Experience in running relational databases on clustered storage

This content has been downloaded from IOPscience. Please scroll down to see the full text.

2015 J. Phys.: Conf. Ser. 664 042020
(http://iopscience.iop.org/1742-6596/664/4/042020)

View the table of contents for this issue, or go to the journal homepage for more

Download details:

IP Address: 137.138.93.202
This content was downloaded on 09/03/2016 at 08:55

Please note that terms and conditions apply.
Experience in running relational databases on clustered storage

Ruben Gaspar Aparicio, Miroslav Potocky

European Organization for Nuclear Research (CERN), CH-1211, Genève 23, Switzerland

E-mail: ruben.gaspar.aparicio@cern.ch

Abstract. For past eight years, CERN IT Database group has based its backend storage on NAS (Network-Attached Storage) architecture, providing database access via NFS (Network File System) protocol. In last two and half years, our storage has evolved from a scale-up architecture to a scale-out one. This paper describes our setup and a set of functionalities providing key features to other services like Database on Demand [1] or CERN Oracle backup and recovery service. It also outlines possible trend of evolution that, storage for databases could follow.

1. Introduction

Storage service within the CERN database services group provides databases services for two main types of clients. First of all, about 100 Oracle databases ranging from approximately 1 TB up to over 280 TB, with a growth ratio of up to 90 TB per year for the largest one. These databases run usually in clusters ranging from two up to six nodes and may have a set of other technologies involved. For example GoldenGate [2] for long distance replication or Active Data Guard [3] for performance and data protection. Additionally about 200 single instances running a variety of RDBMS systems running mainly MySQL Community Edition version, followed by PostgreSQL and Oracle in the Database on Demand (DBoD) service with a smaller storage footprint of about 100 GB in average per instance. In total the service manages about 5.5 PB of available capacity.

Originally based on SAN (Storage Area Network), the service moved to a NAS architecture about eight years ago after carefully considering pros and cons. The stability and number of advanced functionality provided by the NFS implementation of NetApp and simplification that brings alignment with CERN IT Ethernet based network were key factors in the decision.

NAS storage has been based on a set of HA (High Availability) pairs of mid-range controllers with fast fibre channel disks, 10 krpm (thousand revolution per minute) or 15 krpm. With the arrival of flash cache storage systems, storage shifted to slower SATA (Serial ATA) 7200 rpm disks with a layer of SSD (Solid State Drive) configured to improve random read access. In recent years, there have been a scale-up on controller type and size of the SSD layer, see figure 1.
In the last two and a half years, CERN database storage infrastructure has been moving from running NetApp ONTAP 7-mode to its scale-out version clustered Data ONTAP. Clustered Data ONTAP scales-out traditional clustered storage based, at minimum, on two nodes working in active/active mode. This basic architecture is kept, but it is replicated up to 24 controllers - current maximum number of controllers in a clustered setup. These controllers are interconnected by means of a clustered network for internal data operations like volume move and use management network to allow interaction with the clustered storage. The setup of storage backbone is therefore more complex. On the other hand it brings two important features: data motion [4] can be accomplished transparently with NFS mounts and a unified view of whole storage is achieved via a global namespace [5]. This features are key for rebalancing load on a big cluster e.g. moving hot data file systems from a HDD (Hard Disk Drive) based storage to Flash enabled storage with spare capacity. It can be also used during planned intervention e.g. changing a disk shelf chassis. Such scenario is likely the worst case for a planned intervention as it usually means to swap disks from the impaired disk shelf to the new one once it has been replaced in the rack. This intervention can now be done online thanks to the data motion functionality.

Monitoring and management have been greatly improved by deploying clustered ONTAP in our storage infrastructure. Several instances of DFM (Data Fabric Manager) [6] servers were previously required to administer about 60 controllers on different private networks. Nowadays just one installation of NetApp software OUM (OnCommand Unified Manager) and OPM (OnCommand Performance Manager), new generation of NetApp monitoring software, is required to administer our clusters of 14 and 8 controllers respectively. Consolidation is another key feature provided by clustered Data ONTAP.

