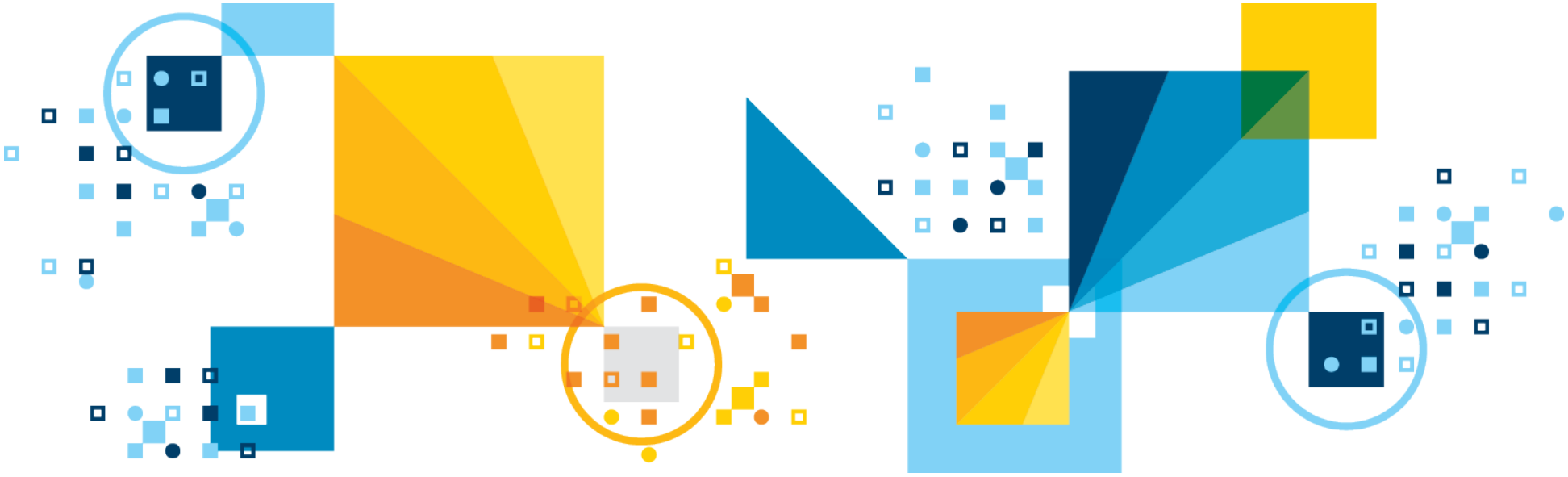
**Enzo Cialini, ecialini@ca.ibm.com**
**IBM STSM & Master Inventor**
**WW Data Servers Technical Sales**

# Db2 pureScale

## Continuously Available and Transparently Scalable



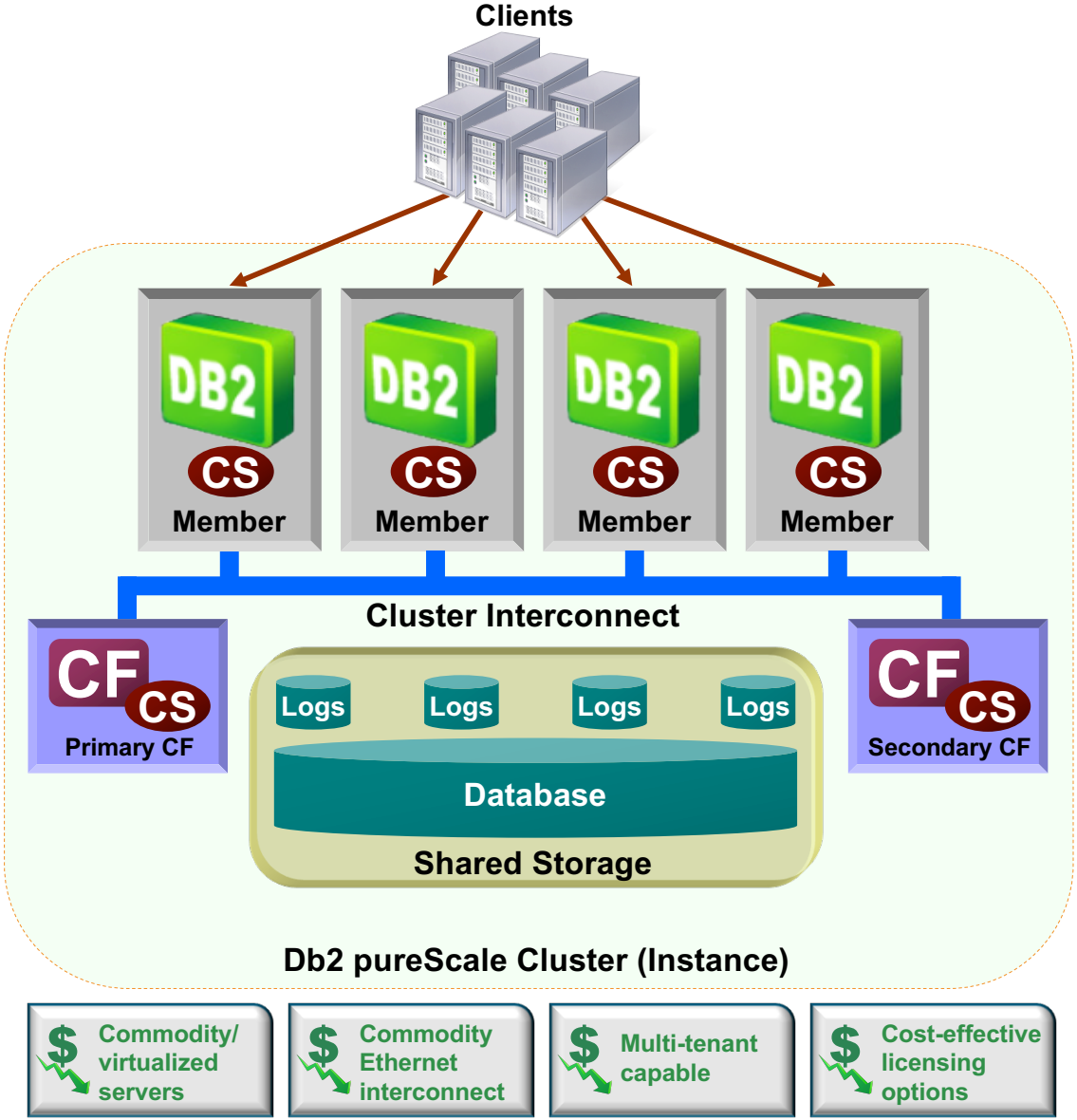
# pureScale - Introduction

- **Extreme capacity**
  - Buy only what you need, add capacity as your needs grow
- **Application transparency**
  - Avoid the risk and cost of application changes
- **Continuous availability**
  - Deliver uninterrupted access to your data with consistent performance


**Learning from the undisputed Gold Standard... System z**

# Db2 pureScale Architecture

*Leveraging IBM's System z Sysplex Experience and Know-How*




- Multiple Db2 members for scalable and available database environment
- Client application connects into any Db2 member to execute transactions
  - Automatic workload balancing
- Shared storage for database data and transaction logs
- Cluster caching facilities (CF) provide centralized global locking and page cache management for highest levels of availability and scalability
  - Duplexed, for no single point of failure
- High speed, low latency interconnect for efficient and scalable communication between members and CFs
- Db2 Cluster Services provides integrated failure detection, recovery automation and the clustered file system

# Ease of Deployment with Db2 pureScale

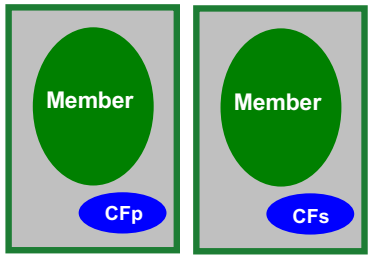
# Machine Deployment Examples

- **Highly flexible topologies due to logical nature of member and CF**
  - A member and CF can share the same machine
  - For AIX, separate members and CFs in different LPARs
  - Virtualized environments via VMware and KVM

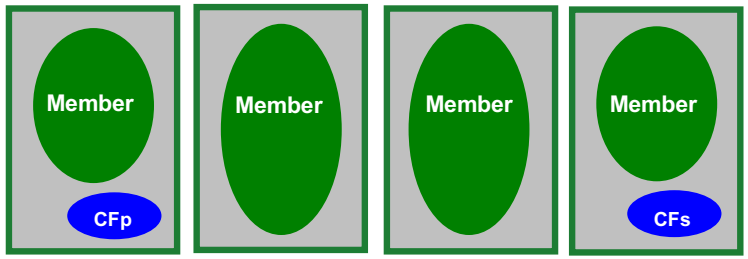
- **Dedicated cores for CFs**
  - Optimizes response time

- **No pureScale licenses required for CF hosts**
  - You only need to license the CPUs for hosts on which members are running

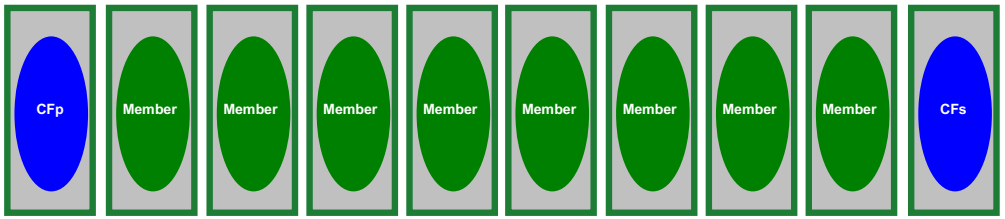
**Two Machines**




**Four Machines**




**Ten Machines**

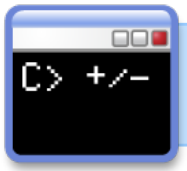


# Db2 pureScale is Easy to Deploy

Single installation for all components

Monitoring integrated into Optim tools

Single installation for fixpacks and updates




Simple command to add and remove members

# Db2 pureScale: Simplified Install and Deployment


- **Fast Up and Running**
  - Up and running in hours compared to competitive cluster databases
- **Install re-engineering includes:**
  - "Push-Button" install for pureScale clusters
    - Socket complexity reduced by at least 40%
    - Smarter defaults, intuitive options, parallel & quick pre-deployment host validation
  - 30-step native GPFS setup reduced to simple 4-step Db2 install process
    - Also easier conversion to GPFS replication post-deployment using db2cluster
  - Increased Resiliency for aborted/partial installations
    - Clean rollback for re-installation
- **Additional assistance via:**
  - Simplified documentation
  - Enhanced pre-checking of storage, tiebreaker devices, network adapters, firmware libraries
  - Intuitive and user-friendly errors & warnings

# pureScale Deployment Flexibility – Flexible Hardware On-Premises or On-Cloud

AIX OR redhat Linux

POWER6
POWER7/7+
POWER8
Flex

suse A NOVELL BUSINESS OR redhat Linux

x86 Intel Compatible Servers

Flex

BladeCenter H22/HS23

TCP/IP sockets interconnect or high speed, low latency RDMA-based interconnect (InfiniBand, RoCE)

GPFS compatible storage

*(ideally storage that supports SCSI-3 PR fast I/O fencing)*

# pureScale Deployment Flexibility – Flexible Hardware

- **Take advantage of the latest generation of POWER8 processors, with game-changing innovation that accelerates analytics**

- **POWER8 support for pureScale implementations using**
  - TCP/IP sockets
  - 10GE and 100GE RoCE (AIX)


POWER8

# pureScale Deployment Flexibility - Virtualization

- **pureScale can be deployed in bare metal or virtual environments**

- **Virtualized environments provide a lower cost of entry and are a perfect for**
  - Development
  - QA and testing
  - Production environments
  - Getting hands-on experience with pureScale


- **Virtualized environments and supported interconnects include**
  - PowerVM LPARs with AIX
    - InfiniBand*, RoCE, TCP/IP
  - KVM with RHEL
    - RoCE, TCP/IP
  - VMware with RHEL or SLES
    - TCP/IP, RoCE


# pureScale Operating Systems and Virtualization

- **Supported Operating Systems**
  - AIX Version 7.1 TL3 SP5, TL3 SP9, TL4 SP4
  - AIX Version 7.2 TL0 SP3, TL0 SP4, TL1 SP2, TL2
  - x86 64-bit
    - Red Hat Enterprise Linux (RHEL) 6.7+,6.8,6.9,7.2+,7.3,7.4
    - SUSE Linux Enterprise Server (SLES) 11 SP4, 12 SP1
  - Power Linux LE (Little Endian)
    - Red Hat Enterprise Linux (RHEL) 7.2+, 7.4
- **Supported Virtualization**
  - IBM Power
    - IBM PowerVM and PowerKVM
  - Linux X86-64 Platforms
    - Red Hat KVM
  - VMWare ESXi
    - Red Hat
    - SUSE

# pureScale Operating Systems Compatibility

- **Chrony support on RHEL 7.2**
  - Chrony replaced NTP as the default network time protocol in RHEL 7
  - Supported by pureScale (as an alternative to NTP)

# Upgrade Process - pureScale

- **Upgrade directly from Db2 Version 10.1 and 10.5**

- **Upgrading from Db2 Version 9.8**
  - Upgrade to latest fixpack of 10.1 or 10.5, then upgrade to 11.1

- **Ability to roll-forward through database version upgrades**
  - Upgrading from Db2 Version 10.5 Fix Pack 9*, or later
  - Users are no longer required to perform an offline backup of existing databases before or after they upgrade
  - A recovery procedure involving roll-forward through database upgrade now exists


*non-pureScale available 10.5 FP7

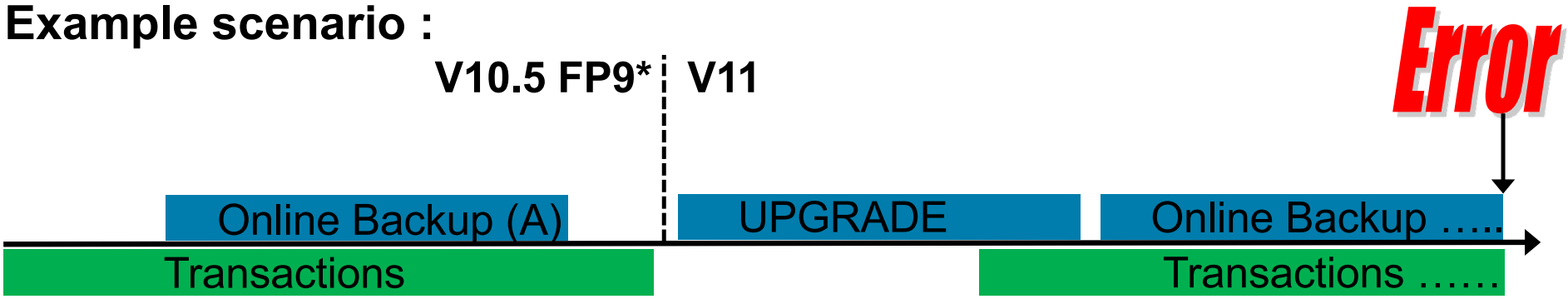
# Db2 Database Migration Log Records

- **Two New Propagatable Log Records**
  - The database migration begin log record is written to mark the start of database upgrade
  - The database migration end log record is written to mark the successful completion of database upgrade

- **Supports the ability to roll forward (replay) through a database upgrade**
  - Also used to allow for an HADR update on secondary sites without re-initialization of the database

# UPGRADE without Offline Backup

- **No more need to take an offline backup to ensure recoverability across upgrade!**
- **Example scenario :**

