# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>EXECUTIVE SUMMARY</td>
<td>2</td>
</tr>
<tr>
<td>PURE STORAGE INTRODUCTION</td>
<td>3</td>
</tr>
<tr>
<td>FlashArray//M R2 Specifications</td>
<td>5</td>
</tr>
<tr>
<td>Purity Operating Environment</td>
<td>5</td>
</tr>
<tr>
<td>Pure1</td>
<td>6</td>
</tr>
<tr>
<td>Experience Evergreen Storage</td>
<td>6</td>
</tr>
<tr>
<td>Goals and Objectives</td>
<td>7</td>
</tr>
<tr>
<td>Audience</td>
<td>7</td>
</tr>
<tr>
<td>Operating System</td>
<td>7</td>
</tr>
<tr>
<td>SAP HANA Shared File System</td>
<td>8</td>
</tr>
<tr>
<td>Solution Architecture (Suggested)</td>
<td>9</td>
</tr>
<tr>
<td>Network Configuration</td>
<td>10</td>
</tr>
<tr>
<td>Pure Storage Configuration</td>
<td>12</td>
</tr>
<tr>
<td>Cluster Setup</td>
<td>13</td>
</tr>
<tr>
<td>OCFS2 Setup</td>
<td>19</td>
</tr>
<tr>
<td>Configuration of Udev Rules</td>
<td>22</td>
</tr>
<tr>
<td>Multipathing Configuration</td>
<td>23</td>
</tr>
<tr>
<td>Hana Configuration Files</td>
<td>24</td>
</tr>
<tr>
<td>Conclusion</td>
<td>26</td>
</tr>
<tr>
<td>References</td>
<td>27</td>
</tr>
</tbody>
</table>
EXECUTIVE SUMMARY

The SAP® HANA® database takes advantage of main memory (RAM) and the data processing capabilities of multicore processors to deliver better performance for analytical and transactional applications. SAP HANA offers a multiengine query processing environment that allows it to support relational data (with both row- and column-oriented physical representations in a hybrid engine) as well as graph and text processing for semi-structured and unstructured data management within the same system. The SAP HANA database is 100 percent compliant with atomicity, consistency, isolation, and durability (ACID) requirements.

For more information about SAP HANA, go to the SAP help portal: http://help.sap.com/hana/.

The performance of the underlying infrastructure, and in particular storage performance, is the key issue that can hold back a HANA project. Traditional disk-based storage cannot meet the performance needs of HANA and is inherently too complicated to manage. The Pure Storage® FlashArray//M R2 not only delivers the required performance, but also the resiliency and management simplicity required by SAP HANA environments – at a cost this is often lower that of disk-based and hybrid arrays.

Specifically, enterprises running SAP HANA on the certified Pure Storage platform can benefit in the following ways:

- High performance storage with sub-millisecond latency
- Highly efficient data reduction (inline compression and de-duplication)
- 6 9’s high availability with no single points of failure
- Simplified management with rapid provisioning and replication capabilities

There are two ways to implement SAP HANA:

- An “Appliance” option where customers can purchase pre-configured systems for their HANA implementation. These configurations often tend to be less flexible as they come with fixed server, storage, and network components.
- The SAP HANA Tailored Datacenter Integration (TDI) option, where customers can choose their server, storage, and network components as long as they are all certified by SAP. Many customers prefer this option, as they can leverage existing server, storage, and networking infrastructure for their SAP HANA implementation.

The Pure Storage FlashArray//m R2 series has been validated as a certified SAP HANA hardware platform as per SAP’s "Certified Enterprise Storage: HANA-HWC-ES 1.1" program. Customers can use the Pure Storage FlashArray//M R2 series as a part of their SAP HANA TDI implementation.
PURE STORAGE INTRODUCTION

Who knew that moving to all-flash storage could help reduce the cost of IT? FlashArray//M R2 makes server and workload investments more productive, while also lowering storage spend. With FlashArray//M R2, organizations can dramatically reduce the complexity of storage to make IT more agile and efficient, accelerating their journey to the cloud.

The performance of FlashArray//M R2 can also make your business smarter by unleashing the power of real-time analytics, driving customer loyalty, and creating new, innovative customer experiences that simply weren’t possible with disk.

FlashArray//M R2 enables you to transform your data center, cloud, or entire business with an affordable all-flash array capable of consolidating and accelerating all your key applications.