2. Network setup

In order to successfully run database services on a NAS backend client data access should be carefully designed. To ensure deterministic network latency a private network is used, see figure 2. Data access from a given client, a database server, is limited to one hop. Use of MTU (Maximum Transmission Unit) 9000 jumbo frames is recommended to avoid unnecessary packet fragmentation and increase throughput. Most RDBMS use a block size of 8 KB for which there is no fragmentation with 9000 bytes Jumbo frames.
Client access to data is provided by two 10gbps links connected to primary and secondary switch. They are trunked in active/passive mode to allow continuous data access in case of primary switch unavailability.

Storage layer provides similar type of redundancy but its design is based on network features of clustered Data ONTAP. Links towards private switches are already trunked providing 20 Gbps. Controller access to disk pools is using SAS-2 (Serial Attached SCSI) loops using dual path and multiple stacks. Usually one per 2 disk shelves or one stack per disk shelves provisioned with SSD drives. In general in our setup a controller would have a maximum throughput of about 3GBps. Failover groups [7] provide protection in case of controller or switch failure, in that case all LIFs (logical interfaces) [8], used to provide connectivity towards clients, are migrated to a different port on a different controller. Additionally in case of a controller failure a takeover is triggered. From that point until the impaired controller is brought back online, there will be no direct access to data previously served by failed controller. It will go from the controller which took over to the one having the LIF via cluster network and then back to client. This setup ensures data accessibility in most scenarios of planned/unplanned interventions.

Public network redundancy is provided by an IP service spanning several switches using HP IRF (Intelligent Resilient Framework) [9] protocol. In that case, public network access can only be warranted for clustered databases e.g. Oracle RAC (Real Application Clusters) [10] where each node in the cluster is connected to a different switch.

3. Flash technology

CERN database services storage relies on flash technology to deliver the level of performance applications require, see figure 3. Initially used was flash cache [11] provided by PAM (Performance Acceleration Module) cards of 512 GB configured to cache only random read IO. PAM cards are installed in a PCI slot on the controller. Despite great performance boost of flash cache, size of PAM is the limiting factor. It also presents certain limitations for cluster operations (takeover/giveback), though some have been overcome in recent releases of ONTAP. Warm-up periods will be still present in case of unplanned takeover. Additionally, despite of PAM configuration to only cache random IO, long running backup operations taking several days to complete flush cached random IO from PAM, affecting database performance.

In latest CERN database services storage setup, flash pool [12] technology is used. When using flash pool cache is located on disk shelves by means of SSD drives. It is aimed at random IO therefore block size should be smaller than 16KB. It has a big advantage over flash cache as it works not only with reads but also with overwrites, data likely to be re-read in the near time. As writes operations in a NetApp controller have latency in order of µs, warm-up period for flash pools is considerably lower if workload...
consists of random writes too. Size can be higher than flash cache allowing more applications to take advantage of this caching mechanism. It has been tested the performance affecting flushing effect while running long backups is minimised with flash pools, though it is still present. Behaviour of flash pool can be tracked per aggregate and per volume using NetApp instrumentation. Despite there is not much literature about them, `wafl_hya_per_vvol` and `wafl_hya_per_aggr` can provide quite a lot of information about how SSD blocks are used per volume and aggregate [5] respectively. In order to drill down what the counters mean, you can use node shell command `stats explain`. Especially interesting are counters about read ahead algorithm used by ONTAP to populate flash cache e.g. `read_rc_ra_hit_blks_rate` and `read_wc_ra_hit_blks_rate`. Processes to retrieve this information have been automated by means of our own scripts. In fact many operations requiring deep knowledge of the storage framework are made easier for our DBA (Database Base Administrators) like automatic auto sizing, retrieving features of a volume e.g. compression or number of snapshots, creating snapshots or doing a restore, getting NFS histogram of the IO being served by a controller, etc. In some cases direct access to CLI (command line interface) is used. Otherwise the NMSDK (NetApp Management Software Development Kit) API is utilised using Perl. Whole idea of this set of scripts is to allow our clients, DBAs or application servers, to access advanced functionality of storage infrastructure either using shell or programmatically.

![Graph](image) **Figure 3.** 'db file sequential read' Oracle wait event that represents time waited to retrieve synchronously a random single block measured using an Oracle benchmark tool.

4. Snapshots

NetApp WAFL (Write Anywhere File Layout) was designed in 1994 as a copy-on-write file system specially optimized to create and efficiently store snapshots. Basic snapshot technology is provided by NetApp as a core functionality, nevertheless to access fast snapshot restoration or other snapshot related functionality like cloning or replication further licenses need to be purchased.

