



# Deploy IBM DB2 pureScale on Azure

By Benjamin Guinebertière, Alessandro Vozza, Jonathon Frost, Mukesh Kumar, and Larry Mead  
Azure Customer Advisory Team (AzureCAT)  
Commercial Software Engineering Team (CSE)  
Data Migration JumpStart Team (DMJ)

May 2018

This document is provided "as-is". Information and views expressed in this document, including URL and other Internet Web site references, may change without notice.

Some examples depicted herein are provided for illustration only and are fictitious. No real association or connection is intended or should be inferred.

This document does not provide you with any legal rights to any intellectual property in any Microsoft product. You may copy and use this document for your internal, reference purposes.

© 2018 Microsoft. All rights reserved.

# Contents

Introduction .....	3
Architecture .....	4
Compute considerations.....	5
Storage considerations.....	5
Networking considerations.....	6
Solution deployment.....	7
How the deployment works .....	7
DB2 pureScale response file .....	8
Troubleshooting and known issues.....	9
Learn more .....	10

Authored by Benjamin Guinebertière, Alessandro Vozza, Jonathon Frost, Mukesh Kumar, and Larry Mead. Edited by Nanette Ray.  
Reviewed by AzureCAT, CSE, and DMJ.

© 2018 Microsoft Corporation. This document is for informational purposes only. MICROSOFT MAKES NO WARRANTIES, EXPRESS OR IMPLIED, IN THIS SUMMARY. The names of actual companies and products mentioned herein may be the trademarks of their respective owners.

# Introduction

Enterprises have long used traditional RDBMS platforms to cater to OLTP needs. These days, many are migrating their mainframe-based database environments to the Azure cloud as a way to expand capacity, reduce costs, and maintain a steady operational cost structure. Migration is often the first step in modernizing a legacy platform.

The AzureCAT, CSE, and DMJ teams recently worked with an enterprise that rehosted their IBM DB2 environment running on z/OS to IBM DB2 pureScale on Azure. The DB2 pureScale database cluster solution provides high availability and scalability on Linux operating systems. We successfully ran DB2 standalone on a large scale-up system on Azure prior to installing DB2 pureScale.

While not identical to original environment, IBM DB2 pureScale on Linux delivers similar high availability and scalability features as IBM DB2 for z/OS running in a Parallel Sysplex environment on the mainframe.

This guide describes the steps we took during the migration so you can take advantage of our learnings. Installation scripts are available in the [repository on GitHub](#). These scripts are based on the architecture we used for a typical medium-sized OLTP workload.

Consider this guide and the scripts a starting point for your DB2 implementation plan. Your business requirements will differ, but the same basic pattern applies. This architectural pattern may also be used for OLAP applications on Azure.

This guide does not cover differences and possible migration tasks for moving IBM DB2 for z/OS to IBM DB2 pureScale running on Linux. Nor does it provide equivalent sizing estimations and workload analyses for moving from DB2 z/OS to DB2 pureScale architectures. Before you decide on the best DB2 pureScale architecture for your environment, we highly recommend that you complete a full sizing estimation exercise and establish a hypothesis. Among other factors, on the source system make sure to consider DB2 z/OS Parallel Sysplex with Data Sharing Architecture, Coupling Facility configuration, and DDF usage statistics.

---

**NOTE:** This guide is intended to describe one approach to DB2 migration, but there are others. For example, DB2 pureScale can also run in virtualized environments on premises. IBM supports DB2 on Microsoft Hyper-V in various configurations. For more information, see [Db2 pureScale virtualization architecture](#) in the IBM knowledge Center.

---

# Architecture

To support high availability and scalability on Azure, we set up a scale-out, shared data architecture for DB2 pureScale. We used the following architecture for our customer migration.