**MINI SIZE** – Reduce power, space and complexity by 90%
- 3U base chassis with 15-1500+ TBs usable
- ~1kW of power
- 6 cables

**MIGHTY PERFORMANCE** – Transform your datacenter, cloud, or entire business
- Up to 370,000 32K IOPS
- Up to 11.5 GB/s bandwidth
- <1ms average latency

**MODULAR SCALE** – Scale FlashArray//M R2 inside and outside of the chassis for generations
- Expandable to 1.5 PB usable via expansion shelves
- Upgrade controllers and drives to expand performance and/or capacity

**MEANINGFUL SIMPLICITY** – Appliance-like deployment with worry-free operations
- Plug-and-go deployment that takes minutes, not days
- Non-disruptive upgrades and hot-swap everything
- Fewer parts = more reliability

The FlashArray//m R2 expands upon the FlashArray’s modular, stateless architecture, designed to enable expandability and upgradability for generations. The FlashArray//m R2 leverages a chassis-based design with customizable modules, enabling both capacity and performance to be independently improved over time with advances in compute and flash, to meet your business’ needs today and tomorrow.
THE PURE STORAGE FLASHARRAY IS IDEAL FOR:

Accelerating Databases and Applications – Speed transactions by 10x with consistent low latency, enable online data analytics across wide datasets, and mix production, analytics, dev/test, and backup workloads without fear.

Virtualizing and Consolidating Workloads – Easily accommodate the most IO-hungry Tier 1 workloads, increase consolidation rates (thereby reducing servers), simplify VI administration, and accelerate common administrative tasks.

Delivering the Ultimate Virtual Desktop Experience – Support demanding users with better performance than physical desktops, scale without disruption from pilot to >1000s of users, and experience all-flash performance for under $100/desktop.

Protecting and Recovering Vital Data Assets – Provide always-on protection for business-critical data, maintain performance even under failure conditions, and recover instantly with FlashRecover.

PURE STORAGE FLASHARRAY SETS THE BENCHMARK FOR ALL-FLASH ENTERPRISE STORAGE ARRAYS. IT DELIVERS:

Consistent Performance – FlashArray delivers consistent <1ms average latency. Performance is optimized for real-world applications workloads that are dominated by I/O sizes of 32K or larger vs. 4K/8K hero performance benchmarks. Full performance is maintained even under failures/updates.

Lower Cost than Disk – Inline de-duplication and compression deliver 5 – 10x space savings across a broad set of I/O workloads including databases, virtual machines and virtual desktop infrastructure.

Mission-Critical Resiliency – FlashArray has proven >99.9999% delivered availability, as measured across the Pure Storage installed base, inclusive of non-disruptive everything and without performance impact.

Disaster Recovery Built-In – FlashArray offers native, fully-integrated, data reduction-optimized backup and disaster recovery at no additional cost. Setup disaster recovery with policy-based automation within minutes. And recover instantly from local, space-efficient snapshots or remote replicas.

Simplicity Built-In – FlashArray offers game-changing management simplicity that makes storage installation, configuration, provisioning, and migration a snap. No more managing performance, RAID, tiers, or caching. Achieve optimal application performance without any tuning at any layer. Manage the FlashArray the way you like it: web-based GUI, CLI, VMware® vCenter, Rest API, or OpenStack.
FLASHARRAY/M R2 TECHNICAL SPECIFICATIONS