Use cases for snapshots are abundant: schema upgrades, performance testing, backups, recoveries are just some examples. Efficiency on performing these operations at the storage level is essential. CERN database services group storage use cases are mainly related with RDBMS systems. Special care needs to be taken to set target database in a consistent state so the snapshot could be used for later restore in case it is required. See table 1, for how this is achieved on three supported RDBMS systems in our environment.
Table 1. Pre and post steps to trigger a consistent snapshot on major RDBMS systems. SSNAP will hold the name of the snapshot. Oracle12c does not require these steps.

| Pre-step | Post-step |
|----------|-----------|
| mysql> FLUSH TABLES WITH READ LOCK; | mysql> UNLOCK TABLES; |
| mysql> FLUSH LOGS; | |
| Oracle> alter database begin backup; | Oracle> alter database end backup; |
| Postgresql> SELECT pg_start_backup('$SNAP') | Postgresql> SELECT pg_stop_backup(), pg_create_restore_point('$SNAP'); |

Snapshots are taken using storage API we have developed based on NMSKDK API. Functions for taking or restoring previous consistent snapshot are shown in figure 4. CERN databases storage API can be called for any type of application. It is application responsibility to set its state in a consistent mode, flushing dirty blocks to disk.

```
sub SnapCreate {
  my($Server_zapi,Volume_name,Create)=Eg;
  my $this=(caller(0))[3];
  my $output = $Server_zapi->invoke("snapshot-create", "volume", $volume_name, "snapshot", Create);
  if (defined ($RunTime_Zapi::CheckErrorAPIInvoke($output))) {
    my $SR = $output->results_reason();
    LoggerActions::Log("$this::snapshot-create failed: $SR");
    return 0; #error
  } else {
    LoggerActions::Log("$this::Create1!");
    return 1; #ok
  }
}
```

```
sub SnapRestore {
  my($Server_zapi,Volume_name,Restore)=Eg;
  my $this=(caller(0))[3];
  my $output = $Server_zapi->invoke("snapshot-restore-volume", "volume", $volume_name, "snapshot", Restore);
  if (defined ($RunTime_Zapi::CheckErrorAPIInvoke($output))) {
    my $SR = $output->results_reason();
    LoggerActions::Log("$this::snapshot-restore-volume failed: $SR");
    return 0; #error
  } else {
    LoggerActions::Log("$this::Volume <$volume_name> has been restored using snapshot <$restore>!");
    return 1; #ok
  }
}
```

Figure 4. Taking or restoring a snapshot using NMSDK API by the CERN databases storage API.

A snapshot taken on an Oracle database of 25 TB (Atlas database) with about 2.5 TB transaction logs per day takes less than 10 seconds. This is incredibly low compared with the same operation done either using Oracle RMAN (Recovery Manager) or operating system tools to do a cold copy or TSM (Tivoli Storage Manager). Similar time improvement applies for a restore.

For MySQL this is particularly important as some of its storage engines are not transactional e.g. MyISAM. Most backup software needs to acquire a locking block that will freeze the database for a long time if MyISAM schemas are big.

Cloning a volume is equally simple, although for databases, despite of having a consistent snapshot, it will require transactional logs to bring it online. Advantage of cloning using IT-DB API is that same process can be used to clone any RDBMS.

NetApp provides a set of free tools with no support that further integrates snapshot techniques with Oracle databases e.g. RMAN proxy [17] or NetApp cloning plug-in [16].
5. **Advanced features: compression, deduplication and data motion.**

Compression [14] and deduplication [15] are embedded features of NetApp ONTAP systems. Compression can be run online and offline. It was introduced in ONTAP 8.0.1 and it does not require license since ONTAP 8.1. Deduplication on the other hand can only run offline. Deduplication and offline compression use the same schedule, ONTAP first compresses and then tries deduplication.

IT-DB Oracle backup to disk infrastructure strongly benefits from this feature. The system was designed from the very beginning to use compression so disk space was projected with compression savings accounted for. Nowadays it is using 641 TB with compression saving of 682 TB. This amounts to approximately 51.5% space saving (see figure 5). In figure 6 a comparison of NetApp compression with different levels of advanced compression for Oracle 12.1.0.1 backupsets is presented.