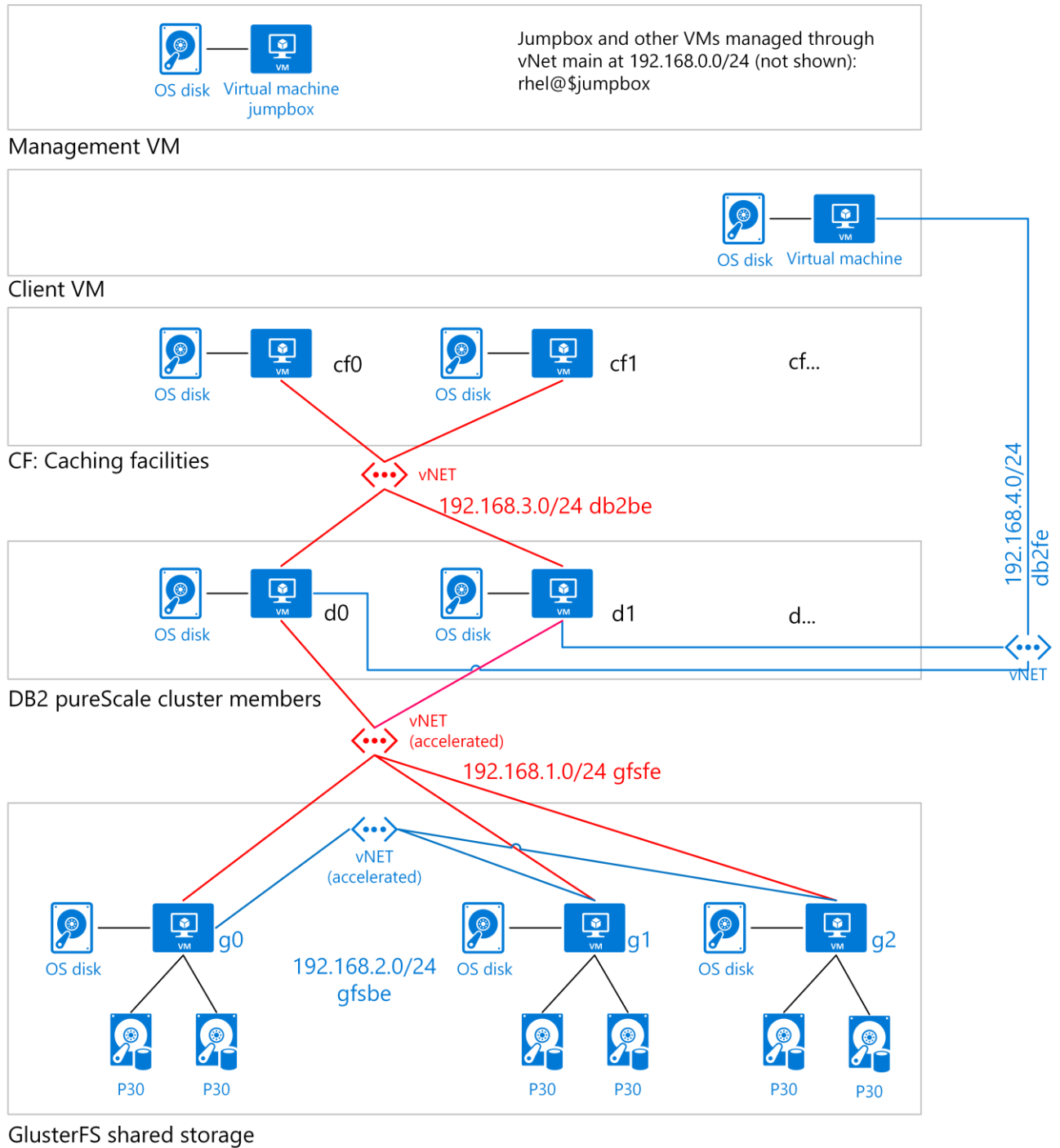


Figure 1. DB2 pureScale on Azure VM, Network and Storage Diagram

This diagram depicts a DB2 pureScale cluster where two nodes are used for the cache and are known as the caching facilities (CF). A minimum of two nodes are used for the database engine

and are known as cluster members. The cluster is connected via iSCSI to a three-node GlusterFS shared storage cluster to provide scale-out storage and high availability. DB2 pureScale is installed on Azure virtual machines running Linux.

Consider our approach a template that you can modify as needed to suit the size and scale needed by your organization. Our architectural approach is based on the following:

- Two or more database nodes are combined with at least two CF nodes that handle the global buffer pool (GBP) for shared memory and global lock manager (GLM) services to control shared access and lock contention from multiple active nodes. One CF node acts as the primary and the other as the secondary CF node. A minimum of four nodes are required for a DB2 pureScale cluster.
- High-performance shared storage (shown in P30 size in the diagram above), which is used by each of the Gluster FS nodes..
- High-performance networking for the data nodes and shared storage.

## Compute considerations

This architecture runs the application, storage, and data tiers on Azure virtual machines. The setup scripts create the following:

- DB2 pureScale cluster. The type of compute resources you need on Azure depend on your setup. In general, there are two approaches:
  - Use a multi-node, high-performance computing (HPC)-style network where multiple small to medium-sized instances access the shared storage. For this HPC type of configuration, Azure memory-optimized G-series or storage-optimized L-series [virtual machines](#) provide the needed compute power.
  - Use fewer large virtual machine instances for the data engines. For large instances, the largest memory optimized [M-series](#) virtual machines are ideal for heavy in-memory workloads, but a dedicated instance may be required depending on the size of the Logical Partition (LPAR) that runs DB2.
  - The DB2 CF uses memory-optimized VMs such as G-series or L-series.
  - GlusterFS storage uses Standard\_DS4\_v2 virtual machines running Linux.
  - A GlusterFS jumpbox is a Standard\_DS2\_v2 virtual machine running Linux.
  - The client is a Standard\_DS3\_v2 virtual machine running Windows to use for testing.
  - A witness server is a Standard\_DS3\_v2 virtual machine running Linux used for DB2 pureScale.