<table>
<thead>
<tr>
<th></th>
<th>//M10*</th>
<th>//M20*</th>
<th>//M50*</th>
<th>//M70*</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Capacity</strong></td>
<td>Up to 25TBs</td>
<td>Up to 250TBs</td>
<td>Up to 500+TBs</td>
<td>Up to 1.5PBs</td>
</tr>
<tr>
<td></td>
<td>effective capacity*</td>
<td>effective capacity*</td>
<td>effective capacity*</td>
<td>effective capacity*</td>
</tr>
<tr>
<td></td>
<td>5 – 10TBs raw capacity</td>
<td>5 – 80TBs raw capacity</td>
<td>20 – 176TBs raw capacity</td>
<td>42 – 512TBs raw capacity</td>
</tr>
<tr>
<td><strong>All-Inclusive</strong></td>
<td>Up to 100,000 32K IOPS'</td>
<td>Up to 200,000 32K IOPS'</td>
<td>Up to 270,000 32K IOPS'</td>
<td>Up to 370,000 32K IOPS'</td>
</tr>
<tr>
<td><strong>Performance</strong></td>
<td>&lt;1ms average latency</td>
<td>&lt;1ms average latency</td>
<td>&lt;1ms average latency</td>
<td>&lt;1ms average latency</td>
</tr>
<tr>
<td></td>
<td>Up to 3 GB/s bandwidth**</td>
<td>Up to 6 GB/s bandwidth**</td>
<td>Up to 9 GB/s bandwidth**</td>
<td>Up to 11.5 GB/s bandwidth**</td>
</tr>
<tr>
<td><strong>Connectivity</strong></td>
<td>16 Gb/s Fibre Channel</td>
<td>16 Gb/s Fibre Channel</td>
<td>16 Gb/s Fibre Channel</td>
<td>16 Gb/s Fibre Channel</td>
</tr>
<tr>
<td></td>
<td>10 Gb/s Ethernet iSCSI</td>
<td>10 Gb/s Ethernet iSCSI</td>
<td>10 Gb/s Ethernet iSCSI</td>
<td>10 Gb/s Ethernet iSCSI</td>
</tr>
<tr>
<td></td>
<td>1 Gb/s Management &amp;</td>
<td>1 Gb/s Management &amp;</td>
<td>1 Gb/s Management &amp;</td>
<td>1 Gb/s Management &amp;</td>
</tr>
<tr>
<td></td>
<td>Replication ports</td>
<td>Replication ports</td>
<td>Replication ports</td>
<td>Replication ports</td>
</tr>
<tr>
<td><strong>Physical</strong></td>
<td>3U</td>
<td>3U – 5U</td>
<td>3U – 5U</td>
<td>3U – 5U</td>
</tr>
<tr>
<td></td>
<td>95 lbs (43.1 kg)</td>
<td>95 lbs (43.1 kg)</td>
<td>95 lbs (43.1 kg)</td>
<td>97 lbs (44.0 kg)</td>
</tr>
<tr>
<td></td>
<td>5.12” x 18.94” x 29.72”</td>
<td>fully loaded</td>
<td>fully loaded</td>
<td>fully loaded</td>
</tr>
<tr>
<td></td>
<td>chassis</td>
<td>chassis</td>
<td>chassis</td>
<td>chassis</td>
</tr>
</tbody>
</table>

* Stated specifications are applicable to //M R2 versions.
** Effective capacity assumes HA, RAID, and metadata overhead, GB-to-GB conversion, and includes the benefit of data reduction with always-on inline deduplication, compression, and pattern removal. Average data reduction is calculated at 5-to-1.
*** Includes always-on RAID-3D, deduplication, compression, and encryption.
† Why does Pure Storage quote 32K, not 4K IOPS? The industry commonly markets 4K IOPS benchmarks to inflate performance numbers, but multiple real world workloads consolidating on a single array average closer to 32K. FlashArray adapts automatically to 512B – 32KB IO for superior performance, scalability, and data reduction. IOPS are based on 100% 32KB reads.
†† Data throughput is based on 100% 32KiB reads.

PURITY OPERATING ENVIRONMENT

Purity implements advanced data reduction, storage management, and flash management features. All features of Purity are included in the base cost of the FlashArray//M R2.

**Storage Software Built for Flash** – FlashCare technology virtualizes the entire pool of flash within the FlashArray, and allows Purity to both extend the life and ensure the maximum performance of consumer-grade MLC flash.
Granular and Adaptive – Purity Core is based upon a 512-byte variable block size metadata layer. This fine-grain metadata enables all of Purity’s data and flash management services to operate at the highest efficiency.

Best Data Reduction Available – FlashReduce implements five forms of inline and post-process data reduction to offer the most complete data reduction in the industry. Data reduction operates at a 512-byte aligned variable block size, to enable effective reduction across a wide range of mixed workloads without tuning.

Highly Available and Resilient – FlashProtect implements high availability, dual-parity RAID-3D, non-disruptive upgrades, and encryption, all of which are designed to deliver full performance to the FlashArray during any failure or maintenance event.

Backup and Disaster Recovery Built In – FlashRecover combines space-saving snapshots, replication, and protection policies into an end-to-end data protection and recovery solution that protects data against loss locally and globally. All FlashProtect services are fully-integrated in the FlashArray and leverage native data reduction capabilities.

PURE1®

Pure1 Manage – By combining local web-based management with cloud-based monitoring, Pure1 Manage allows you to manage your FlashArray wherever you are – with just a web browser.

Pure1 Connect – A rich set of APIs, plugin-is, application connectors, and automation toolkits enable you to connect FlashArray//M R2 to all your data center and cloud monitoring, management, and orchestration tools.

Pure1 Support – FlashArray//M R2 is constantly cloud-connected, enabling Pure Storage to deliver the most proactive support experience possible. Highly trained staff combined with big data analytics help resolve problems before they start.

Pure1 Collaborate – Extend your development and support experience online, leveraging the Pure1 Collaborate community to get peer-based support, and to share tips, tricks, and scripts.