![Figure 5](image_url). As shown by NetApp Management Console compression ratio can be deducted from the difference between incoming network traffic and data written to disk. Throughput view from a backup to disk controller.

| Data Type          | Incompressible (T) | Basic | Low   | Medium | High   | No-compressed-fs | Inline-compression NetApp & NFS |
|--------------------|--------------------|-------|-------|--------|--------|------------------|--------------------------------|
| Raw data           | 336GB (deeb11 upgraded to 12c) | 48.8GB (1.9GB) | 64.13GB (22G) | 62.96GB (48G) | 34.17GB (5x17G) | 252.8GB (22G) | 90GB (20G) |
| Percentage saved   | 82.1%              | 74.6% | 79%   | 98.4%  | 0%     | 0%               | 64.5%                          |

**Figure 6.** Comparison of different levels of Oracle compression for backupsets with NetApp.

Deduplication works at WAFL block size of 4KB. This size is a disadvantage in a database environment like Oracle where logical block size is usually 8KB with a header and footer usually making it unique. Nevertheless deduplication is used in our backup to disk infrastructure as it presents big savings in databases where the control file is big. This control file is stored on average 100 times per day in our environment based on actual Oracle recommended configuration.

Another use case where compression is being used is compression of big read-only “tablespaces” like the one of Oracle Accelerator Logging database, where savings of about 50% are reported. Compression happens completely transparently from the application layer. In such a huge database, 289TB nowadays, this is a great gain of space. With controllers having enough CPU capacity, no impact on performance has been noticed at application level.

Data motion command `vol move` is a key feature of clustered ONTAP. It is based on SnapMirror [18]. In combination with cluster networking and global namespace it allows moving volume between all controllers in a cluster provided there is enough space available to execute the move. This brings a useful flexibility while designing the system since volume can be moved if its data growth is unexpected, or its performance requirements change, etc. This feature has been successfully used a number of times.
Especially during replacement of disk shelf chassis, when all volumes of affected storage pool need to be evacuated to avoid data service interruption. For volumes with high number of write operations per second, special care must be taken in the cutover phase and tuning is possible on the command via some cutover attributes.

6. NFS mount options and special NFS case

As access to our databases is provided using NFS version 3 protocol special care must be taken when setting mount options. There is an extensive documentation on this topic for MySQL and Oracle [19]. However there is almost no related documentation for PostgreSQL as traditionally NFS is not commonly considered as a storage access protocol for PostgreSQL by its community. Nevertheless with the right setup, PostgreSQL runs without problems on NFS. Following are mount options we use for our single instances of MySQL and PostgreSQL:

```
--WAL or BINLOGS & DATA
controller:/ORA/dbs0X/CPGSQL on /ORA/dbs0X/CPGSQL type nfs
(rw,noatime,bg,hard,nointr,tcp,vers=3,timeo=600,rsize=65536,wsize=65536)
```

Mount options can greatly impact performance for particular operations of database as it occurred on a PostgreSQL replication setup [23]. Impact should be estimated carefully.

When running MySQL instances under a cluster software like Oracle RAC [10], in case server crashes, locks are retained by related NFS server so a spare server will not be able to open the instance. To resolve this situation, before attempting to restart the instance, locks need to be cleaned up using clustered ONTAP vserver locks break command [21].

7. Conclusion

After more than two and a half year of running storage infrastructure on clustered Data ONTAP, CERN database services group has a positive experience. Storage clusters have shown to be reliable in a number of planned and unplanned interventions. NetApp flash pool caching technology is providing expected performance and good relationship with vendors have proven to be very valuable to solve critical issues. In the consolidation phase, moving from 7-mode to C-mode has allowed us to reuse a lot of components from our old systems: disk shelves, 10gbps network cards, PAM cards, SAS cables, etc. The infrastructure has been also used to offer iSCSI data access under CINDER OpenStack service.

