case that a user wishes to downgrade it is necessary to remove all of the friendly name metadevices and hot spare pools.

Downgrade Issues (2)

- To assist the user the '-D' option has been added to metastat. All of the descriptive name devices will be listed when this option is used

Userland Changes

- No longer possible to map explicitly between minor and metadvice name
 - ◆ Get rid of all code that took advantage of this
 - ◆ Add calls necessary to map from minor to name and name to minor
- No longer possible to determine the name type by name.
 - ◆ Use context whenever possible
 - ◆ Add code to disambiguate when necessary

Name Mapping

- All of the code that either directly created a `dev_t` from a canonical metadvice name had to be changed.
 - ◆ Now necessary to make an `ioctl` call to `md` to determine the mapping. The function in `libmeta` to make this translation is `meta_getnmXXX`.
- All of the code that created a canonical metadvice from a `dev_t` also had to be changed.

Name Mapping - Old Style

```
/* build corresponding device name */
if (metaislocalset(sp)) {
    uname = Malloc(20);
    (void) sprintf(uname, "d%lu", MD_MIN2UNIT(mnum));
}
else {
    len = strlen(sp->setname) + 1 + 20;
    uname = Malloc(len);
    (void) snprintf(uname, len, "%s/d%lu", sp->setname,
        MD_MIN2UNIT(mnum));
}
```

Name Mapping – New Style

```
/* get corresponding device name */  
dev = metamakedev(mnum);  
if ((uname = meta_getnmentbydev(sp->setno, MD_SIDEWILD,  
dev,  
NULL, NULL, &key, ep)) == NULL)  
return (NULL);
```

Determining Name Type

- Code has been added to disambiguate names since it is no longer possible to tell from the syntax of the name what it is.
 - ◆ Metainit foo -p **c1t1d0s0** 10m
- It is a potentially expensive operation to disambiguate so contextual information is used whenever possible.
 - ◆ Metahs -a **acctg_hsp c0t10d0s0**
 - ◆ The function metaname has been augmented to take a type. This type can be:
 - ◆ Metadevice
 - ◆ Logical device
 - ◆ Unknown

Determining Name Type (2)

- If the type is 'unknown' then it is necessary to disambiguate. This means determining what type of device it really is.
 - ◆ Only a metadvice
 - ◆ Only a logical device
 - ◆ Both
 - This would happen in the case where a user had created a metadvice called c1t1d0s0 and also had a disk on the system of the same name.
 - In this case the SVM application will fail and request that the full path name be used.

Kernel Changes

- New ioctl
- Metadb changes
 - ◆ Name space changes
 - ◆ Version changes

ioctl Changes

- MD_IOCTLGET_NM ioctl handler, getnm_ioctl has been changed to always return a metadvice name
 - ◆ If it is a descriptive name device then return the name in the name record
 - ◆ If it is an old style device then
 - Return the name in the name record if it exists (only underlying metadvicees, top level metadvicees don't have a name record)
 - If a name record doesn't exist then construct a name from the minor number
- Added new ioctl, MD_IOCTLGET_HSP_NM, that will return the name of a hot spare pool

Metadb changes

- Added 2 new record block revisions called MDDB_REV_RBFN & MDDB_REV_RBFN64
 - ◆ The presence of records with these revisions will cause a failure to read the metadbs in older releases.
- All newly created metadevices and hotspare pools will have an associated name record

SVM Descriptive Names