EXPERIENCE EVERGREEN™ STORAGE

Get storage that behaves like SaaS and the cloud. Deploy it once and keep expanding and improving performance, capacity, density and/or features for 10 years or more – without downtime, performance impact, or data migrations. Our “Right Size” capacity guarantee ensures you get started knowing you’ll have the effective capacity you need. And our Capacity Consolidation program keeps your storage modern and dense as you expand. With Evergreen Storage, you’ll never re-buy a TB you already own.
GOALS AND OBJECTIVES

The purpose of this document is to describe a test plan and best practices for deploying a high performance SAP HANA environment with a Pure Storage FlashArray. This document will serve a guide for customers going through the process of configuring HANA clusters, underlying file systems, and storage area networks for HANA.

AUDIENCE

This document is intended for Database Administrators (DBAs), Storage Administrators, System Administrators, and anybody who wants to build an SAP HANA environment on a Pure Storage FlashArray with best results. This is a technical paper that assumes familiarity with SAP HANA administration as well as basic Linux system and storage administration tasks as required for a typical database installation, such as setting up file systems, LUN configuration, etc.

OPERATING SYSTEM

SAP HANA supports both Red Hat® Enterprise Linux (RHEL) and SUSE® Enterprise Linux. SAP HANA initially was only supported to run on SUSE Enterprise Linux. Red Hat support for SAP HANA came in mid 2014. Here are the master SAP Notes for SUSE Linux and RHEL:

- SAP Note 1944799 – SAP HANA Guidelines for SLES Operating System
- SAP Note 2009879 – SAP HANA Guidelines for Red Hat Enterprise Linux (RHEL) Operating System
SAP HANA SHARED FILE SYSTEM

SAP HANA scale-out implementation the /hana/shared or sapmnt or installation path is the mount directory which is used for has shared files and binaries which needs to be accessed by all the hosts.

Some of the following file system recommendations for /hana/shared can be used:

- **NFS on RHEL Operating System (NFSv3, NFSv4)**
  
  A Network File System (NFS) allows remote hosts to mount file systems over a network and interact with those file systems as though they are mounted locally. This enables /hana/shared to consolidate/share resources by SAP HANA nodes on the network. SAP HANA supports NFSv3 and NFSv4 for RHEL.

  NFS version 3 (NFSv3) supports safe asynchronous writes. It also supports 64-bit file sizes and offsets, allowing clients to access more than 2 GB of file data. NFSv4 works through firewalls and on the Internet, no longer requires an rpcbind service, supports ACLs, and utilizes stateful operations.

  RHEL also supports NFS version 4 which provides a number of performance and security enhancements.

- **GPFS**

  GPFS provides high performance by allowing data to be accessed over multiple computers at once. Most existing file systems are designed for a single server environment, and adding more file servers does not improve performance. GPFS provides higher input/output performance by "striping" blocks of data from individual files over multiple disks, and reading and writing these blocks in parallel.

- **OCFS2 on SUSE Linux Operating System**

  Oracle® Cluster File System 2 (OCFS2) is a general-purpose journaling file system that is fully integrated in the Linux 2.6 kernel and later. OCFS2 allows you to store application binary files, data files, and databases on devices in a SAN. All nodes in a cluster have concurrent read and write access to the file system. A distributed lock manager helps prevent file access conflicts. OCFS2 supports up to 32,000 subdirectories and millions of files in each directory. The O2CB cluster service (a driver) runs on each node to manage the cluster.

For SAP HANA Data and Log volumes are only certified to use the XFS file system.
SOLUTION ARCHITECTURE
(SUGGESTED)

The SAP HANA SPS11 certification infrastructure uses a 2+1 cluster configuration. Three Cisco UCS® B200 M4 blades are used with SUSE 12 SP1 OS, including High Availability Option and OCFS to create the environment.

The Pure Storage FlashArray used in this configuration is the FlashArray//M10. There are four 16GB Fibre channel ports from the array attached to the SAN Fabric. Each node has two Fibre channel ports for storage and four 10 GBPS ports for network.

All the servers boot from SAN with dedicated drive for their OS-related partitions. All other LUNs are shared across the nodes. The /hana/shared partition is used for storing SAP binaries and is mounted in parallel simultaneously.

HANA clustering controls the DATA and LOG LUNs that are needed by the master and worker nodes. These are mounted in the appropriate node with the role of master or worker.

Figure 1: Pure Storage HANA Certification setup
The SAP HANA system communicates using different network channels. It is recommended practice to have a well-defined network topology to control and limit network access to SAP HANA to only those communication channels required for your scenario, and to apply appropriate additional security and performance measures as necessary.

Clients require access to the SAP HANA database, and while SAP HANA is an in-memory database which stores and processes its data in memory, the data is also saved in persistent storage locations to provide protection against data loss. In distributed (scale-out) systems, communication must be possible between hosts; in disaster recovery setups, data is replicated over the network to other data centers.