In either case, a minimum of two DB2 instances are required in a DB2 pureScale cluster. A Cache instance and Lock Manager instance are also required.

## Storage considerations

Like Oracle RAC, DB2 pureScale is a high-performance block I/O, scale-out database. We recommend using the largest available [Azure Premium Storage](#) that suits your needs. For example, smaller storage options may be suitable for a test environment while production

environments often use larger. We chose [P30](#) because of its ratio of IOPS to size and price. Regardless of size, use Premium Storage for best performance.

DB2 pureScale uses a shared everything architecture, where all data is accessible from all cluster nodes. Premium storage must be shared across multiple instances—whether on-demand or on dedicated instances.

A large DB2 pureScale cluster can require 200 terabytes (TB) or higher of Premium shared storage, with IOPS of 100,000. DB2 pureScale supports an iSCSI block interface that can be used on Azure. The iSCSI interface requires a shared storage cluster that can be implemented with GlusterFS, S2D, or another tool. This type of solution creates a virtual SAN (vSAN) device in Azure. DB2 pureScale uses the vSAN to install the General Parallel File System (GPFS) used to share data among multiple VMs.<sup>1</sup>

For this architecture, we use the GlusterFS file system, a free, scalable, open source distributed file system specifically optimized for cloud storage.

## Networking considerations

IBM recommends InfiniBand networking for all nodes in a DB2 pureScale cluster (both data and management nodes). For performance, DB2 pureScale also uses RDMA (where available) for the caching node.

During setup, an Azure [resource group](#) is created to contain all the virtual machines. In general, resources are grouped based on their lifetime and who will manage them. The virtual machines in this architecture require [accelerated networking](#), an Azure feature that provides consistent, ultra-low network latency via single root I/O virtualization (SR-IOV) to a virtual machine.

Every Azure virtual machine is deployed into a virtual network that is segmented into multiple subnets: main, Gluster FS front end (gfsfe), Gluster FS back end (bfsbe), DB2 pureScale (db2be), and DB2 pureScale front end (db2fe). The installation script also creates the primary [NICs](#) on the virtual machines in the main subnet.

[Network security groups](#) (NSGs) are used to restrict network traffic within the virtual network and isolate the subnets.

On Azure, DB2 pureScale needs to use TCP/IP as the network connection for storage.

---

<sup>1</sup> The performance benchmarks for the various vSAN implementations have yet to be established as this writing.

# Solution deployment

To deploy this architecture, run the `deploy.sh` script in the [DB2onAzure](#) repository on GitHub.

In addition, the repository also includes scripts you can use to set up a Grafana dashboard that supports querying Prometheus.

---

**NOTE:** The `deploy.sh` script on the client creates private SSH keys and passes them to the deployment template over HTTPS. For greater security, we recommend using [Azure Key Vault](#) to

---

## How the deployment works

The `deploy.sh` script creates and configures the Azure resources that are used in this architecture. The script prompts you for the Azure subscription and VMs for the target environment and then creates the following resources:

- Sets up the resource group, virtual network, and subnets on Azure for the installation.
- Sets up the NSGs and SSH for the environment.
- Sets up multiple NICs on both the GlusterFS and DB2 pureScale virtual machines.
- Creates the GlusterFS storage virtual machines.
- Creates the jumpbox virtual machine
- Creates the DB2 pureScale virtual machines.
- Creates the witness virtual machine that DB2 pureScale pings.
- Creates a Windows virtual machine to use for testing but does not install anything on it.

Next, the deployment scripts set up iSCSI vSAN for shared storage on Azure. In this example, iSCSI connects to GlusterFS. This solution also gives you the option to install the iSCSI targets as a single Windows node. iSCSI provides a shared block storage interface over TCP/IP that allows the DB2 pureScale setup procedure to use a device interface to connect to shared storage. For GlusterFS basics, see the [Architecture: Types of volumes](#) topic in Getting started with GlusterFS.

The deployment scripts follow these general steps:

1. Sets up a shared storage cluster on Azure. We use GlusterFS to set up our shared storage cluster. This involves at least two Linux nodes. For setup details, see [Setting up Red Hat Gluster Storage in Microsoft Azure](#) in the Red Hat Gluster documentation.
2. Sets up an iSCSI Direct interface on target Linux servers for GlusterFS. For setup details, [GlusterFS iSCSI](#) in the GlusterFS Administration Guide.
3. Sets up the iSCSI Initiator on the Linux virtual machines that will access the GlusterFS cluster using iSCSI Target. For setup details, see the [How To Configure An iSCSI Target And Initiator In Linux](#) in the RootUsers documentation.
4. Installs GlusterFS as the storage layer for the iSCSI interface.