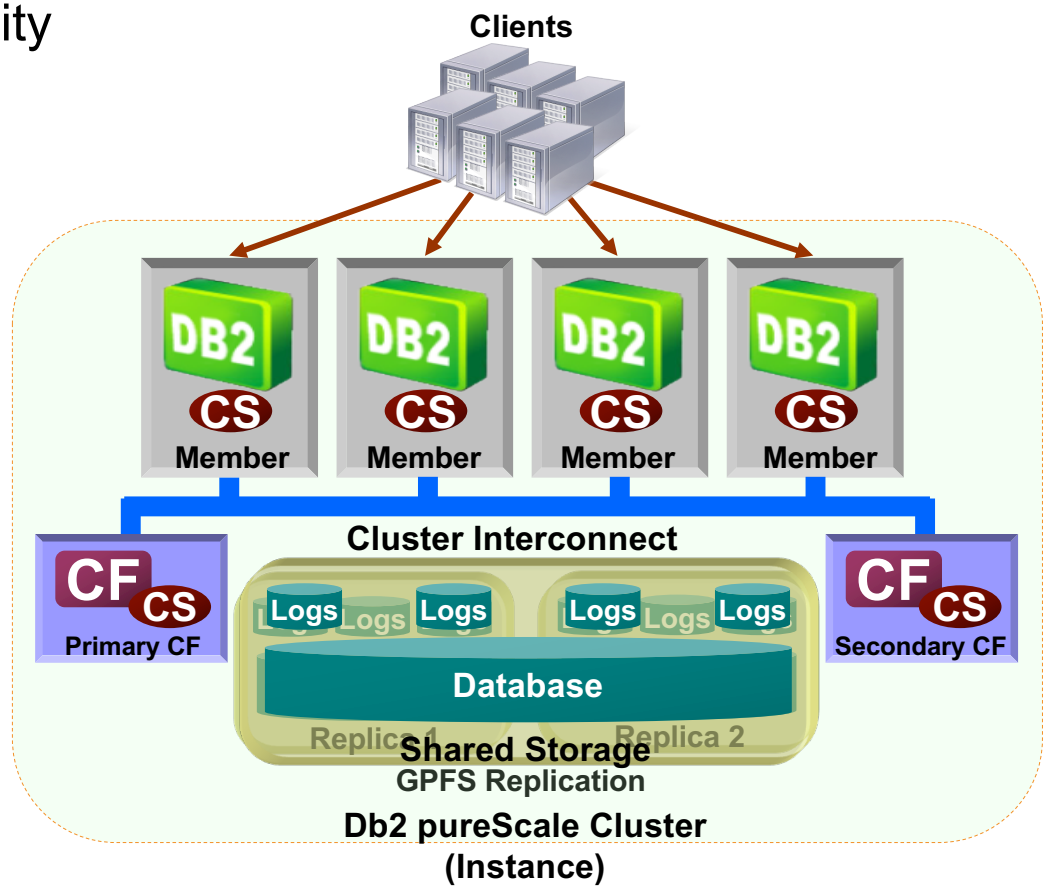



- **Recovery Procedure Overview:**
  - Revert Instance back to V10.5 FP9* (or higher)
  - Restore online backup (A)
  - Rollforward to a desired point-in-time just before the Error
    - Receive SQL2463N or SQL2464N indicating the start of upgrade
  - Upgrade Instance to V11
  - Continue Rollforward

*planned pureScale support in 10.5 FP9; non-pureScale available 10.5 FP7

# pureScale Deployment Flexibility - GPFS Replication

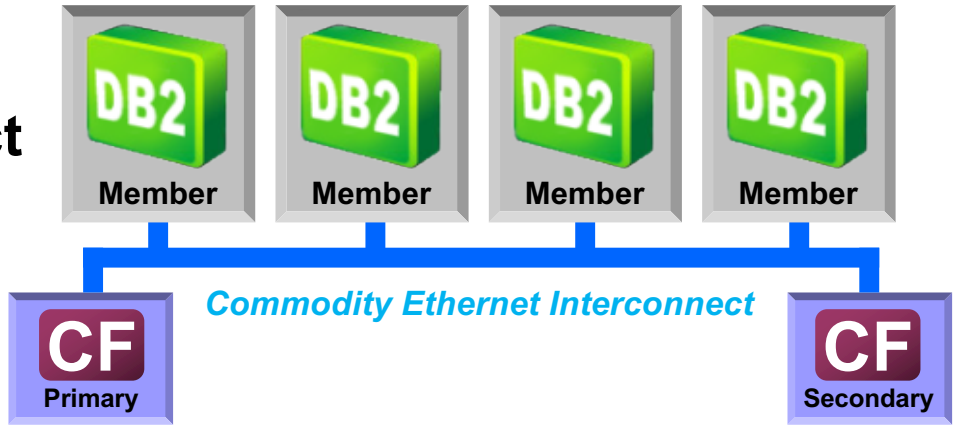
- **GPFS Configuration**
  - The configuration of GPFS replication in traditional Db2 pureScale cluster and Geographically Dispersed Db2 pureScale Cluster* (GDPC) has been enhanced by integrating all management and monitoring task with the **db2cluster** utility




*IBM LAB Services engagement still required for GDPC deployments

# pureScale Deployment Flexibility - TCP/IP Interconnect

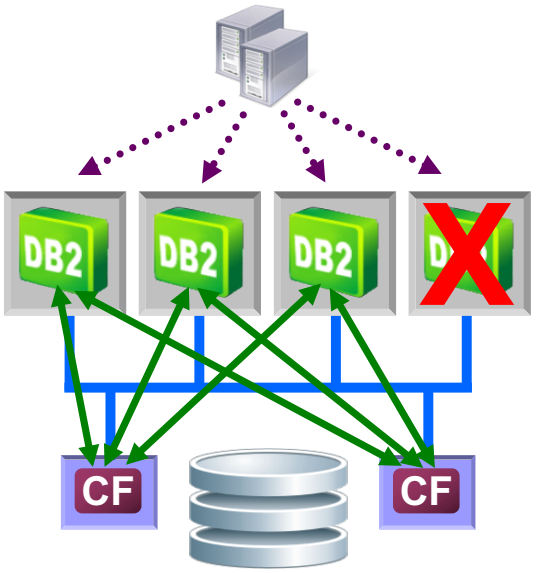
- **TCP/IP (sockets) interconnect for faster cluster setup and lower cost deployments using commodity network hardware**
- **Provides exactly the same level of high availability as RDMA-based pureScale environments**
- **Appropriate for small clusters with moderate workloads where availability is the primary motivator for pureScale**
- **Support for 1, 10 or 100 (AIX) Gigabit Ethernet (GE) interconnect**
  - At least 10 Gigabit Ethernet (10GE) strongly recommended for production installations

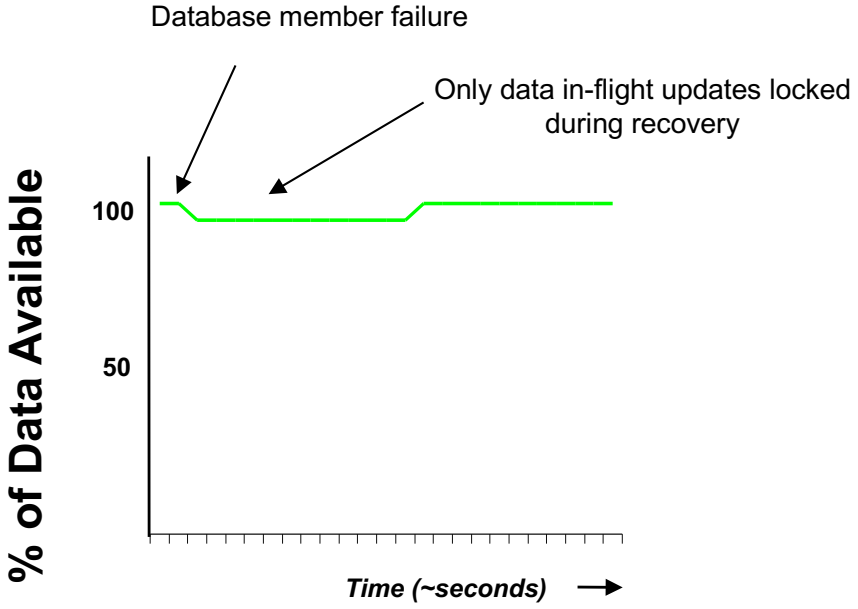


# High Availability in Db2 pureScale

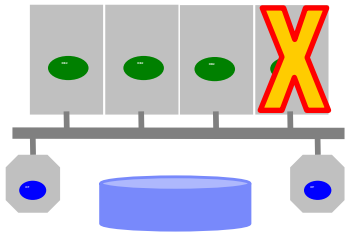
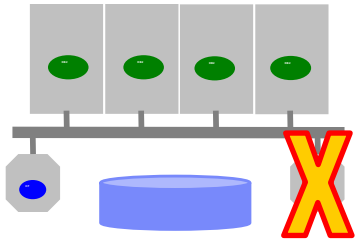
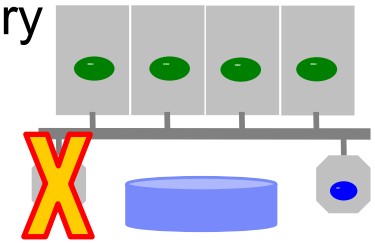
# pureScale HA - Online Recovery

- **Db2 pureScale design point is to maximize availability during failure recovery processing**
- **When a database member fails, only in-flight data remains locked until member recovery completes**
  - In-flight = data being updated on the failed member at the time it failed
- **Target time to availability of rows associated with in-flight updates on failed member in seconds**

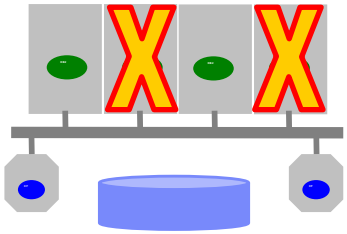
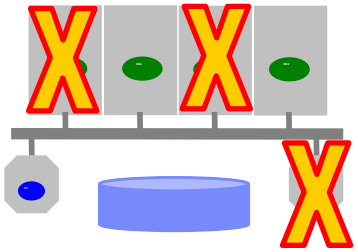
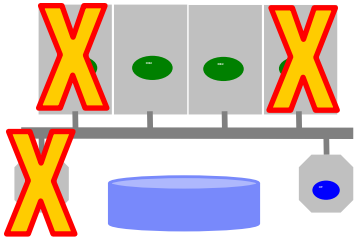


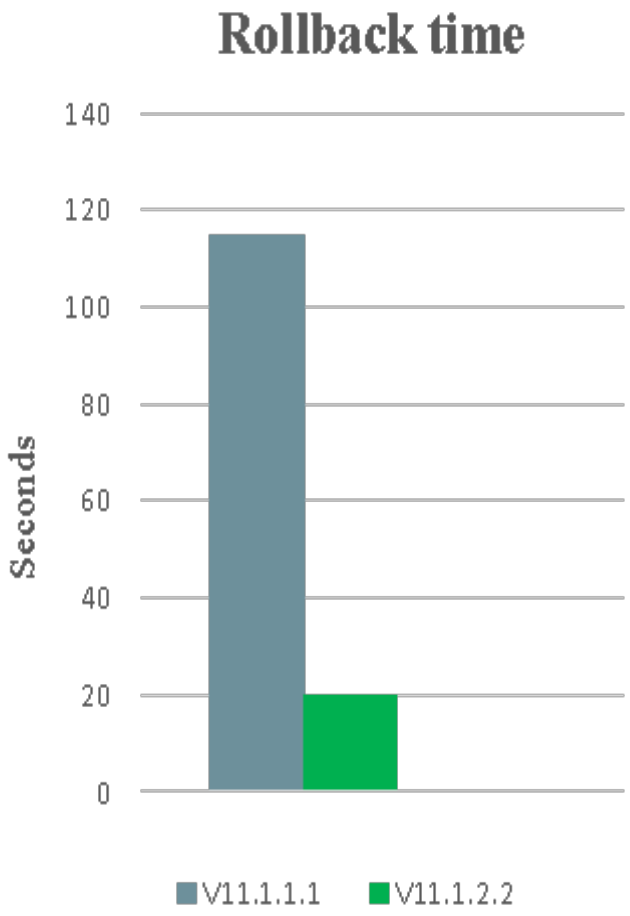
# Single Failure Scenarios

| Failure Mode | | Other Members Remain Online ? | Automatic and Transparent ? |
|---|---|---|---|
| Member |  | ✓ | ✓ Connections to failed member transparently move to another member |
| Primary *CF* |  | ✓ | ✓ |
| Secondary *CF* |  | ✓ | ✓ |

# Examples of Simultaneous Failures

| Failure Mode | Other Members Remain Online ? | Automatic and Transparent ? |
|---|---|---|
|  | ✓ | ✓ Connections to failed member transparently move to another member |
|  | ✓ | ✓ Connections to failed member transparently move to another member |
|  | ✓ | ✓ Connections to failed member transparently move to another member |

# Faster Transaction Rollback

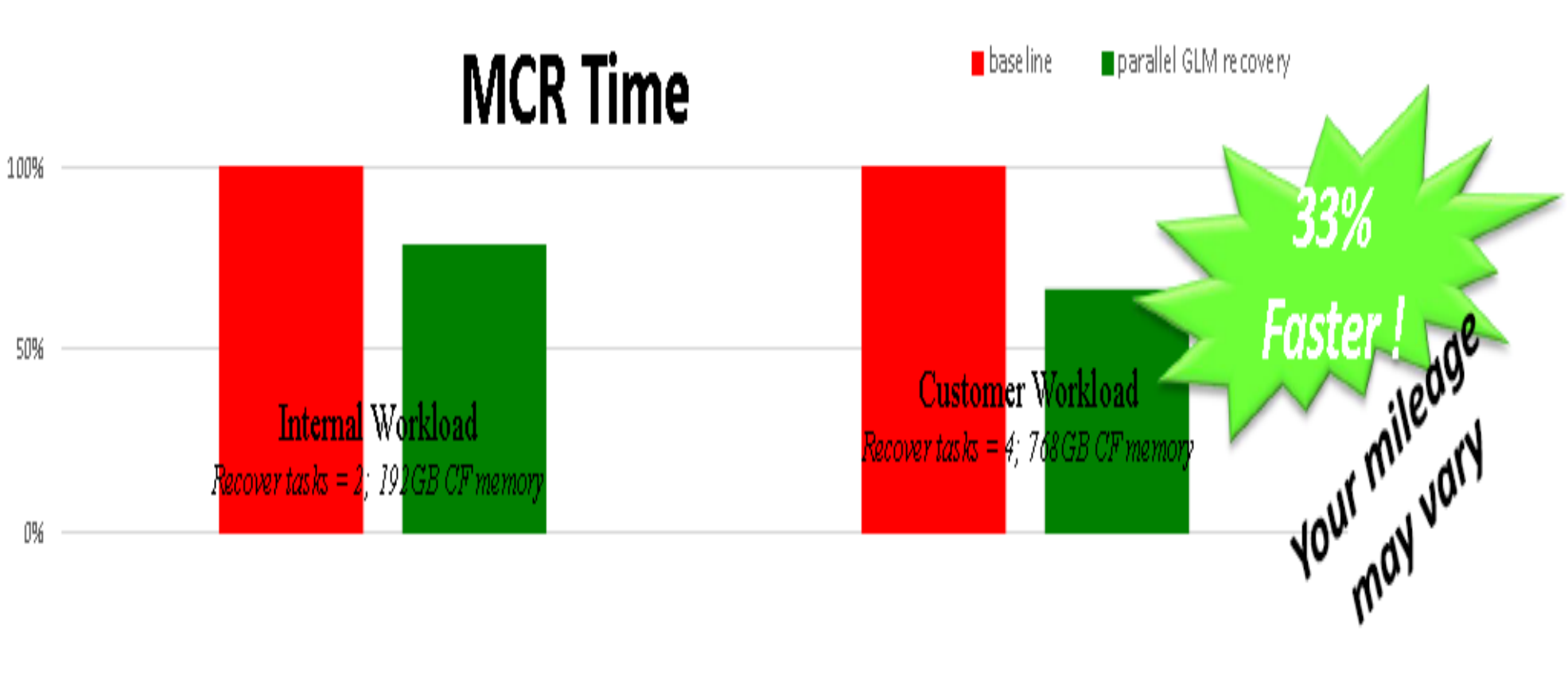



- 4 concurrent threads performing large insert transaction
- All threads issue **ROLLBACK** at approximately the same time
- Total rollback time (V11.1.1.1) : 115 sec
- Total rollback time (new) : **20 sec**

Power 7+ 16 cores, 64 logical cores (SMT4), 128GB

Performance is based on measurements and projections using internal IBM benchmarks in a controlled environment. The actual throughput or performance that any user will experience will vary depending upon many factors, including considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve results similar to those stated here.