Future of storage for databases seems oriented on all flash arrays technologies. Emerging companies and traditional storage vendors are now competing in this new market as flash technology prices continues to drop. Evolution of SSD technologies using MLC (Multi-level Cell) currently available as TLC (Triple level cell), should allow higher density at more affordable price. Major difference between vendors is based on their architecture scale-out vs scale-up and core functionality features like compression, deduplication, cloning, snapshots and thin provisioning. Another aspect is whether these features are embedded in appliance (e.g. Pure Storage or EMC XtremIO), provided as add-ons (e.g. Violin memory), or sold as exclusive features (e.g. HP and IBM) [22]. Understanding and following trends in this new technology and possibly adapting IT infrastructure with new networking or servers with different internal buses e.g. PCIe SSD and access protocols like NVM Express allows sustaining massive IOPS SSDs can deliver.
Acknowledgments
To CERN Database services colleagues involved in operations and utilization of our storage solution, former colleague Azzurra Chinzer, our encouraging section leader Eva Dafonte and NetApp support engineers specially: Nicolas Jacquot and Jeffrey Steiner.

References
[1] R Gaspar Aparicio, I Coterillo Coz et al 2012 J. Phys.: Conf. Ser. 396 052034
[2] Oracle GoldenGate, http://www.oracle.com/technetwork/middleware/goldengate/overview/index.html
[3] Oracle Active Data Guard, http://www.oracle.com/technetwork/database/availability/active-data-guard-wp-12c-1896127.pdf
[4] Data Motion, http://www.netapp.com/in/products/platform-os/datamotion.aspx
[5] Global namespace, https://library.netapp.com/ecmdocs/ECMP1196891/html/GUID-31CE89BB-9A85-43A4-BBC1-28497CC0DFD3.html
[6] Data Fabric Manager, https://library.netapp.com/ecm/ecm_download_file/ECMP1122473
[7] Failover groups, https://library.netapp.com/ecmdocs/ECMP1196907/html/GUID-94D4F8DD-3437-4474-81A6-9DB6FF6A244D.html
[8] Logical interfaces, https://kb.netapp.com/support/index?page=content&id=1013689&pmv=print&impressions=false
[9] HP IRF, http://h17007.www1.hp.com/docs/reports/irf.pdf
[10] Oracle RAC, http://www.oracle.com/technetwork/database/options/clustering/overview/index.html
[11] Flash cache technology best practices, http://www.netapp.com/us/system/pdf-reader.aspx?m=tr-3832.pdf
[12] Flash pool design and implementation guide, http://www.netapp.com/us/system/pdf-reader.aspx?m=tr-4070.pdf
[13] WAFL, http://community.netapp.com/fukiw75442/attachments/fukiw75442/data-ontap-discussions/23341/WAFL.pdf
[14] NetApp ONTAP compression basics, http://community.netapp.com/t5/Tech-OnTap-Articles/Back-to-Basics-Data-Compression/ta-p/86691
[15] NetApp ONTAP deduplication basics, http://community.netapp.com/t5/Tech-OnTap-Articles/Back-to-Basics-Deduplication/ta-p/84888
[16] NetApp cloning plug-in, http://www.netapp.com/us/system/pdf-reader.aspx?pdfuri=tcm:10-119669-16&m=tr-4266.pdf
[17] NetApp RMAN Proxy Copy Media Manager Library, http://community.netapp.com/fukiw75442/attachments/fukiw75442/fas-data-ontap-and-related-plugins-articles-and-resources/1291/NetApp+for+RMAN+Install+and+Testing+Guide+V2+-+11g.pdf
[18] NetApp SnapMirror, http://community.netapp.com/t5/Tech-OnTap-Articles/Back-to-Basics-SnapMirror/ta-p/86296
[19] My Oracle Support, Mount Options for Oracle files when used with NFS on NAS devices (Doc ID 359515.1) & Best Practices for Oracle Databases on NetApp Storage, http://www.netapp.com/us/system/pdf-reader.aspx?m=tr-3633.pdf&cc=us
[20] PostgreSQL slave setup issue due to NFS mount options, http://www.postgresql.org/message-id/6FCBD86BDC789C43AC696969EC34022801093E6947@CERNXCHG44.cern.ch
[21] My Oracle Support, InnoDB: Unable to lock ./ibdata1, error: 11" Error Sometimes Seen With MySQL on NFS (Doc ID 1522745.1)
[22] TechTarget white paper, Features to Weigh When Choosing an All-Flash Array. TechTarget white paper, PCIe Solid-State Storage: What To Know Before Getting Started