The components belonging to SAP HANA communicate over several logical network zones:
Figure 2.1: Logical architecture of the Pure Storage HANA setup (Network configuration)

- **Client Zone** – Different clients, such as SQL clients on SAP application servers, browser applications using HTTP/S to the SAP HANA XS server, and other data sources (such as BI) need a network communication channel to the SAP HANA database.

- **Inter-Node Zone** – The internal zone covers the communication between hosts in a distributed SAP HANA system as well as the communication used by SAP HANA system replication between two SAP HANA sites.

- **Storage/Cluster/OCFS2 Zone** – Although SAP HANA holds the bulk of its data in memory, the data is also saved in persistent storage locations – which probably need to be accessed via a network – to provide protection against power failures or other events that cause hosts to become unavailable. In addition, this zone will be used for cluster communication in our recommended architecture for the cluster file system OCFS2.

To achieve this, it is recommended to have three Ethernet NICs in each node which will have different hostnames and will be part of a different network zone.

The `/etc/hosts` file should appear as shown below:

```
10.11.10.15 ext saphana1 saphana1
10.11.10.16 saphana1 saphana1
10.11.10.17 saphana1 saphana1
10.11.10.18 ext saphana1-int saphana1-int
10.11.10.19 ext saphana1-int saphana1-int
```

Install the SAP HANA scale out in the following way:

- Set up cluster and OCFS2 with one hostname that uses a private network
- Set up SAP HANA system with a different hostname that uses a private network
• Set up SAP HANA clients with a third hostname that uses a public network.

The global.ini should be configured for inter-node communication or internal communication. Please see the latter part of the document for global.ini configuration.

**PURE STORAGE CONFIGURATION**

The Pure Storage array FlashArray//M10 R2 is used in this configuration. The raw capacity of the array is 10 TB, but with Pure Storage’s industry-leading de-duplication and compression capabilities the usable capacity is significantly larger.

![Figure 3: Pure Storage Array System Dashboard](image)

The three Cisco UCS servers used for the HANA configuration are attached to the SAN fabric and zoned to the array. The three nodes are seen in the array as shown below.

![Figure 4: Hosts zoned to the array](image)

**LUN CONFIGURATION**

Multiple LUNs are configured in the Pure Storage array for use in the HANA setup. There are three boot LUNS, as shown, that are zoned to the individual hosts. All other LUNS are shared between the three nodes. The data and log LUNS are used by the HANA
database. The saphanasharedvolume LUN is used for the OCFS2 partition that is always mounted on all the nodes simultaneously and houses the HANA binaries.

The quorum device is also a shared LUN (sbd_lun) that is used for fencing hosts in the event of contention in the cluster.

Figure 5: Pure Storage LUNS used in the setup

CLUSTER SETUP

• Enabled SSH for password-less access on all three nodes.
• Installed SLES High Availability components:
  – Some of the high availability components are directly available by YaST.
  – The other components are installed by downloading the two ISO images representing the HA SW.

Invoke YAST and install additional third party components. Choose the ISO images as the source of the software. Cluster components such as Corosync, Pacemaker, etc., are installed.

Install the SLES HA software packages with YaST.
Alternatively, you can install them from the command line with zypper:

```
saphananodel:~ # zypper in -t pattern ha_sles
Loading repository data...
```

Create sbd on a separate lun using the create sdb on a shared lun.

```
saphananodel/dev/mapper # sdb -d /dev/mapper/362a99707d76780c0df17b000011097
Creating version 2.1 header on device 3 (uuid: 0d0b3d4c-845ad-4a8b-27d3079838b7)
Initialising 25% write on device 3.
saphananodel/dev/mapper
```

Define the first communication channel (Unicast)

To use unicast go to YaST → Network

a. Set the Transport protocol to Unicast.

b. Define the Bind Network Address. Set the value to the subnet you will use for cluster unicast.

c. Define the Multicast Port.

d. For unicast communication, Corosync needs to know the IP addresses of all nodes in the cluster. For each node that will be part of the cluster, click Add and enter the following details:
   - IP Address
– Redundant IP Address (only required if you use a second communication channel in Corosync)

![YaST Network Communication channels view](image)

Figure 7: YaST Network → Communication channels view

Then Run the ha-cluster-init command and choose communication modes and other parameters in the first node. The stonith SBD device is automatically recognized and leveraged as the quorum device for the cluster.
Figure 8: SUSE HA Init