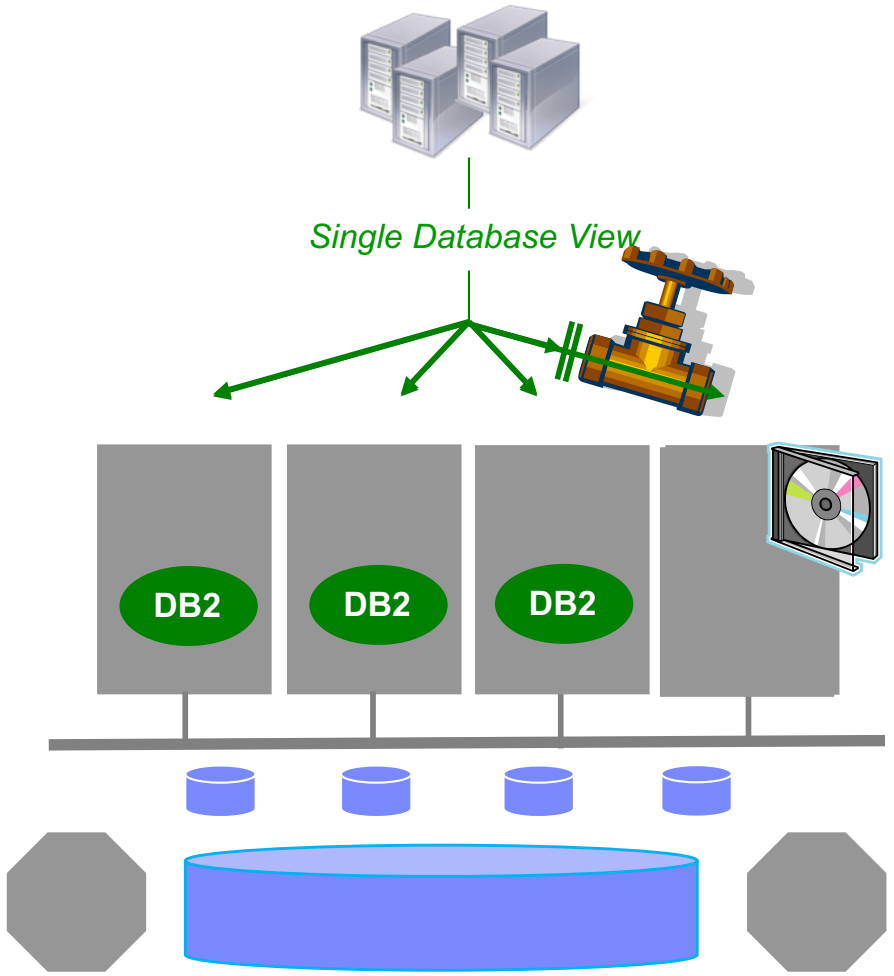
# Faster Member Crash Recovery (MCR)



# On-line Maintenance in Db2 pureScale

# pureScale HA - Stealth System Maintenance

- **Goal:** Allow DBAs to apply OS and system maintenance without negotiating an outage window
- **Example:** Upgrade the OS in a rolling fashion across the cluster
- **Procedure:**
  1. Drain (a.k.a. Quiesce)
     - Wait for transactions to end their life naturally; new transactions routed to other members
  2. Remove & maintain
  3. Reintegrate into cluster
     - Workload balancing starts sending it work as a least loaded machine
  4. Repeat until done

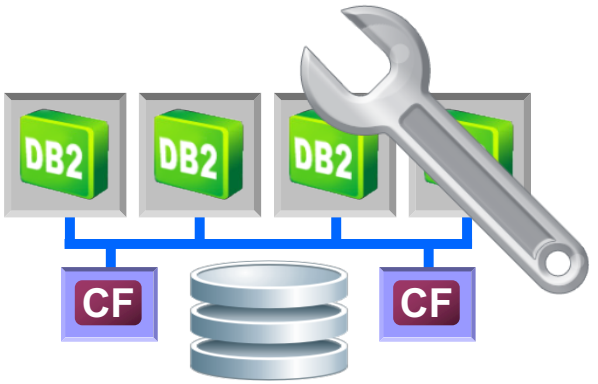


# pureScale HA – Online Rolling Database Fix Pack Updates

- **Transparently install pureScale fix packs in an online rolling fashion**

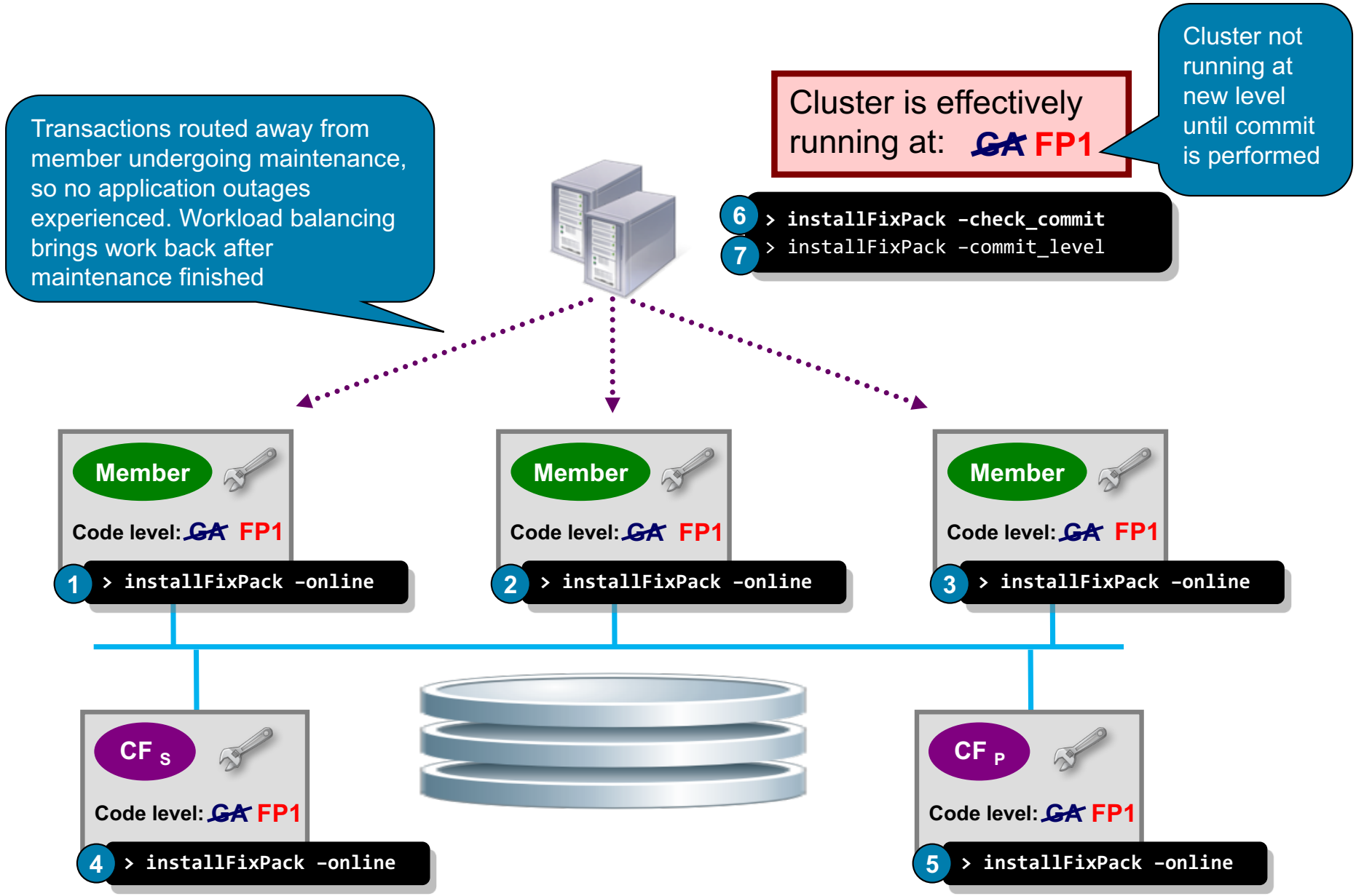
- **No outage experienced by applications**

- **Single `installFixPack` command run on each member/CF**
  - Quiesces member
    - Existing transactions allowed to finish (configurable timeout, default is 2 minutes)
    - New transactions sent to other members
  - Installs binaries
  - Updates instance
    - Member still behaves as if running on previous fix pack level
  - Unquiesces member




- **Final `installFixPack` command to complete and commit updates**
  - Instance now running at new fix pack level

# pureScale HA – Online Rolling Database Fix Pack Updates

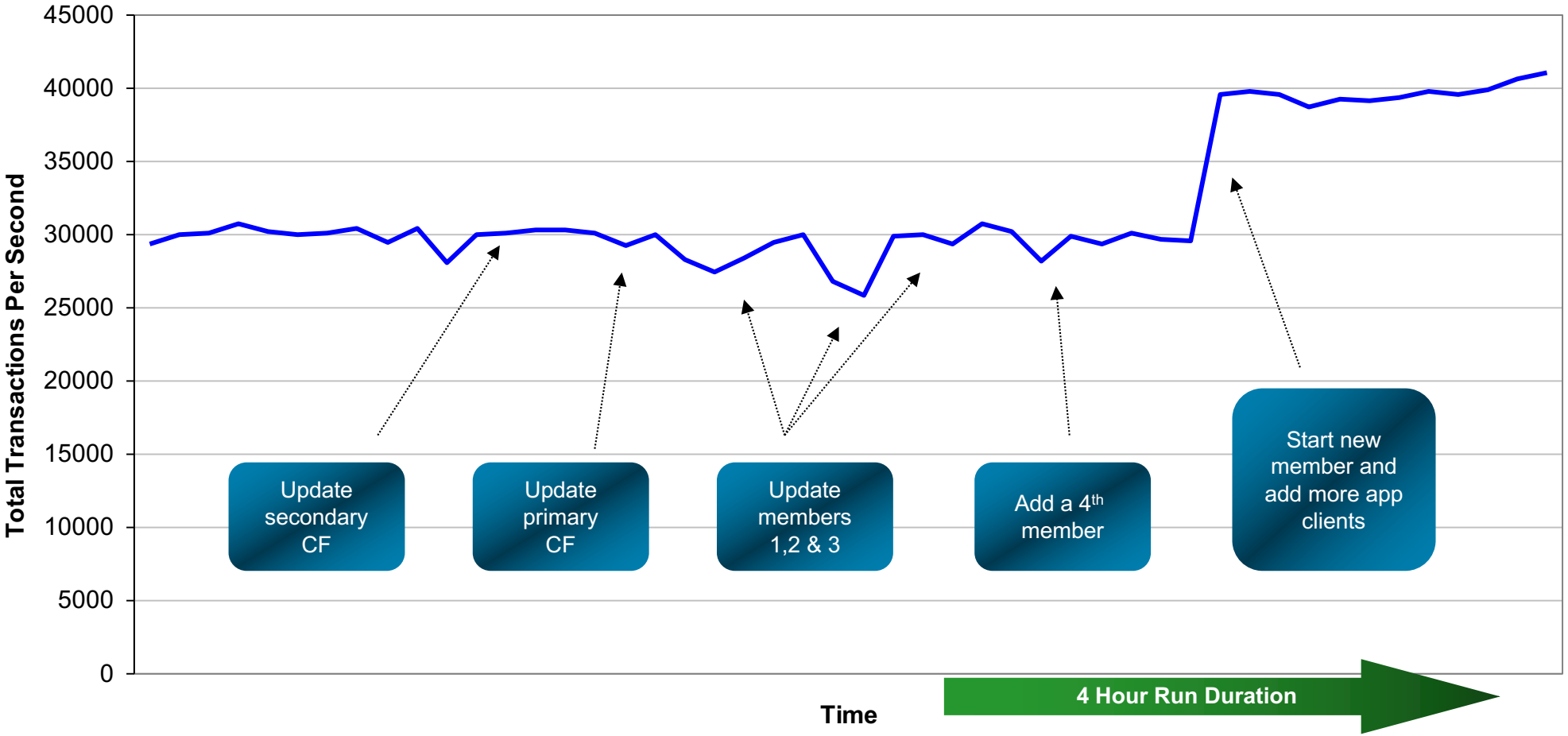







Simplified installFixPack commands are shown here for example purposes

# Continuous Availability During Maintenance and Growth




## Database servers

- SUSE Linux Enterprise Server 11 SP 1
- 6 - IBM x3950 X5s (Intel XEON X7560 @ 2.27GHz (4s/32c/64t))
- Mellanox ConnectX-2 IB Card
- 128GB system memory

## Storage server

- 1 - IBM Storwize v7000
- 8 - SSD drives (2TB usable capacity)
  - 4 - for data, 4 - for logs

# On-line Management with Db2 pureScale

# pureScale On-line Management – Reorg

*In-place (Online) Table Reorganization*

- **Online table reorganization fully supported in pureScale**
  - Reclaim free space
  - Eliminate overflows
  - Re-establish clustering

- **Example**

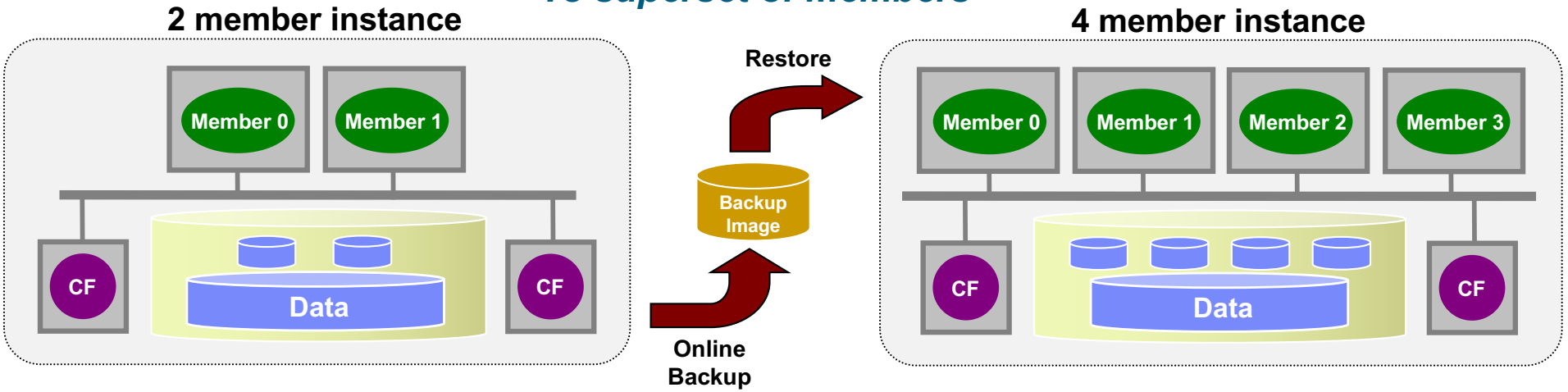
```
REORG TABLE <tableName> INPLACE ALLOW READ ACCESS
```