After creating the iSCSI device, the final step is to install DB2 pureScale. The DB2 pureScale setup also compiles and installs [IBM GPFS](#) on the GlusterFS cluster. GPFS enables DB2 pureScale to

share data among the multiple virtual machines that run the DB2 pureScale engine. To tune your configuration, see [Best Practices: DB2 databases and the IBM GPFS](#).

For more information, see [Install and configure General Parallel File System \(GPFS\) on xSeries](#) on the IBM website. These installation instructions are for x86 versions of Linux but also apply to Linux virtual machines on Azure. To tune your configuration, see [Best Practices: DB2 databases and the IBM GPFS](#).

## DB2 pureScale response file

The repo includes DB2server.rsp, a response (.rsp) file that enables you to generate an automated script for the DB2 pureScale installation. The following table lists the DB2 pureScale options that the response file uses for setup. You can customize the response file for your installation environment. A sample response file (DB2server.rsp) is included. If you use this response file, you must edit it to work in your environment.

Screen name	Field	Value
Welcome		New Install
Choose a Product		DB2 Version 11.1.2.2. Server Editions with DB2 pureScale
Configuration	Directory	/data1/opt/ibm/DB2/V11.1
"	Select the installation type	Typical
"	I agree to the IBM terms	Checked
Instance Owner	Existing User For Instance, User name	DB2sdin1
Fenced User	Existing User, User name	DB2sdfe1
Cluster File System	Shared disk partition device path	/dev/dm-2
"	Mount point	/DB2sd_1804a
"	Shared disk for data	/dev/dm-1
"	Mount point (Data)	/DB2fs/datafs1
"	Shared disk for log	/dev/dm-0
"	Mount point (Log)	/DB2fs/logfs1
"	DB2 Cluster Services Tiebreaker. Device path	/dev/dm-3
Host List	d1 [eth1], d2 [eth1], cf1 [eth1], cf2 [eth1]	



"	Preferred primary CF	cf1
"	Preferred secondary CF	cf2
Response File and Summary	first option	Install DB2 Server Edition with the IBM DB2 pureScale feature and save my settings in a response file
"	Response file name	/root/DB2server.rsp

Note that /dev-dm0, /dev-dm1, /dev-dm2 and /dev-dm3 can change after a reboot on the virtual machine where the setup takes place (d0 in the automated script). To find the right values, you can issue the following command before completing the response file on the server where the setup will be run:

```
[root@d0 rhel]# ls -als /dev/mapper
total 0
0 drwxr-xr-x  2 root root    140 May 30 11:07 .
0 drwxr-xr-x 19 root root   4060 May 30 11:31 ..
0 crw-----  1 root root 10, 236 May 30 11:04 control
0 lrwxrwxrwx  1 root root    7 May 30 11:07 db2data1 -> ../dm-1
0 lrwxrwxrwx  1 root root    7 May 30 11:07 db2log1 -> ../dm-0
0 lrwxrwxrwx  1 root root    7 May 30 11:26 db2shared -> ../dm-2
0 lrwxrwxrwx  1 root root    7 May 30 11:08 db2tieb -> ../dm-3
```

The setup scripts use aliases for the iSCSI disks so that the actual names can be found easily.

Also, when the setup is run on d0, the /dev/dm-\* values may be different on d1, cf0 and cf1. The pureScale setup doesn't care.

## Troubleshooting and known issues

The GitHub repo includes a [Knowledge Base](#) maintained by the authors. It lists potential issues you may encounter and resolutions to try. For example, known issues can occur when:

- Trying to reach the gateway IP address.
- Compiling GPL.
- The security handshake between hosts fails.
- The DB2 installer detects an existing file system.
- Manually installing GPFS.
- Installing DB2 pureScale when GPFS is already created.
- Removing DB2 pureScale and GPFS.

For more information about these and other known issues, see [kb.md](#) in the [DB2onAzure](#) repo.

## Learn more

[GlusterFS iSCSI](#)

[Creating required users for a DB2 pureScale Feature installation](#)

[DB2icrt - Create instance command](#)

[DB2 pureScale Clusters Data Solution](#)

[IBM Data Studio](#)

[Platform Modernization Alliance: IBM DB2 on Azure](#)

[Azure Virtual Data Center Lift and Shift Guide](#)

### Feedback and suggestions

If you have feedback or suggestions for improving this data migration asset, please contact the Data Migration Jumpstart Team ([askdmjfordmtools@microsoft.com](mailto:askdmjfordmtools@microsoft.com)). Thanks for your support!

**Note:** For additional information about migrating various source databases to Azure, see the [Azure Database Migration Guide](#).