Now you will have one-node cluster. Check the cluster status with crm status:

```
saphananode:~ # crm status
Last change: Mon Oct 24 13:58:30 2016 by root via cibadmin on saphananode
Type: corosync
Current DC: saphananode (version 1.1.14-18.4-4f22ad7) - partition WITHOUT quorum
1 node and 1 resource configured
Online: [ saphananode ]
Full list of resources:
   saphananode  (saphananode-external)/sd: started saphananode
```

**CONFIGURING CSYNC2 WITH YAST**

- In the YaST cluster module, switch to the Csync2 category.
- To specify the synchronization group, click Add in the Sync Host group and enter the local host names of all nodes in your cluster. For each node, you must use exactly the strings that are returned by the hostname command.
- In case host name resolution does not work properly in your network, you can also specify a combination of host name and IP address for each cluster node. To do so,
use the string HOSTNAME@IP such as alice@192.168.2.100, for example. Csync2 will then use the IP addresses when connecting.

• Click Generate Pre-Shared-Keys to create a key file for the synchronization group. The key file is written to /etc/csync2/key_hagroup. After it has been created, it must be copied manually to all members of the cluster.

• To populate the Sync File list with the files that usually need to be synchronized among all nodes, click Add Suggested Files.

• Activate Csync2 by clicking Turn Csync2 ON. This will execute the following command to start Csync2 automatically at boot time:

  root # systemctl enable csync2.socket

• Reboot the nodes ... then the next step is to use csync2 to transfer the config files across the nodes.

Figure 9: YaST Cluster→Network→ Configure Csync2
SYNCHRONIZING THE CONFIGURATION FILES WITH CSYNC2

To successfully synchronize the files with Csync2, make sure that the following prerequisites are met:

- The same Csync2 configuration is available on all nodes. Copy the file `/etc/csync2/csync2.cfg` manually to all nodes after you have configured it as described in Procedure 3.8, “Configuring Csync2 with YaST”. We recommend you include this file in the list of files to be synchronized with Csync2.
- Copy the file `/etc/csync2/key_hagroup` you have generated on one node in Step 3 to all nodes in the cluster. It is needed for authentication by Csync2. However, do not regenerate the file on the other nodes as it needs to be the same file on all nodes.
- Csync2 must be running on all nodes.

**Note: Starting Services at Boot Time**

Execute the following command on all nodes to make the service start automatically at boot time:

```
root # systemctl enable csync2.socket
```

- On the node that you want to copy the configuration from, execute the following command:

```
root # csync2 -xv
```

  - This will synchronize all the files once by pushing them to the other nodes. If all files are synchronized successfully, Csync2 will finish with no errors.
  - If one or more files that are to be synchronized have been modified on other nodes (not only on the current one), Csync2 will report a conflict. You will get an output similar to the one below.
  - If you are sure that the file version on the current node is the “best” one, you can resolve the conflict by forcing this file and resynchronizing:

```
root # csync2 -f /etc/corosync/corosync.conf csync2 -x
```

Then we need to run the following bootstrap script for the other nodes to join this cluster.

- Run `ha-cluster-join` in the other two nodes.

If you decide to continue anyway, you will be prompted for the IP address of an existing node. Enter the IP address.
Figure 10: Cluster join from nodes

CLUSTER STATUS AFTER SETUP

CRM status after two joins, which are on the other two nodes.

OCFS2 SETUP

Oracle Cluster File System 2 (OCFS2) is a general-purpose journaling file system that has been fully integrated since the Linux 2.6 Kernel. OCFS2 allows you to store application binary files, data files, and databases on devices on shared storage. All nodes in a cluster have concurrent read and write access to the file system. A user-space control daemon, managed via a clone resource, provides the integration with the HA stack, in particular with Corosync and the Distributed Lock Manager (DLM).

The OCFS2 Kernel module (ocfs2) is installed automatically in the High Availability Extension on SUSE® Linux Enterprise Server 12 SP1.
OCFS STEPS

- Configure a DLM clone
- Format SAN volume with an OCFS2 file system
- Configure an OCFS2 volume clone cluster resource

CONFIGURE OCFS2

A prerequisite for an OCFS2 volume, you must configure the following resources as services in the cluster: DLM. OCFS2 uses the cluster membership services from Pacemaker which run in user space. Therefore, DLM needs to be configured as a clone resource that is present on each node in the cluster.

Follow the steps below for one node in the cluster:

- Run crm configure.

  Enter the following to create the primitive resource for DLM:

  ```bash
  crm(live)configure# primitive dlm ocf:pacemaker:controld 
  op monitor interval="60" timeout="60"
  ```

- Create a base-group for the DLM resource. As further cloned primitives are created, it will be added to this group.

  ```bash
  crm(live)configure# group base-group dlm
  ```

- Clone the base-group so that it runs on all nodes.

  ```bash
  crm(live)configure# clone base-clone base-group 
  meta interleave=true target-role=Started
  ```

Then commit. It should look as shown below.