# pureScale On-line Management – Backup Restore

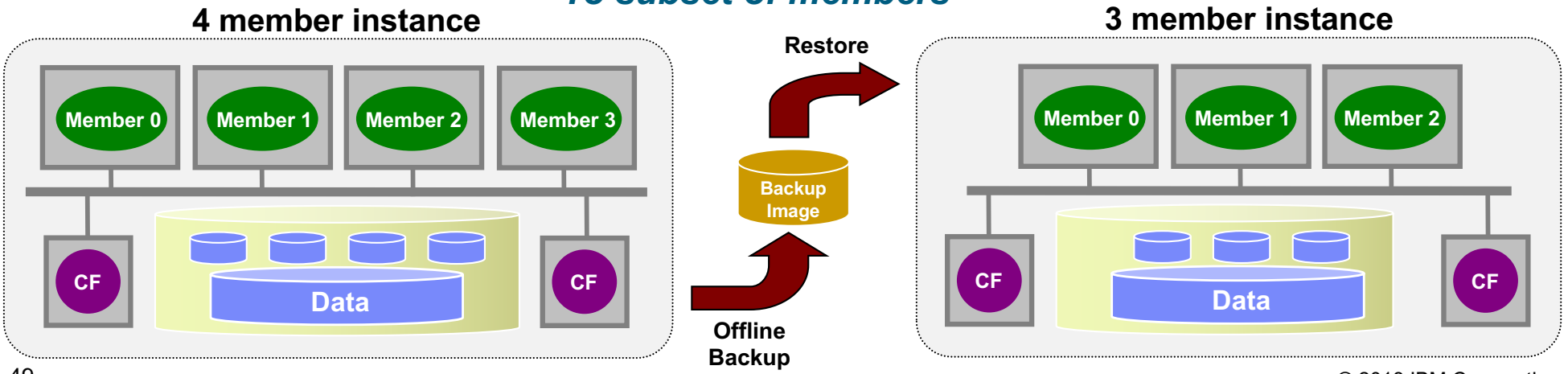
*Topology-Changing B/R*

- **Backup and restore between topologies with differing numbers of members**

***To superset of members***



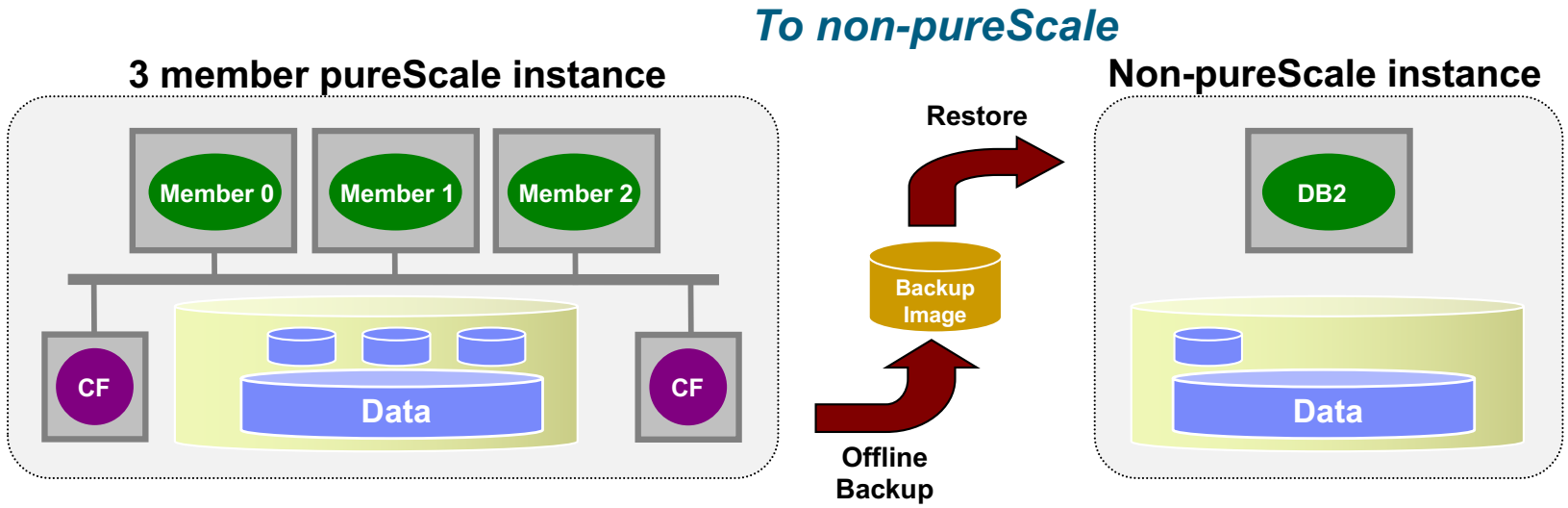

***To subset of members***

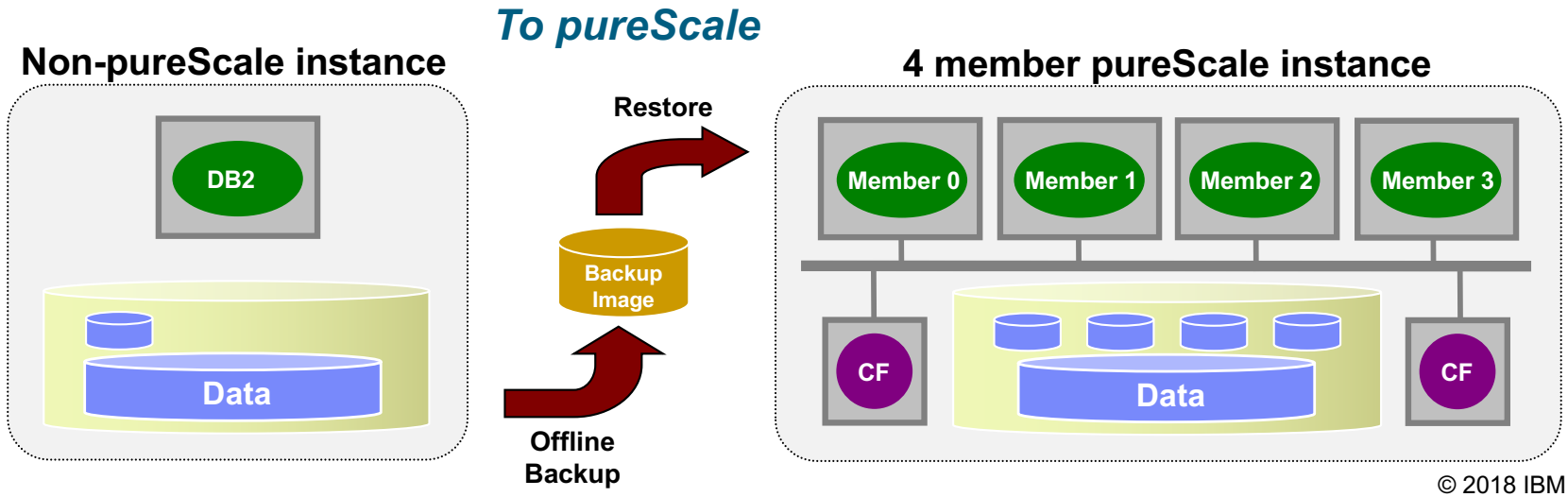


# pureScale Management – Backup Restore

*Topology-Changing B/R*

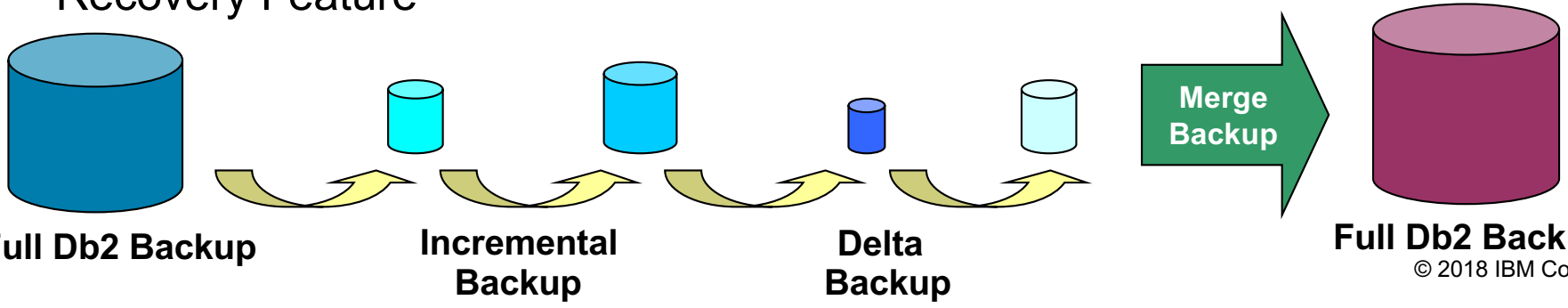
- **Backup and restore between pureScale to non-pureScale**



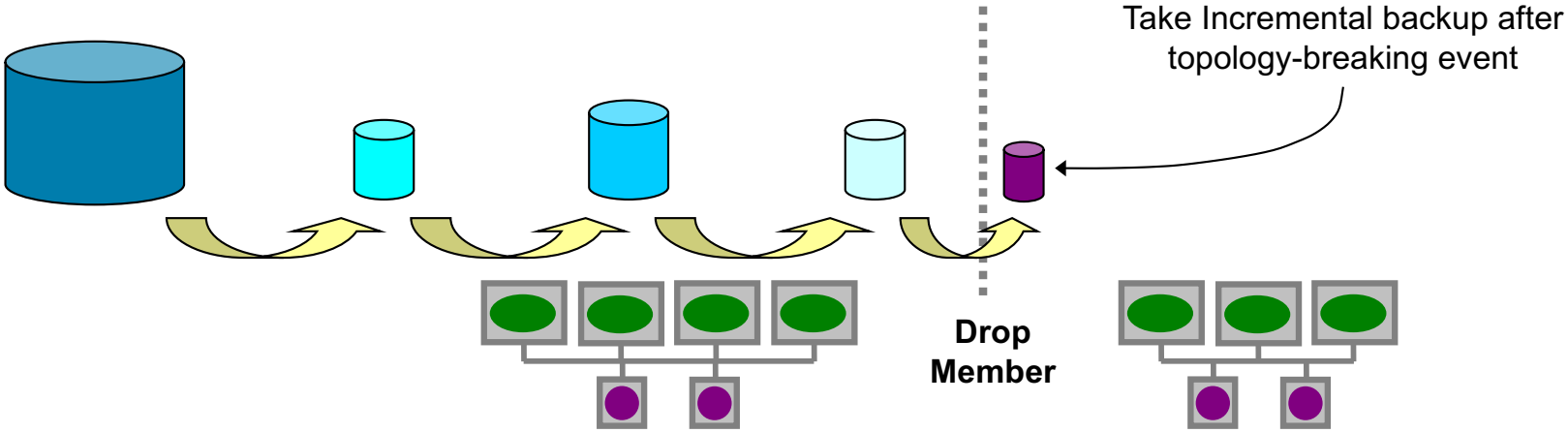
# pureScale Management - Backup and Db2 Merge Backup

- **Incremental backups supported for pureScale**
  - Allows for smaller backup images, as unchanged data not backed up
  - Applicable to database-level or table space-level backups
  - Enabled via `TRACKMOD` database configuration parameter
- **Two types of backups**
  - Incremental: Copy of all data that has changed since the most recent, successful, full backup operation (also know as cumulative backup)
  - Delta: Copy of all data that has changed since the last successful backup of any type (full, incremental, or delta) (also known as a differential backup)
- **Support for pureScale in Db2 Merge Backup V2.1 FP1**
  - The Merge Backup utility combines an older full backup with subsequent incremental and delta backups to create a new full backup image
  - Tool available separately but also included in the IBM Db2 Advanced Recovery Feature



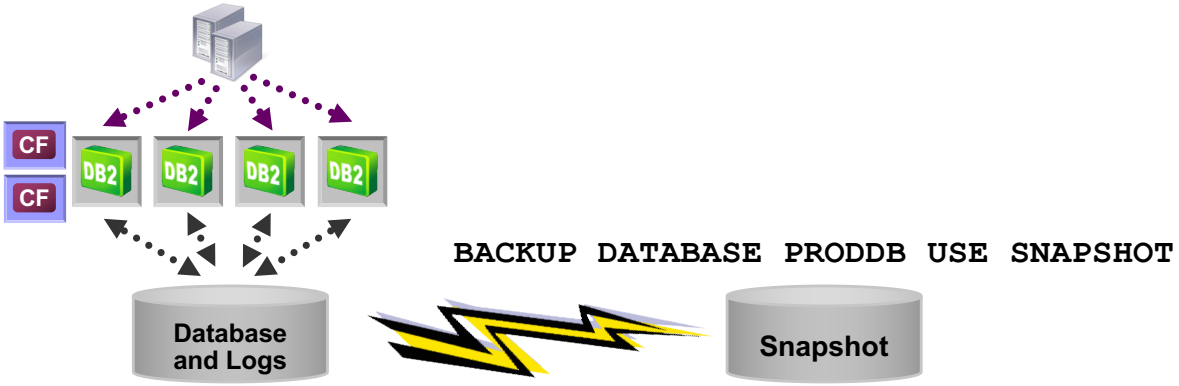
# pureScale Management - Database Topology Changes with Incremental Backup

- **Certain operations are considered "topology-breaking" for a database, such as**
  - Drop member from cluster
  - Restore database backup to a cluster with a subset of the members
  - Restore non-pureScale database backup into pureScale instance
  - Restore pureScale database backup into non-pureScale instance
- **A full offline database backup no longer required to provide a new recovery starting point for the database**
- **An incremental offline database backup can be performed instead**
  - Likely to be faster and resulting backup image will be smaller



# pureScale Management - Integrated Snapshot Backups

- **Backup large pureScale databases very fast, very easily!**




- **Db2 uses Advanced Copy Services (ACS) to perform integrated snapshot backups**
  - Db2 ACS API driver required for a given storage device
  - Db2 also ships with the Tivoli FlashCopy Manager (FCM) driver
  - Tivoli FCM 4.1 now supports GPFS-based file system snapshots
    - This is the method used for integrated snapshot backups in pureScale
    - Supports all storage that GPFS itself supports (so no hardware limitations)

- **Alternative methods previously available can still be used**
  - Manual snapshot process
  - Snapshot backup scripts

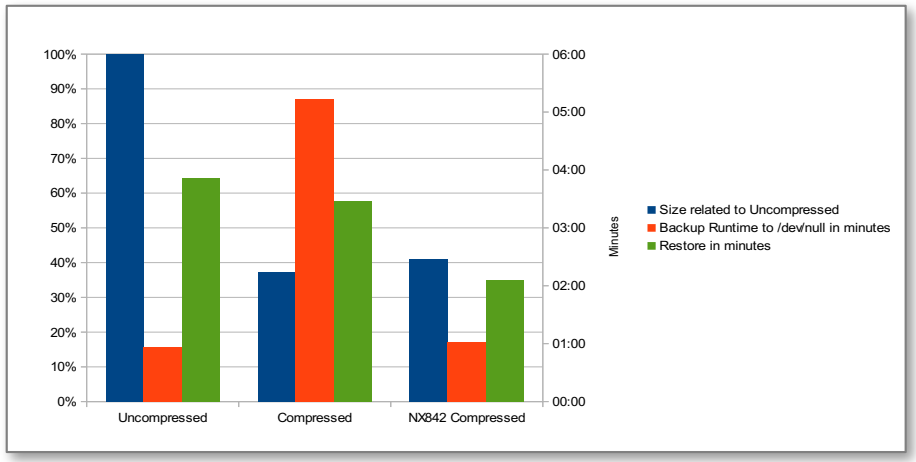
# Db2 Support for the NX842 Accelerator

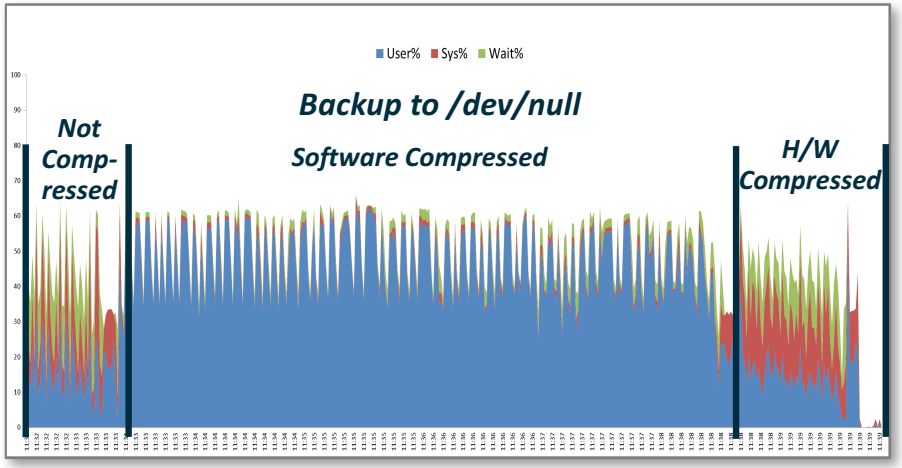
- **Db2 backup and log archive compression now support the NX842 hardware accelerator on POWER 7+ and POWER 8 processors**

- **Db2 BACKUPs require the use of a specific NX842 library**
  - `backup database <dbname> compress comprlib libdb2nx842.a`
- **Backups can be compressed by default with NX842**
  - Registry variable **DB2_BCKP_COMPRESSION** has to be set to **NX842**
  - Use the following backup command format:
  - `backup database <dbname> compress`
- **Log archive compression is also supported**
  - Update the database configuration parameter **LOGARCHCOMPR1** or **LOGARCHCOMPR2** to **NX842**
  - `update database configuration for <dbname> using LOGARCHCOMPR1 NX842`
  - Note: These two parameters can still take different values

# Db2 Backup Compression Performance Results

- **Preliminary results from early system testing**
- **About 50% Db2 backup size reduction compared to uncompressed**
- **Factor 2x less CPU consumption compared to Db2 compression**
  - Very significant reduction in CPU consumption
  - Very significant reduction in elapsed time
  - Maintains almost all of the compression storage benefits




Internal Tests at IBM Germany Research & Development

Performance is based on measurements and projections using standard IBM benchmarks in a controlled environment. The actual throughput or performance that any user will experience will vary depending upon many factors, including considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve results similar to those stated here.

# Remote Storage Option for Utilities

- **Remote storage is now accessible from:**
  - INGEST, LOAD, BACKUP, and RESTORE
  - Accessed through the use of storage access aliases

- **Supported Storage**
  - IBM® SoftLayer® Object Storage
  - Amazon Simple Storage Service (S3)

**Helps facilitate a Db2 Cloud or Hybrid Cloud solution**

# pureScale Multi-host Maintenance Mode

- **In 11.1.2.2 (and earlier) you have 2 options:**
  - Single host in maintenance mode
  - ALL hosts (cluster outage) in maintenance mode

  Iff you have an issue resulting in a member in maintenance mode for a long time (e.g. hardware issue), it will prevent any other host going into maintenance mode, including for a Rolling Update.

- **As of 11.1.3.3 you can now:**
  - Enable multiple hosts to be in maintenance mode concurrently
  - Enter maintenance on hosts one at a time until there is only one host (with at least a CF, and preferably a member - even if 2 hosts ) left in the cluster

# Multiple Hosts in Maintenance Mode

## Sample cluster:

```
0    coralpib252   0 - - MEMBER
1    coralpib249   0 - - MEMBER
2    coralpib250   0 - - MEMBER   *** this host is unavailable due to a hardware issue and already in maintenance mode!
128  coralpib239   0 - - CF
129  coralpib240   0 - - CF
```

## Back when original issue occurred on M2, it was put in maintenance mode:

| | |
|---|---|
| coralpib252 | root@coralpib252:/> /home/user1/sqllib/bin/db2cluster -cm -enter -maintenance<br>Host 'coralpib252' has entered maintenance mode |

## But now a Rolling Update (RU) is needed! Start w/ a CF:

| | |
|---|---|
| coralpib239 | root@coralpib252:/> <media>/installFixPack -p FP-install-path -l instance-name -online -l log-file-name -t trace-file-name |

## Now with v11.1.3.3 you can RU. Starting with this CF , and once it completes then continue to cycle through all the remaining CFs and members

Note: the RU cannot commit till the coralpib252 issue is fixed

## This would also allow RU multiple members/CFs at once

useful if you have a large number

# Online CREATE INDEX (concurrent write access)

- **Online CREATE INDEX with concurrent write access**

```
db2set DB2_CREATE_INDEX_ALLOW_WRITE=ON
db2 connect to MYDB
db2 create index ….
```

  - Default is **OFF** for pureScale, **ON** for non-pureScale
    - Dynamic registry variable setting, so can be enabled online
  - Only affects a recoverable database
    - (logarchmeth1 or logarchmeth2 database configuration parameters set)
  - Not supported for expression-based indexes in pureScale



60

# Db2 pureScale Management Features

- **Pre-Db2 Instance creation**
  - db2prereqcheck –adapter_list <adapter_list_filename> option added
  - Used to verify the network connectivity between all the hosts are pingable using RDMA
  - <adapter_list_filename> Specifies the file name that contains the list of hostname, netname, and adapter names for each of the host to be verified
  - The input file must have the following format:

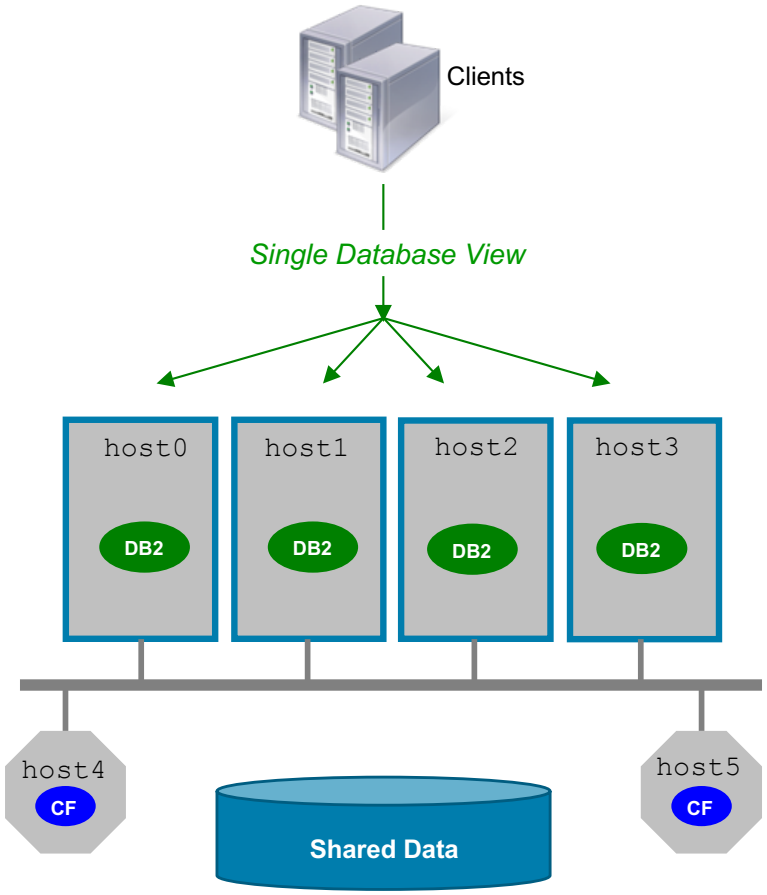
| #Hostname | Netname | Interface-Adapter |
|---|---|---|
| hostname1 | netname1 | devicename-1 |
| hostname2 | netname2 | devicename-2 |
| hostname3 | netname3 | devicename-3 |

any line that is preceded by # is considered as comment and skipped.

# Db2 pureScale Management Features

- **Improved Health Check**
  - db2cluster -verify is a post-installation unified health check tool for a Db2 pureScale cluster
  - The validations performed include, but are not limited to, the following:
    - Configuration settings in peer domain and GPFS cluster
    - Communications between members and CFs (including RDMA)
    - Replication setting for each file system
    - Status of each disk in the file system

# Instance and Host Status




db2nodes.cfg

```
0   host0   0   -   MEMBER
1   host1   0   -   MEMBER
2   host2   0   -   MEMBER
3   host3   0   -   MEMBER
4   host4   0   -   CF
5   host5   0   -   CF
```

```
> db2start
08/24/2008 00:52:59   0   0   SQL1063N  DB2START processing was successful.
08/24/2008 00:53:00   1   0   SQL1063N  DB2START processing was successful.
08/24/2008 00:53:01   2   0   SQL1063N  DB2START processing was successful.
08/24/2008 00:53:01   3   0   SQL1063N  DB2START processing was successful.
SQL1063N  DB2START processing was successful.
```

```
> db2instance -list
```

Instance status

| ID | TYPE | STATE | HOME_HOST | CURRENT_HOST | ALERT |
|---|---|---|---|---|---|
| 0 | MEMBER | STARTED | host0 | host0 | NO |
| 1 | MEMBER | STARTED | host1 | host1 | NO |
| 2 | MEMBER | STARTED | host2 | host2 | NO |
| 3 | MEMBER | STARTED | host3 | host3 | NO |
| 4 | CF | PRIMARY | host4 | host4 | NO |
| 5 | CF | PEER | host5 | host5 | NO |

Host status

| HOST_NAME | STATE | INSTANCE_STOPPED | ALERT |
|---|---|---|---|
| host0 | ACTIVE | NO | NO |
| host1 | ACTIVE | NO | NO |
| host2 | ACTIVE | NO | NO |
| host3 | ACTIVE | NO | NO |
| host4 | ACTIVE | NO | NO |
| host5 | ACTIVE | NO | NO |

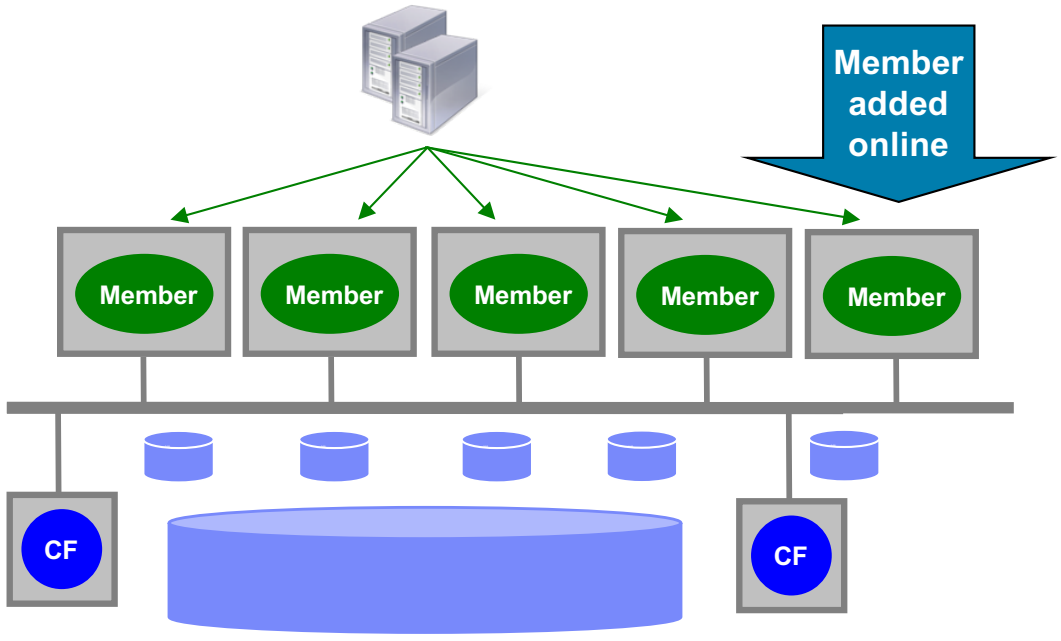
# Performance & Scalability in Db2 pureScale

# pureScale Scalability - Online Scale out Growth

- **New members can be added to an instance while it is online**
  - No impact to workloads running on existing members

```
db2iupdt -add -m <newHost> -mnet <networkName> <instance>
```

- **No application changes**
  - Efficient coherency protocols designed to scale without application change
  - Applications automatically and transparently workload balanced across members
- **No administrative complexities**
  - No database tuning
  - No data redistribution required



# ADD / DROP CF

- **In addition to the ability to ADD a member online**

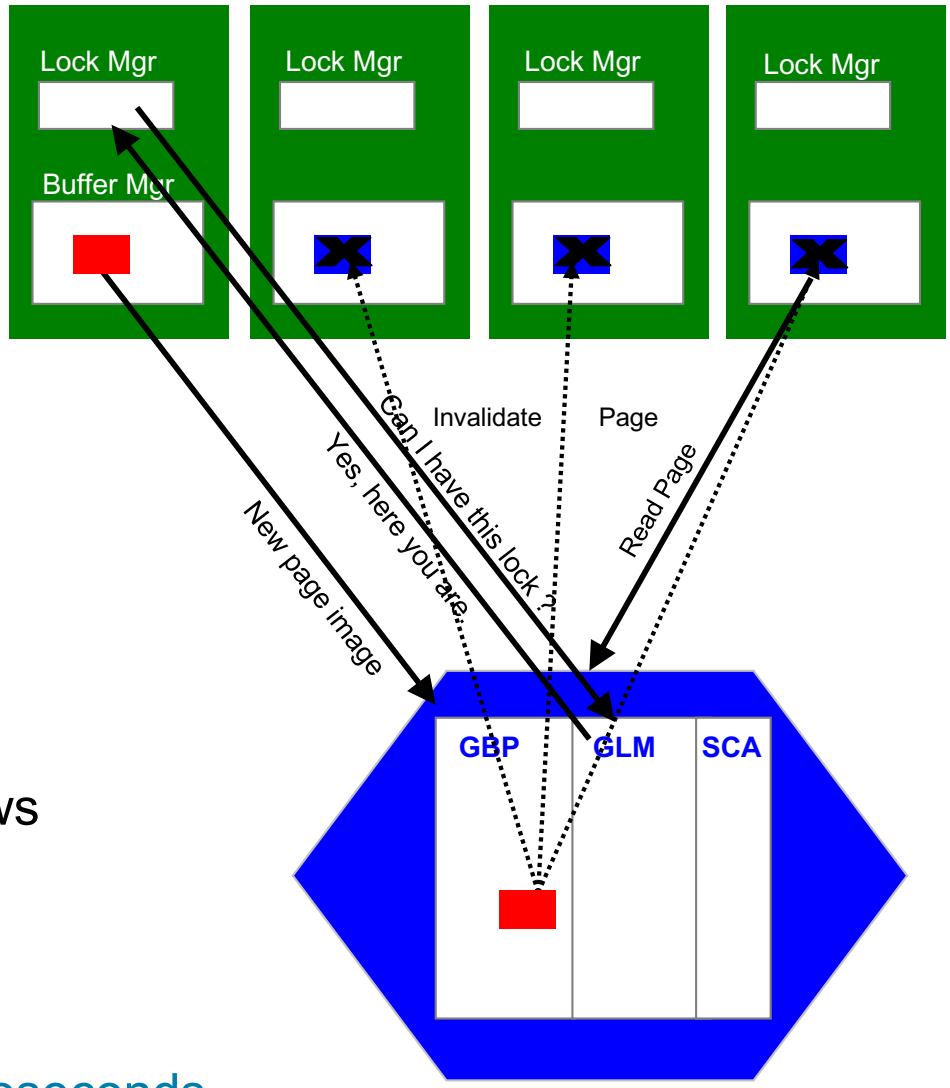
- **As of 11.1.3.3 you can now:**
  - Add a CF online
  - Drop a CF online

  Note: you cannot online drop the primary if the secondary is not in PEER state

- **Expected use cases:**
  - Move CF to a new host
  - Remove a failed or maintenance-mode CF
  - Improve HA of a single-CF cluster by adding a second CF

# Achieving Efficient Scaling – Key Design Points

- **Deep RDMA exploitation over low latency fabric**
  - Memory to memory communication
  - Enables round-trip response time ~10-15 microseconds
- **Silent Invalidation**
  - Informs members of page updates
  - Requires no CPU cycles on those members
  - No interrupt or other message processing required
  - Increasingly important as cluster grows
- **Hot pages available without disk I/O from GBP memory**
  - RDMA and dedicated threads enable read page operations in ~10s of microseconds






71

# Proof of Db2 pureScale Architecture Scalability

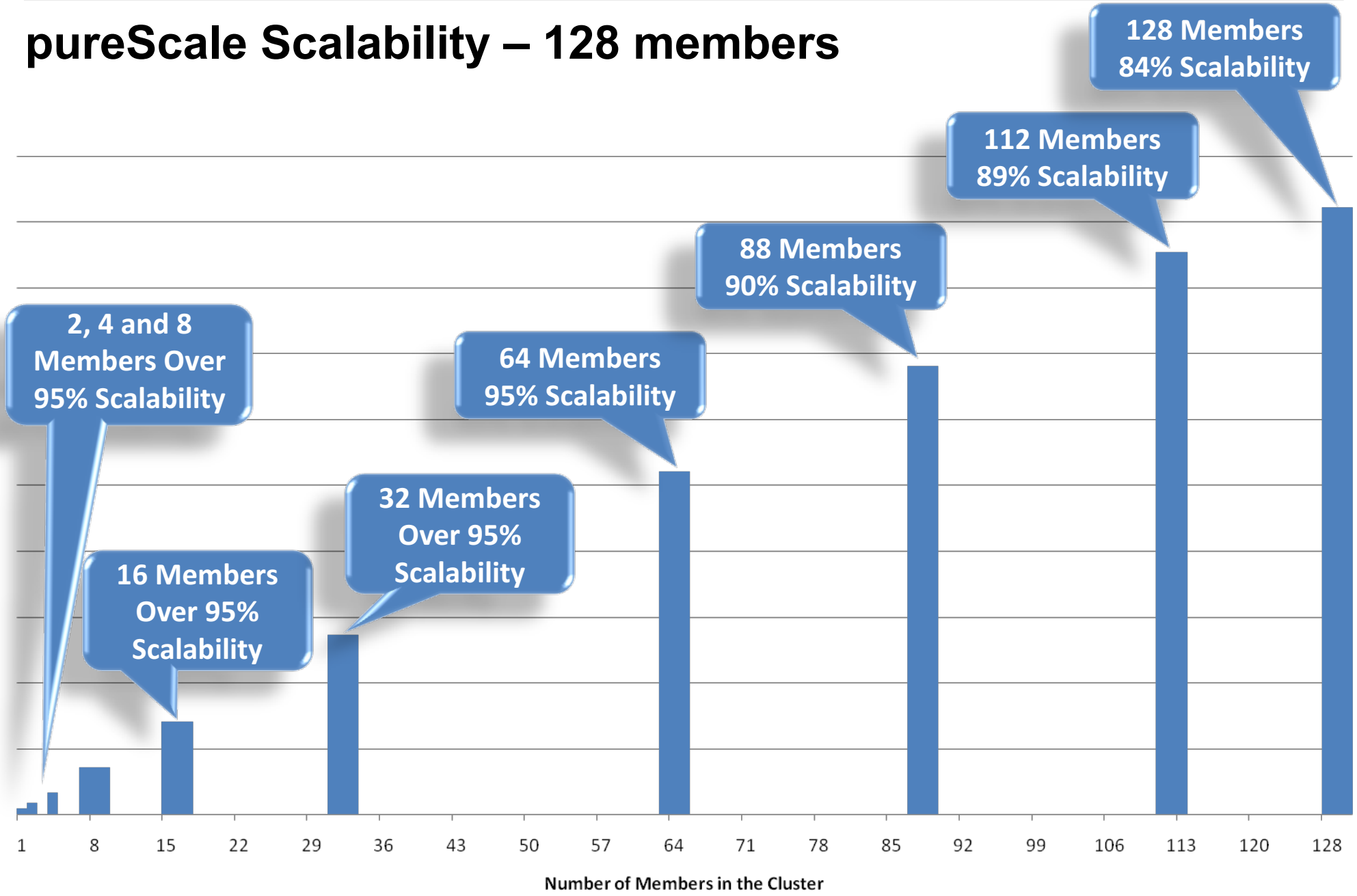
- **How far will it scale?**

- **Take a web commerce type workload**
  - Read mostly but not read only

- **Don't make the application cluster aware**
  - No routing of transactions to members
  - Demonstrate transparent application scaling