---

Figure 11: crm configure for DLM
FORMAT SAN VOLUME WITH AN OCFS2 FILE SYSTEM

- Create and format the volume using the mkfs.ocfs2 utility. For information about the syntax for this command, refer to the mkfs.ocfs2 man page.
- Below is the command to create an ocfs2 for a 3-node cluster.

```
root # mkfs.ocfs2 -N 3 /dev/mapper/<wwid>
```

CONFIGURE AN OCFS2 VOLUME CLONE CLUSTER RESOURCE

To mount an OCFS2 volume with the High Availability software, configure an ocfs2 file system resource in the cluster. The following procedure uses the crm shell to configure the cluster resources. Start a shell and log in as root or equivalent. We need to make the directory for `/hana/shared` to mount this ocfs2 volume.

- Run `crm configure`.
- Configure Pacemaker to mount the OCFS2 file system on every node in the cluster:

  ```
  crm(live)configure# primitive ocfs2-1 ocf:heartbeat:Filesystem params device="/dev/sdb1" directory="/hana/shared" fstype="ocfs2" options="acl" op monitor interval="20" timeout="40"
  ```
- Add the `ocfs2-1` primitive to the base-group you created in Procedure 15.2, “Configuring a DLM Resource”.

  ```
  crm(live)configure# modgroup base-group add ocfs2-1
  ```
This should look like the following:

```
99-PURE-STORAGE.RULES
# Recommended settings for Pure Storage FlashArray.
# Use noop scheduler for high-performance solid-state storage
ACTION=="add|change", KERNEL=="sd[!0-9]", SUBSYSTEM=="block", ENV[ID_VENDOR]=="PURE", ATTR(queue/scheduler)=="noop"

# Reduce CPU overhead due to entropy collection
ACTION=="add|change", KERNEL=="sd[!0-9]", SUBSYSTEM=="block", ENV[ID_VENDOR]=="PURE", ATTR(queue/add_random)=="0"

# Spread CPU load by redirecting completions to originating CPU
ACTION=="add|change", KERNEL=="sd[!0-9]", SUBSYSTEM=="block", ENV[ID_VENDOR]=="PURE", ATTR(queue/rq_affinity)=="2"
ACTION=="add|change", KERNEL=="sd[!0-9]", SUBSYSTEM=="block", ENV[ID_VENDOR]=="PURE", ATTR(queue.nr_requests)=="1024"
```

Figure 12: crm configure with OCFS2 resource

**CONFIGURATION OF UDEV RULES**

The device manager of the kernel needs to be configured as shown below. The most important parameters to be changed are nr_requests and scheduler. Please set parameters for Pure Storage in the udev/rules.d as shown below:

```
ACTION=="add|change", KERNEL=="sd[!0-9]", SUBSYSTEM=="block", ENV[ID_VENDOR]=="PURE", ATTR(queue/scheduler)=="noop"

ACTION=="add|change", KERNEL=="sd[!0-9]", SUBSYSTEM=="block", ENV[ID_VENDOR]=="PURE", ATTR(queue/add_random)=="0"

ACTION=="add|change", KERNEL=="sd[!0-9]", SUBSYSTEM=="block", ENV[ID_VENDOR]=="PURE", ATTR(queue/rq_affinity)=="2"
ACTION=="add|change", KERNEL=="sd[!0-9]", SUBSYSTEM=="block", ENV[ID_VENDOR]=="PURE", ATTR(queue.nr_requests)=="1024"
```

Figure 13: Pure Storage udev Rules
ACTION="add|change", KERNEL="sd*[!0-9]", SUBSYSTEM="block", ENV{ID_VENDOR}="PURE", ATTR{queue/nomerges}="1"
ACTION="add|change", KERNEL="sd*[!0-9]", SUBSYSTEM="block", ENV{ID_VENDOR}="PURE", ATTR{queue/rotational}="0"
ACTION="add|change", ENV{DM_UUID}="mpath-3624a937?*", ATTR{queue/scheduler}="noop"
ACTION="add|change", ENV{DM_UUID}="mpath-3624a937?*", ATTR{queue/rotational}="0"
ACTION="add|change", ENV{DM_UUID}="mpath-3624a937?*", ATTR{queue/nr_requests}="4096"
ACTION="add|change", ENV{DM_UUID}="mpath-3624a937?*", ATTR{queue/rq_affinity}="2"
ACTION="add|change", ENV{DM_UUID}="mpath-3624a937?*", ATTR{queue/nomerges}="1"
ACTION="add|change", ENV{DM_UUID}="mpath-3624a937?*", ATTR{queue/add_random}="0"
ACTION="add|change", KERNEL="dm-*", ATTR{queue/scheduler}="noop"
ACTION="add|change", KERNEL="dm-*", ATTR{queue/rotational}="0"
ACTION="add|change", KERNEL="dm-*", ATTR{queue/nr_requests}="4096"
ACTION="add|change", KERNEL="dm-*", ATTR{queue/rq_affinity}="2"
ACTION="add|change", KERNEL="dm-*", ATTR{queue/nomerges}="1"
ACTION="add|change", KERNEL="dm-*", ATTR{queue/add_random}="0"