- **Scale out to the 128 member limit and measure scalability**

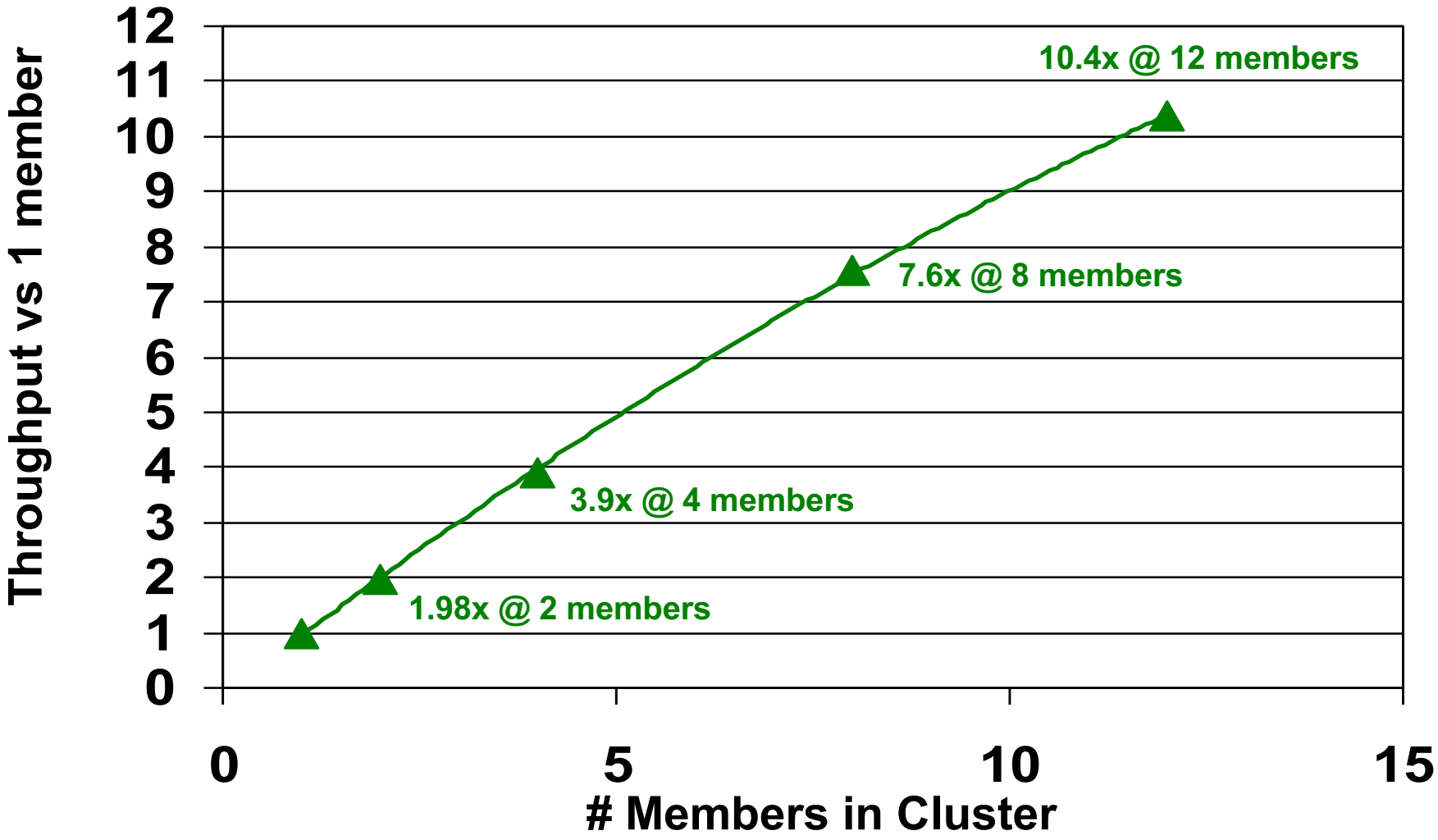
# pureScale Scalability – 128 members



# Dive Deeper into a 12 Member Cluster

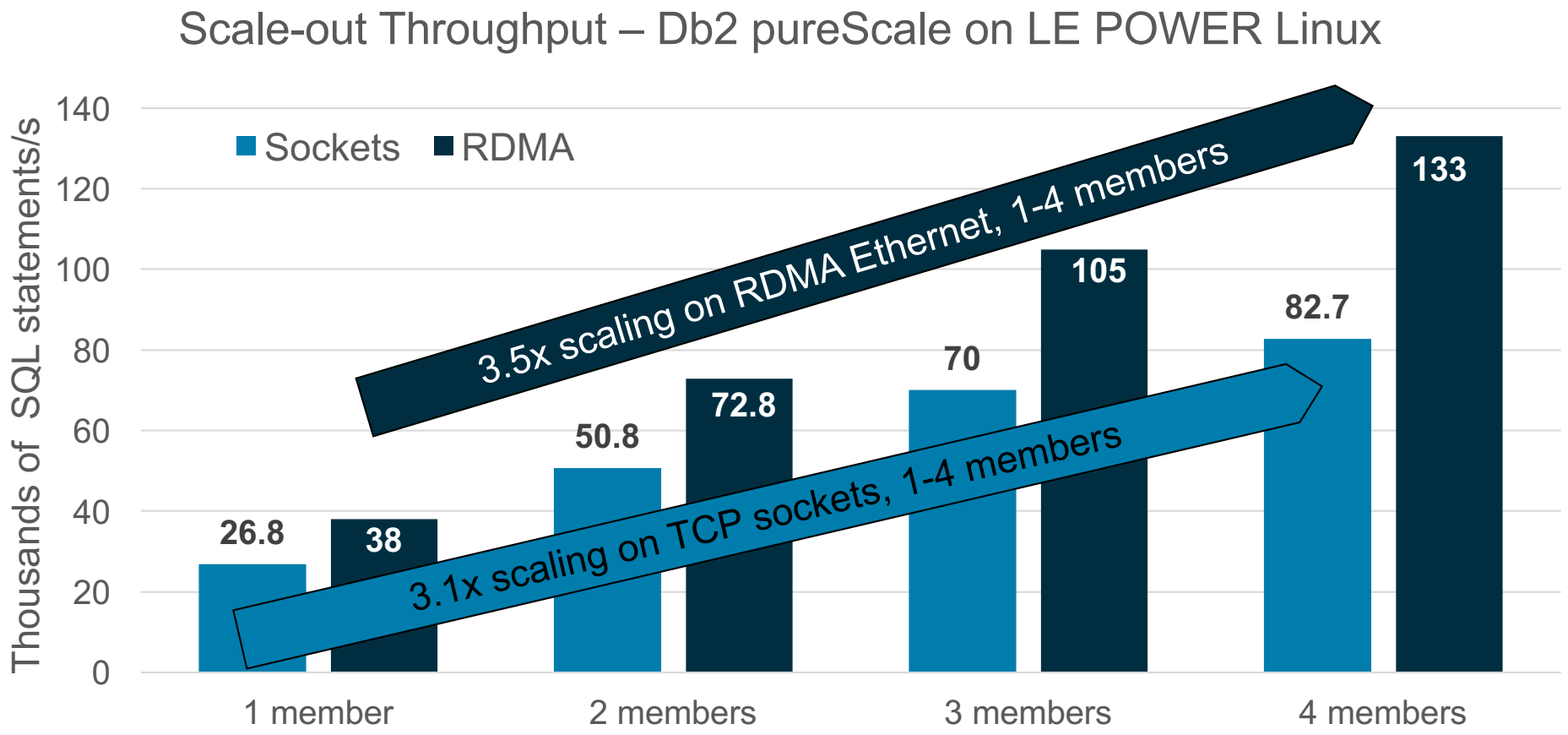
- **Looking at more challenging workload with more updates**
  - 1 update transaction for every 4 read transactions
  - Typical read/write ratio of many OLTP workloads
- **No cluster awareness in the application**
  - No routing of transactions to members
  - Demonstrate transparent application scaling
- **Redundant system**
  - 14 8-core p550s including duplexed CFs
- **Scalability remains above 90%**

# pureScale Scalability – 12 members




| | | | | |
|---|---|---|---|---|
| OLTP 80/20 R/W workload<br>No affinity | 12 8-core p550 members<br>64 GB, 5 GHz each | Duplexed CFs<br>on 2 additional 8-core p550s<br>64 GB, 5 GHz each | 20Gb/s IB HCAs<br>7874-024 IB Switch | DS8300 storage<br>576 15K disks<br>Two 4Gb FC Switches |

# Horizontal Scaling with Db2 pureScale on POWER Linux




- 80% read / 20% write OLTP workload
- POWER8 4c/32t, 160 GB LBP
- 10 Gb RoCE RDMA Ethernet / 10 Gb TCP sockets

Performance is based on measurements and projections using standard IBM benchmarks in a controlled environment. The actual throughput or performance that any user will experience will vary depending upon many factors, including considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve results similar to those stated here.

# pureScale Performance – SAP TRBK

- **Benchmark reflects the typical day-to-day operations of a retail bank**

- **Day processing:**
  - 90 million accounts and 1.8 billion postings
  - Over 56 million postings per hour

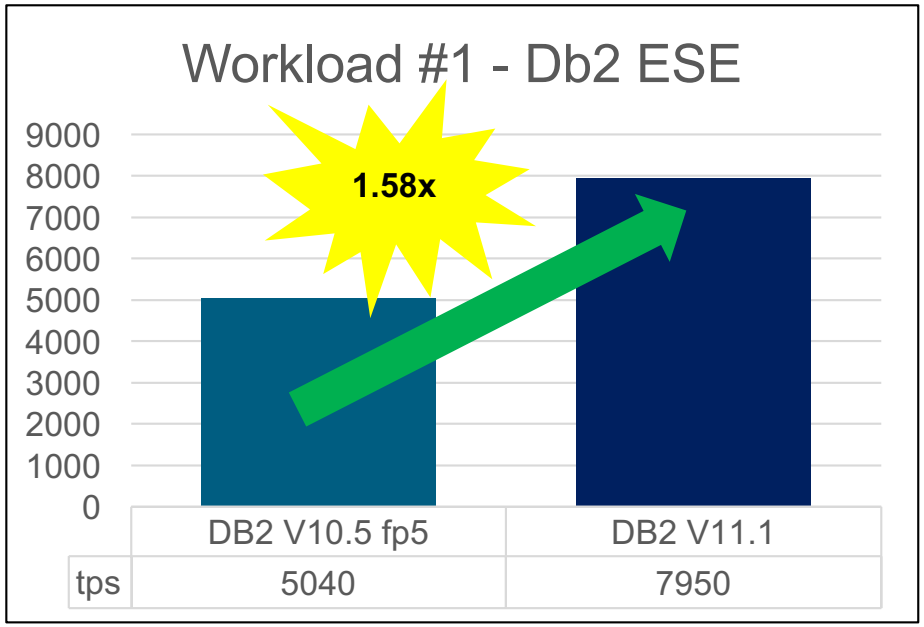

- **Night processing:**
  - Over 22 million accounts balanced per hour

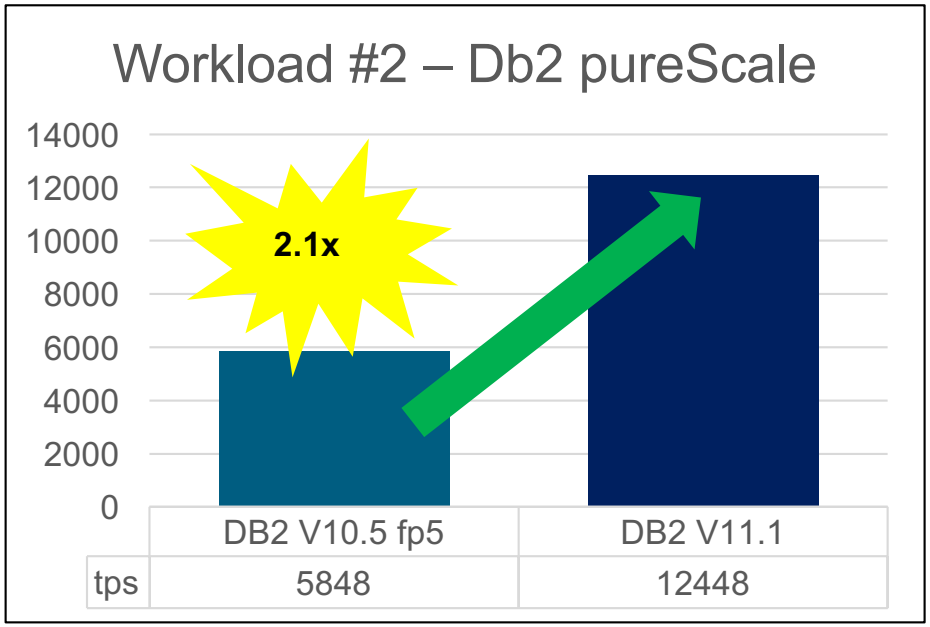
- **Benchmark Configuration**
  - Five 3690 X5 servers
  - Total database size: 9 TB uncompressed, 3.5 TB compressed

http://www.sap.com/solutions/benchmark/trbk3_results.htm

# Improved Performance for Highly Concurrent Workloads

- **Streamlined bufferpool latching protocol implemented in Db2 V11**
  - Reduces contention which can develop on large systems with many threads
  - Particularly helpful with transactional workloads



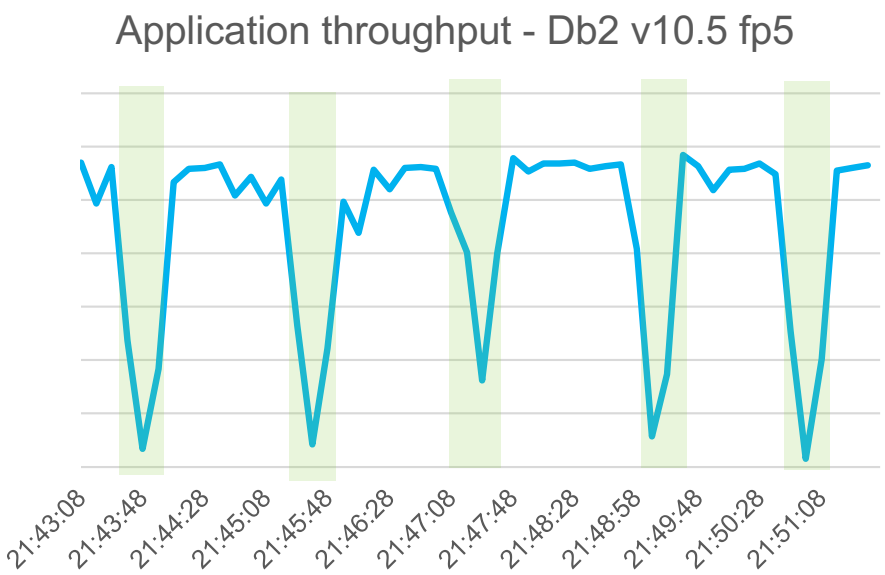

- Workload 1 based on an industry benchmark standard
- POWER7 32c, 512 GB

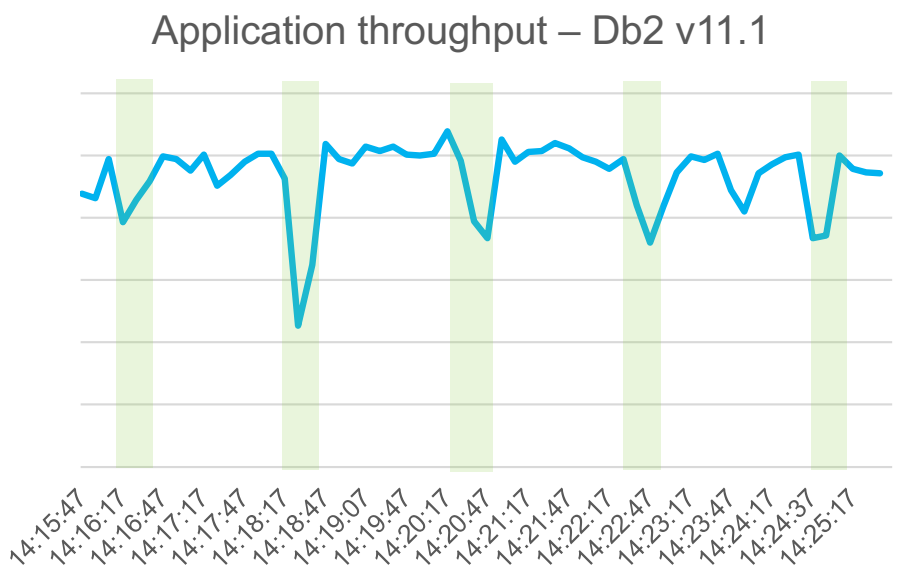
- Workload 2 implements a warehouse-based transactional order system
- 4 members, 2 CFs with 16c, 256 GB

Performance is based on measurements and projections using standard IBM benchmarks in a controlled environment. The actual throughput or performance that any user will experience will vary depending upon many factors, including considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve results similar to those stated here.