# SCSI timeout in line with Customer Best Practices
ACTION="add", SUBSYSTEM="scsi", ATTRS{model}="FlashArray", RUN="/bin/sh -c 'echo 60 > /sys$DEVPATH/timeout"

MULTIPATHING CONFIGURATION

Multipathing needs to be setup to do round-robin for all PURE LUNs by setting it up in /etc/multipath.conf. Install the sysstat package with Yast2 to get iostat in the servers. It is very important to set the path-selector in multipath configuration to round-robin.

The file contents of multipath.conf are shown here:

```
saphananode1:~ # cat /etc/multipath.conf
devices {
  device {
    vendor "PURE"
    path_selector "round-robin 0"
    path_grouping_policy multibus
    path_checker tur
```
HANA CONFIGURATION FILES

The following are the configuration details.

HANA GLOBAL.INI FILE

[communication]

listeninterface = .internal
internal_network = 192.168.1/24

[internal_hostname_resolution]

192.168.1.10 = saphananode1-int
192.168.1.11 = saphananode2-int
192.168.1.12 = saphananode3-int

[persistence]

basepath_datavolumes = /hana/data/PSG
basepath_logvolumes = /hana/log/PSG
use_mountpoints = yes
basepath_shared = yes

[storage]

ha_provider = hdb_ha.fcClient
partition_*_*__prtype = 5
partition_1_data__wwid = 3624a9370122281ffa371414400011070
partition_1_log__wwid = 3624a9370122281ffa371414400011071
partition_2_data__wwid = 3624a9370122281ffa371414400011072
partition_2_log__wwid = 3624a9370122281ffa371414400011073

[system_information]

usage = test

[trace]

ha_fcclient = info
HWCCT: FSPERF PARAMETERS

The following parameters were set for the new performance test tool from SAP HWCCT tool fsperf. This changes I/O behavior to enhance the database for work with the file system and storage.

async_read_submit=off
async_write_submit_blocks=new

For more information regarding these parameters, please refer to SAP Note 1943937.

In order to use these parameters in SAP HANA you need to execute the following commands in the Linux shell as <sid>adm user.

hdbparam –paraset fileio “[<path>]. async_write_submit_blocks=new

To set async_write_submit_blocks for Data persistence: (Check the select query on the view M_VOLUME_IO_TOTAL_STATISTICS shown below to get the information on the path and trigger ratios.)

hdbparam –paraset fileio “[/hana/data/PS1/mnt00001/hdb00003/]. async_write_submit_blocks=new

This command returns lines indicating the success of the parameter setting for all services: NameServer, Preprocessor, IndexServer etc.

To find the path, execute the following query as shown below:
CONCLUSION

Unlike traditional storage architectures, the Pure Storage FlashArray does not require customers to compromise on resiliency and simplicity for the sake of performance. Whether you are looking to get started with SAP HANA, or migrating an existing HANA environment, the FlashArray is the ideal platform for maximizing your HANA investment.

Pure Storage recommends the following guidelines for sizing customer HANA environments. Please consult with your Pure Storage sales team for more detailed information.

<table>
<thead>
<tr>
<th>AFA MODEL</th>
<th>SAP HANA NODES</th>
</tr>
</thead>
<tbody>
<tr>
<td>//M10 R2</td>
<td>Up to 4</td>
</tr>
<tr>
<td>//M20 R2</td>
<td>Up to 8</td>
</tr>
<tr>
<td>//M50 R2</td>
<td>Up to 12</td>
</tr>
<tr>
<td>//M70 R2</td>
<td>Up to 16</td>
</tr>
</tbody>
</table>

Table 3. Sizing Guidelines for Pure Storage FlashArray//M R2.
REFERENCES

1. Pure Storage FlashArray Data Sheet

2. Compare Flash Array with other Flash alternatives
   http://www.purestorage.com/pdf/Pure_Storage_Flash_Array_vs_Flash_Alternatives.pdf

3. Pure Storage Forever Flash