# Improved Table TRUNCATE Performance in pureScale

- **More efficient processing of Global Bufferpool (GBP) pages**
  - Speeds up truncate of permanent tables especially with large GBP sizes
  - Helps DROP TABLE and LOAD / IMPORT / INGEST with REPLACE option
  - Enables improved batch processing with these operations
- **Example**
  - Workload with INGEST (blue) and TRUNCATE (green) of an unrelated table
  - Db2 v11.1 has much smaller impact on OLTP workload than Db2 10.5 fp5



# Improved Performance for High Volume with XA Distributed Transactions

- **Enabled via 2 Db2 Registry Variables**
  - db2set DB2_SAL_SCA_NUM_COCLASSES=1024
  - db2set DB2_SAL_SCA_NON_XA_COCLASSES=true
    - These settings distribute the CF's system communication area data structures across 1024 cast out classes (vs the default of 1)
    - Requires a full cluster shutdown (is not rolling update compatible)

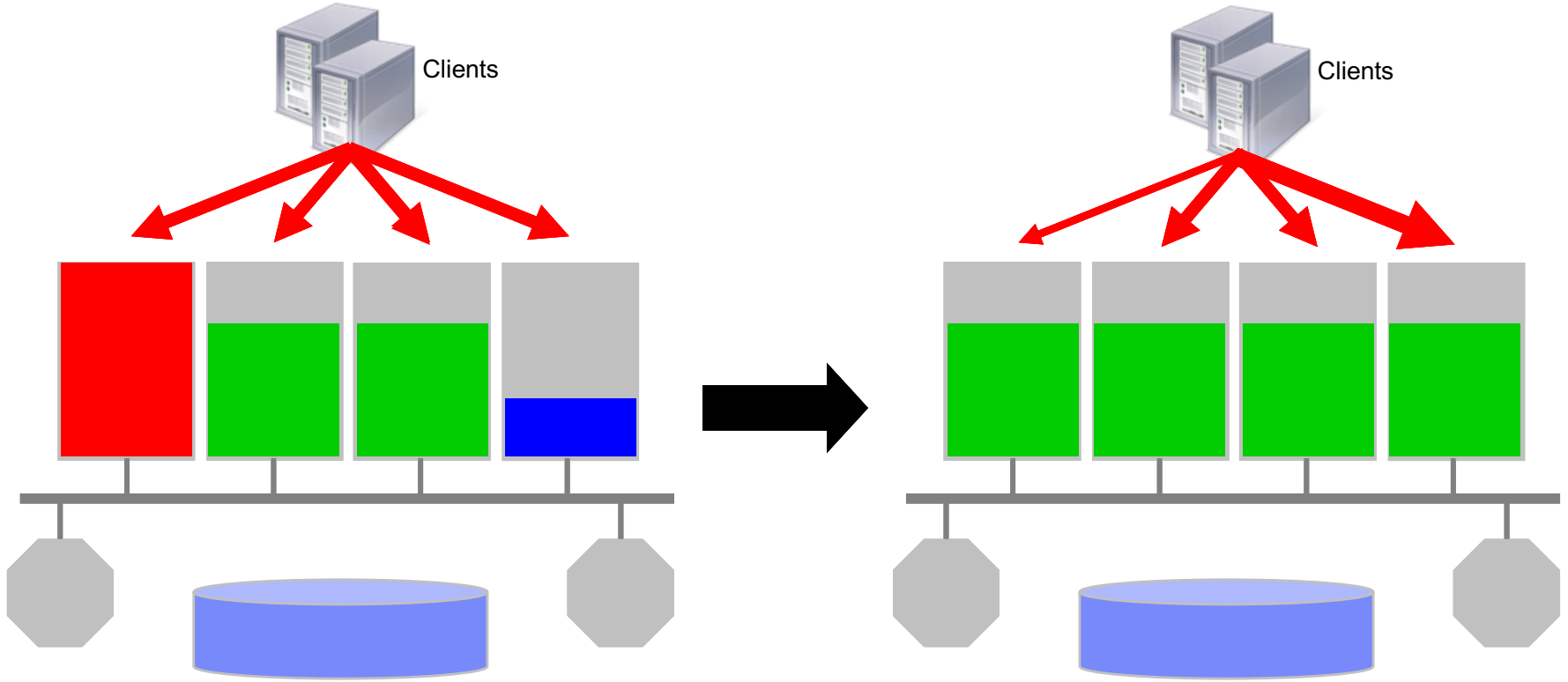
- **In one extreme case, transaction throughput increased by 60%!**
  - And CF response times for XA-related services, were reduced by 80%

- **Refer to APAR IT19994 for more details**

# Multi-tenancy & Application Transparency with Db2 pureScale

# pureScale Workload Balancing

- Run-time load information used to automatically balance load across the cluster
- Failover – automatically distribute workload among remaining “live” members
- Fallback – automatically start leveraging “new live” member upon return



# Multi-tenancy with Db2 pureScale

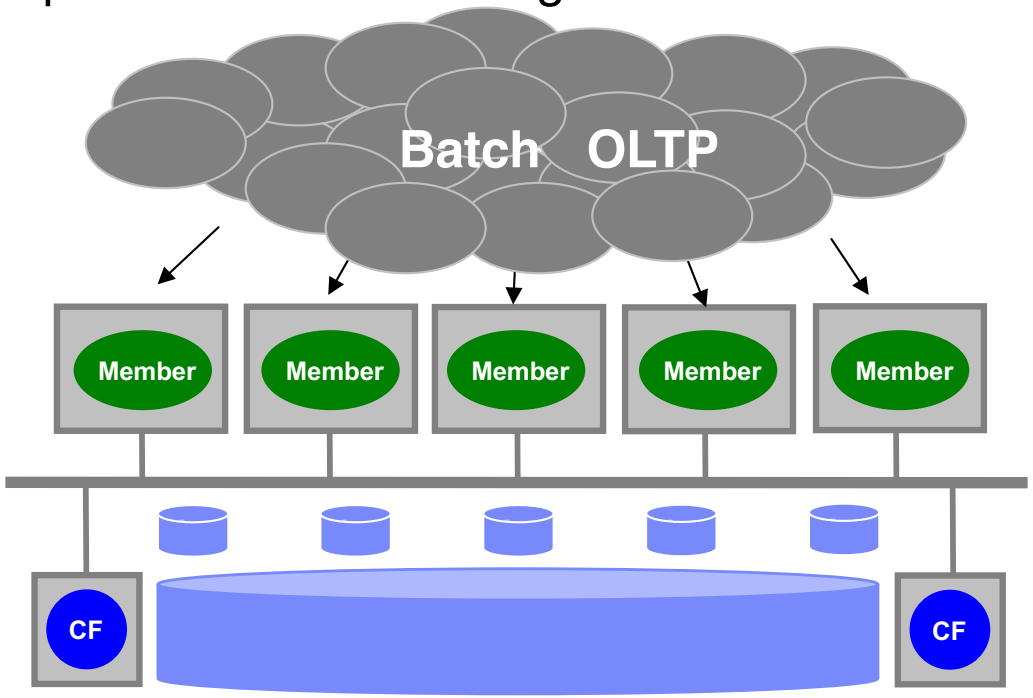
- **Reduce management overhead and improve resource utilization through consolidation of workloads/databases**
  - Individual databases typically have high availability requirements but may or may not require scale-out

- **Multi-tenancy characteristics of a Db2 pureScale environment**
  - Single pureScale instance per set of hosts
  - Multiple databases (up to 200 active) supported per pureScale instance
  - Able to create multiple schemas (logical databases) in a database
  - Isolate workloads on different members using member subsets & affinitization
  - Self-tuning memory management acts and adapts independently on each member
  - Explicit Hierarchical Locking (EHL) reduces data sharing overhead for independent workloads running on different members
  - Workload management enforced on each member

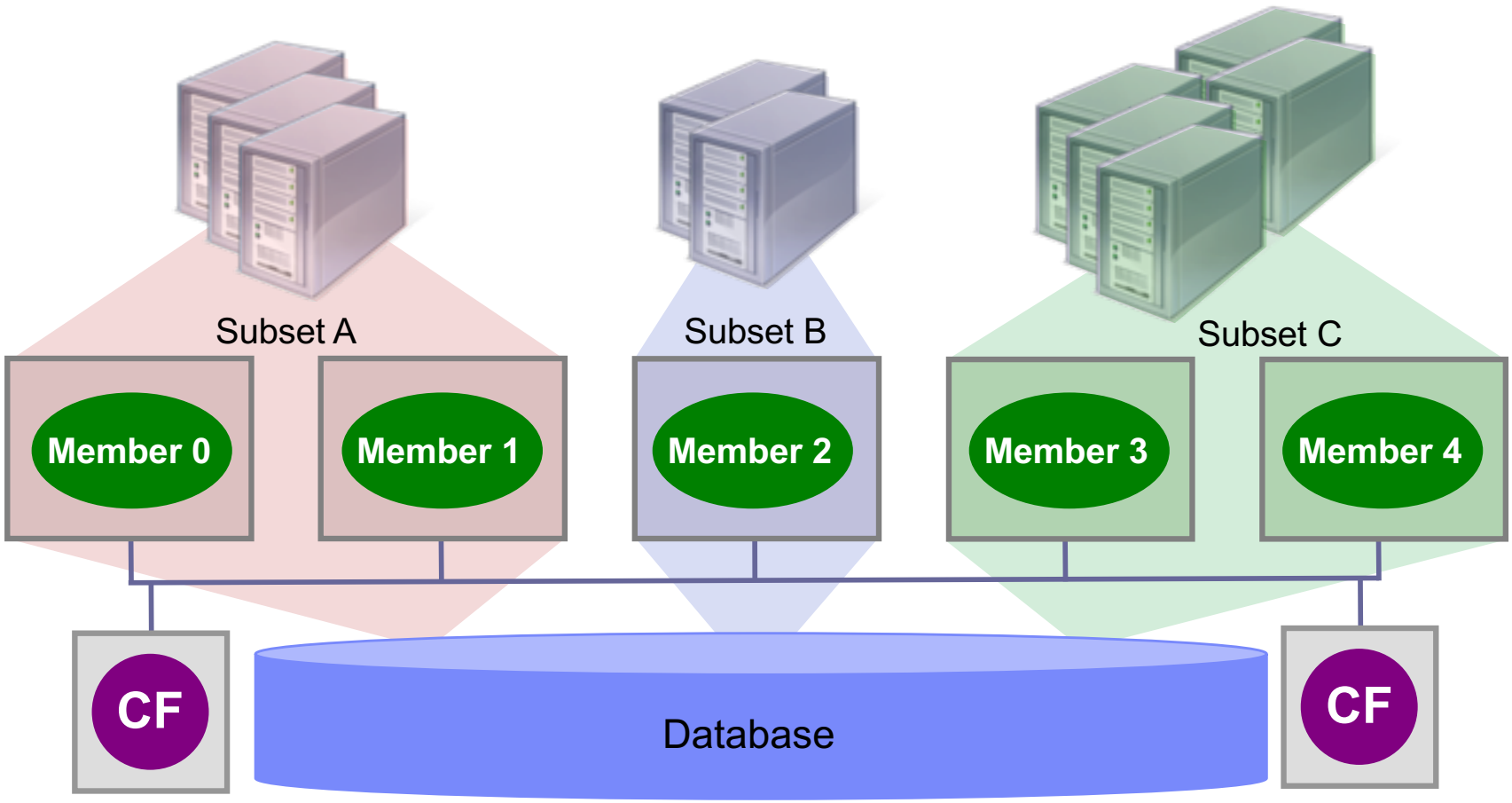
# pureScale Multi-Tenancy - Member Subsets

- **Applications can be workload balanced across all members or a subset of members in the instance**

- **Member subsets allow for**
  - Isolation of different workloads (e.g. batch and transactional) from each other within a single database
  - Workloads for multiple databases in a single instance isolated from each other

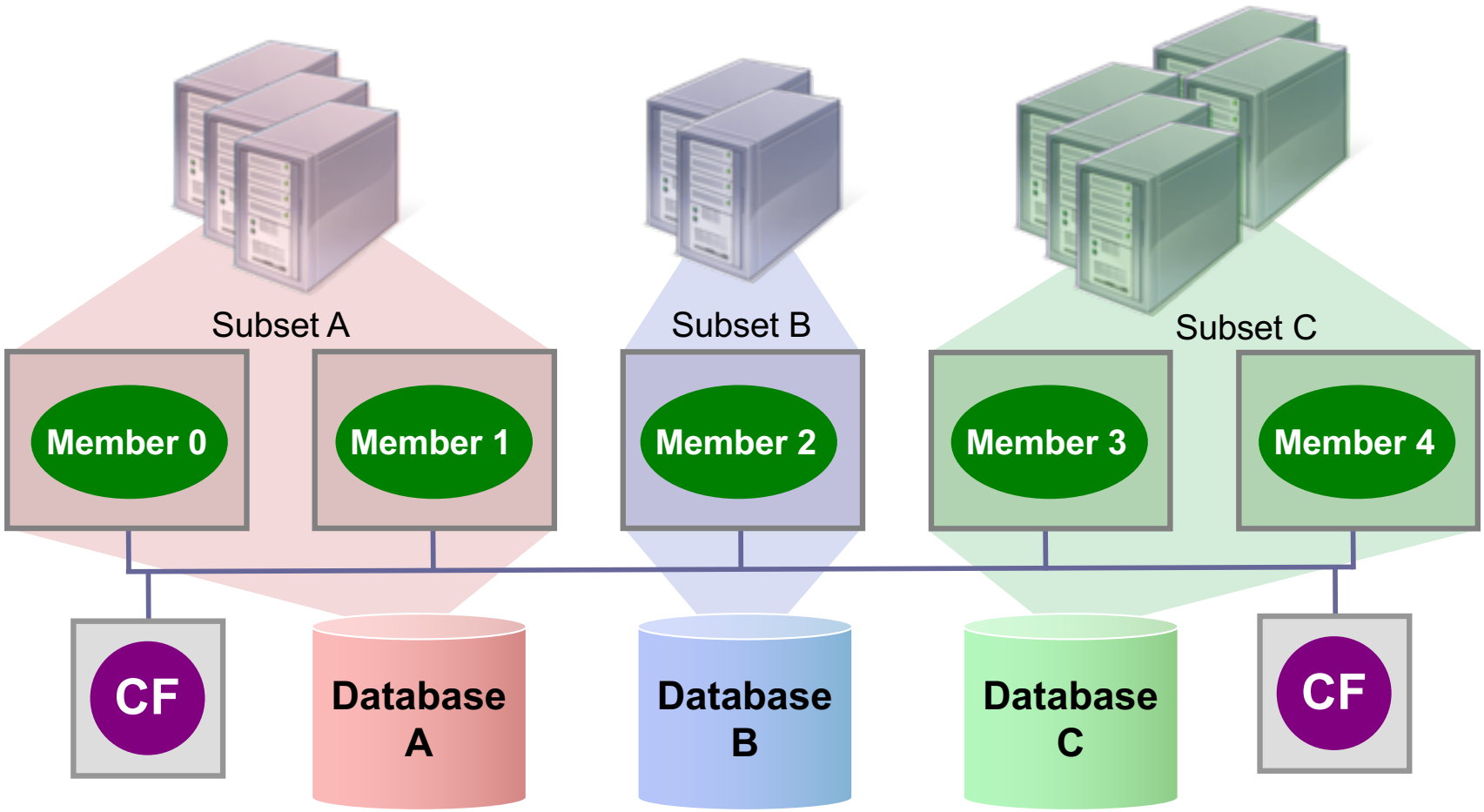


# Multi-Tenancy - Consolidation of Multiple Workloads



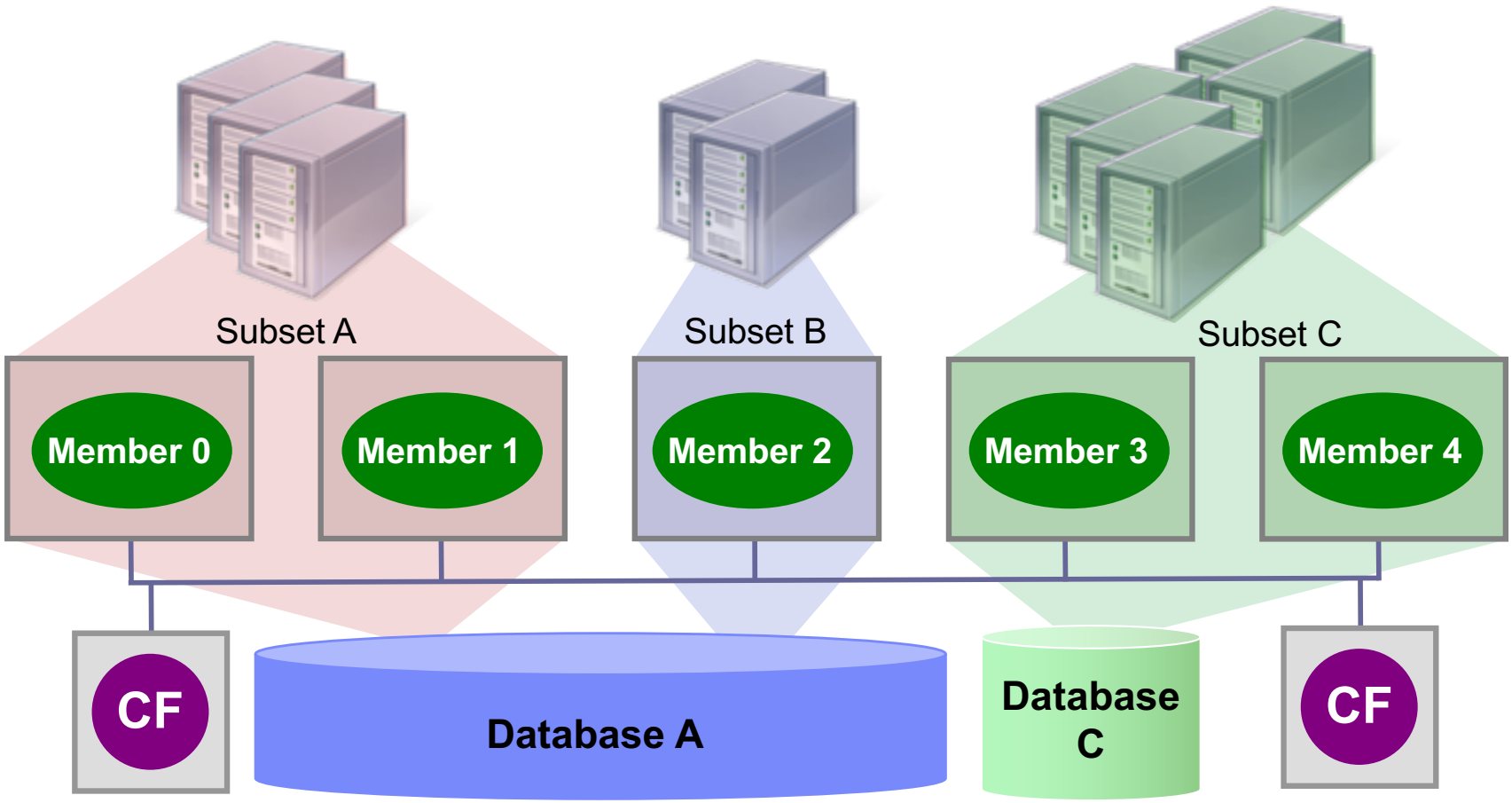

- Single pureScale environment to manage
- Configure the environment differently on members to meet workload SLAs
- Built-in high availability for all workloads
- Scale-out workloads by adding new members to subsets (fully online)

# Multi-Tenancy - Consolidation of Multiple Databases




- Single pureScale environment to manage
- Configure the environment differently on members to meet workload SLAs
- Built-in high availability for all databases
- Scale-out workloads by adding new members to subsets (fully online)

# Multi-Tenancy - Consolidation Combination




- Single pureScale environment to manage
- Configure the environment differently on members to meet workload SLAs
- Built-in high availability for all databases
- Scale-out workloads by adding new members to subsets (fully online)

# Unified Workload Balancing with pureScale

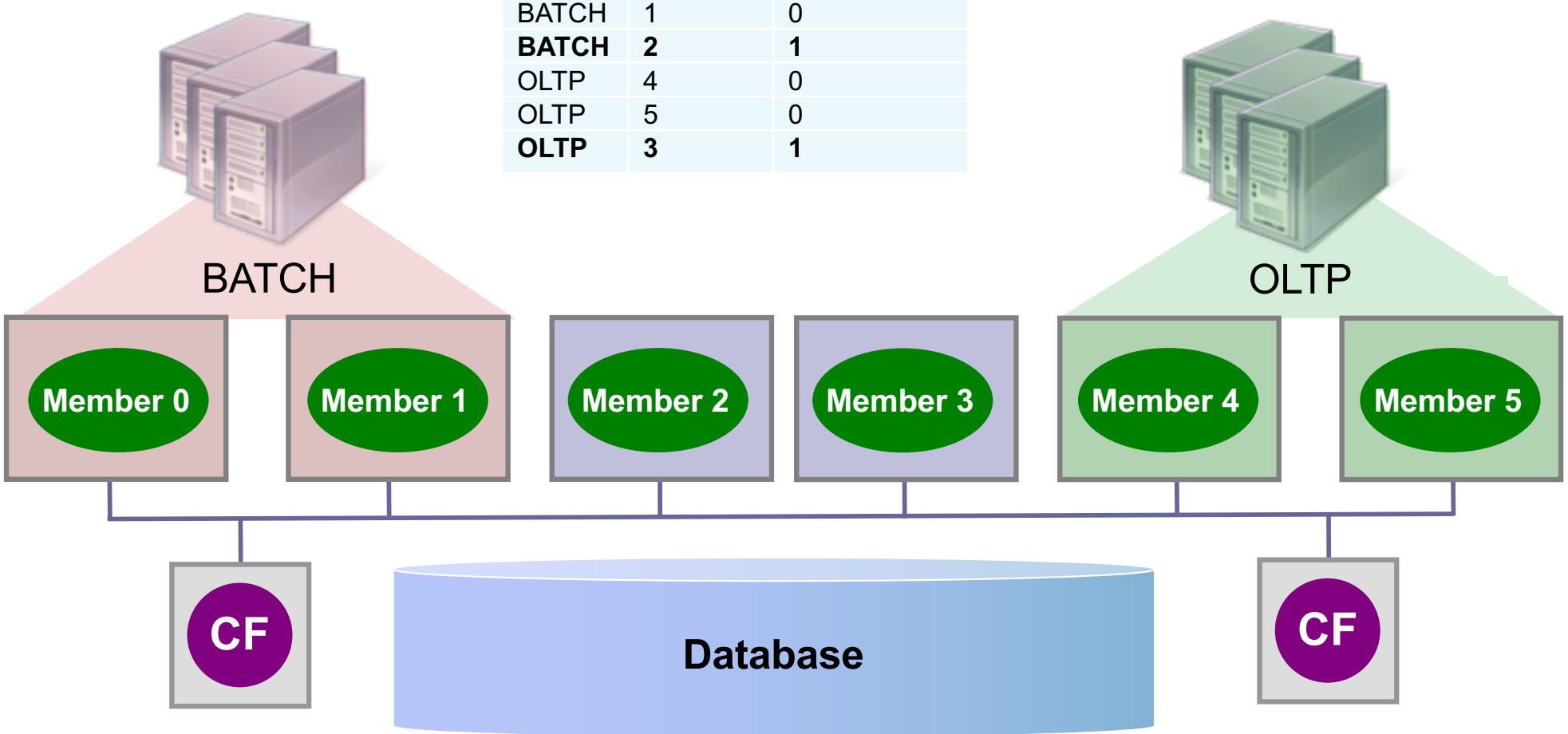
- **Version 11.1 extends the configuration options for member subsets allowing the user to explicitly define alternate members for a subset**
  - This allows users currently using client affinity to move their configuration to using member subsets and failover priority so that they can exploit the new benefits such as dynamic server side reconfiguration
  - Simplification to setting up client affinity with having control at the Server vs client – no need to update db2dsdriver.cfg
- **Failover priority for member subsets added**
  - You can explicitly define the members that are part of the alternate member list by using the FAILOVER_PRIORITY attribute in the WLM_ALTER_MEMBER_SUBSET procedure
  - Members with a failover priority of 0 (the default priority) are considered primary members and members with failover priority 1-254 are considered alternative members
  - The number of primary members in the subset defines the minimum number of members to service an application
  - If FAILOVER_PRIORITY is not specified, default priority of 0 is used

# Member Subsets : FAILOVER PRIORITY

```
CALL SYSPROC.WLM_ALTER_MEMBER_SUBSET('BATCH', NULL,'(ADD 2 FAILOVER_PRIORITY 1)');
CALL SYSPROC.WLM_ALTER_MEMBER_SUBSET('OLTP',  NULL,'(ADD 3 FAILOVER_PRIORITY 1)');
```

| SUBSET | MEMBER | FAILOVER_ PRIORITY |
|---|---|---|
| BATCH | 0 | 0 |
| BATCH | 1 | 0 |
| **BATCH** | **2** | **1** |
| OLTP | 4 | 0 |
| OLTP | 5 | 0 |
| **OLTP** | **3** | **1** |

This information is available from `SYSCAT.MEMBERSUBSETMEMBERS` and `db2pd –membersubsetstatus -detail`

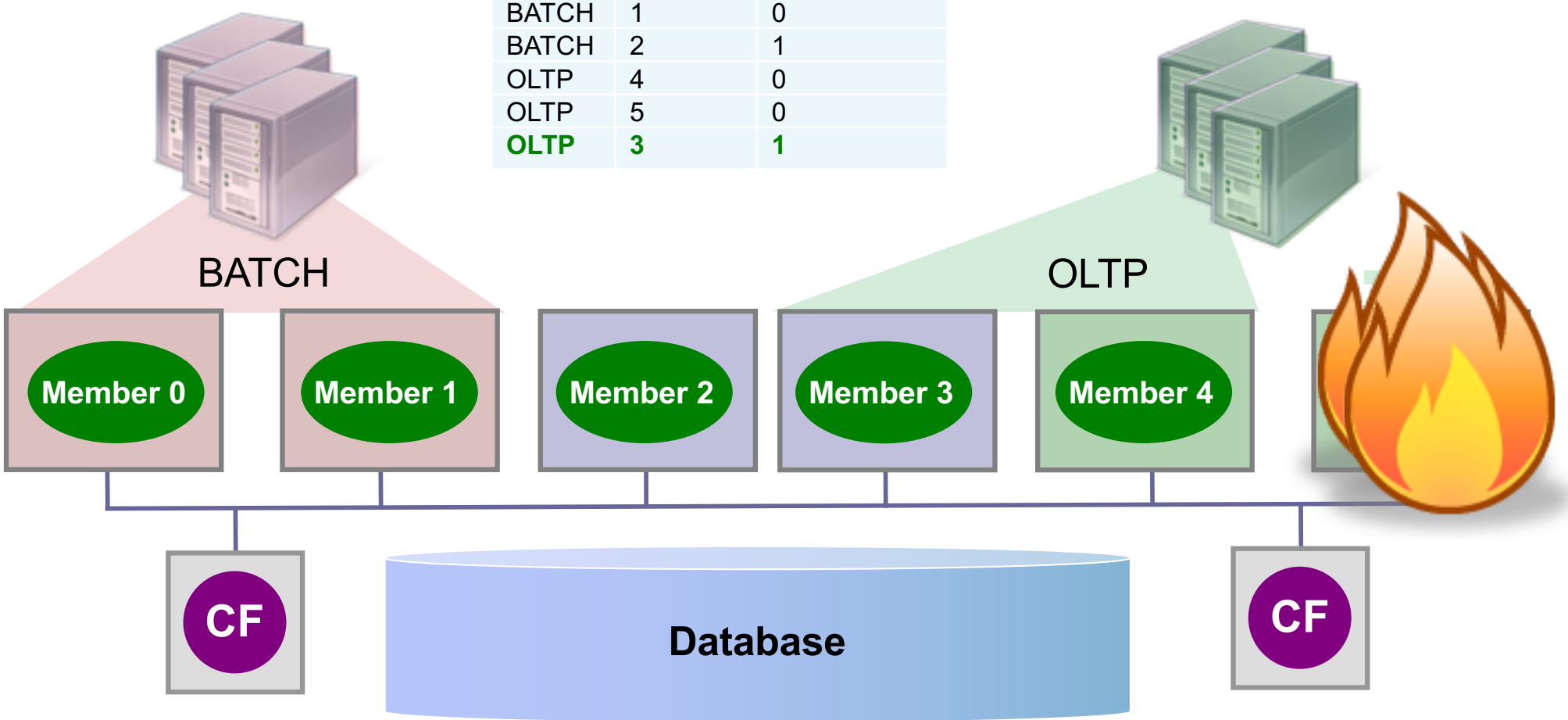


# Member Subsets : FAILOVER PRIORITY

```
CALL SYSPROC.WLM_ALTER_MEMBER_SUBSET('BATCH', NULL,'(ADD 2 FAILOVER_PRIORITY 1)');
CALL SYSPROC.WLM_ALTER_MEMBER_SUBSET('OLTP',  NULL,'(ADD 3 FAILOVER_PRIORITY 1)');
```

| SUBSET | MEMBER | FAILOVER_ PRIORITY |
|---|---|---|
| BATCH | 0 | 0 |
| BATCH | 1 | 0 |
| BATCH | 2 | 1 |
| OLTP | 4 | 0 |
| OLTP | 5 | 0 |
| **OLTP** | **3** | **1** |

This information is available from `SYSCAT.MEMBERSUBSETMEMBERS` and `db2pd –membersubsetstatus -detail`

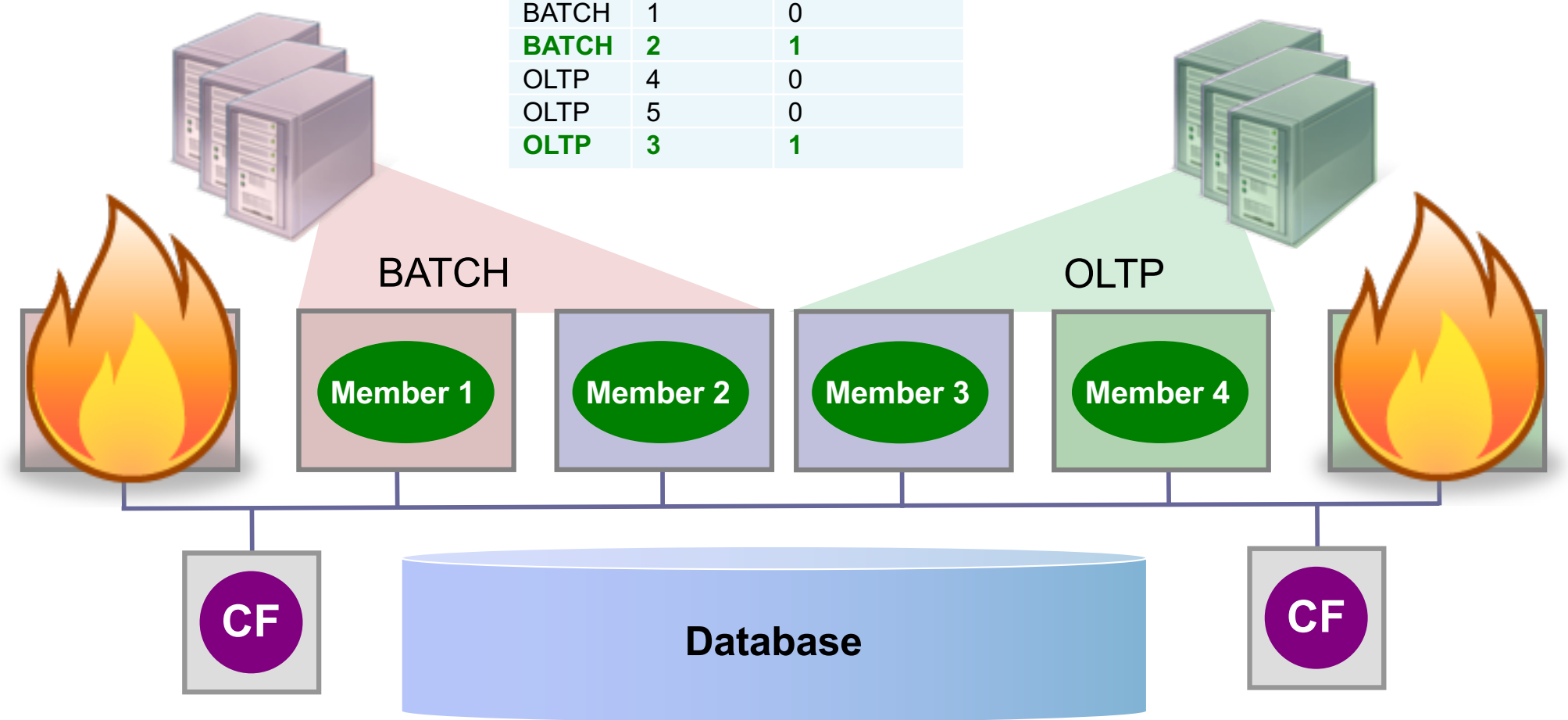


# Member Subsets : FAILOVER PRIORITY

```
CALL SYSPROC.WLM_ALTER_MEMBER_SUBSET('BATCH', NULL,'(ADD 2 FAILOVER_PRIORITY 1)');
CALL SYSPROC.WLM_ALTER_MEMBER_SUBSET('OLTP',  NULL,'(ADD 3 FAILOVER_PRIORITY 1)');
```

| SUBSET | MEMBER | FAILOVER_ PRIORITY |
|---|---|---|
| BATCH | 0 | 0 |
| BATCH | 1 | 0 |
| **BATCH** | **2** | **1** |
| OLTP | 4 | 0 |
| OLTP | 5 | 0 |
| **OLTP** | **3** | **1** |

This information is available from `SYSCAT.MEMBERSUBSETMEMBERS` and `db2pd –membersubsetstatus -detail`

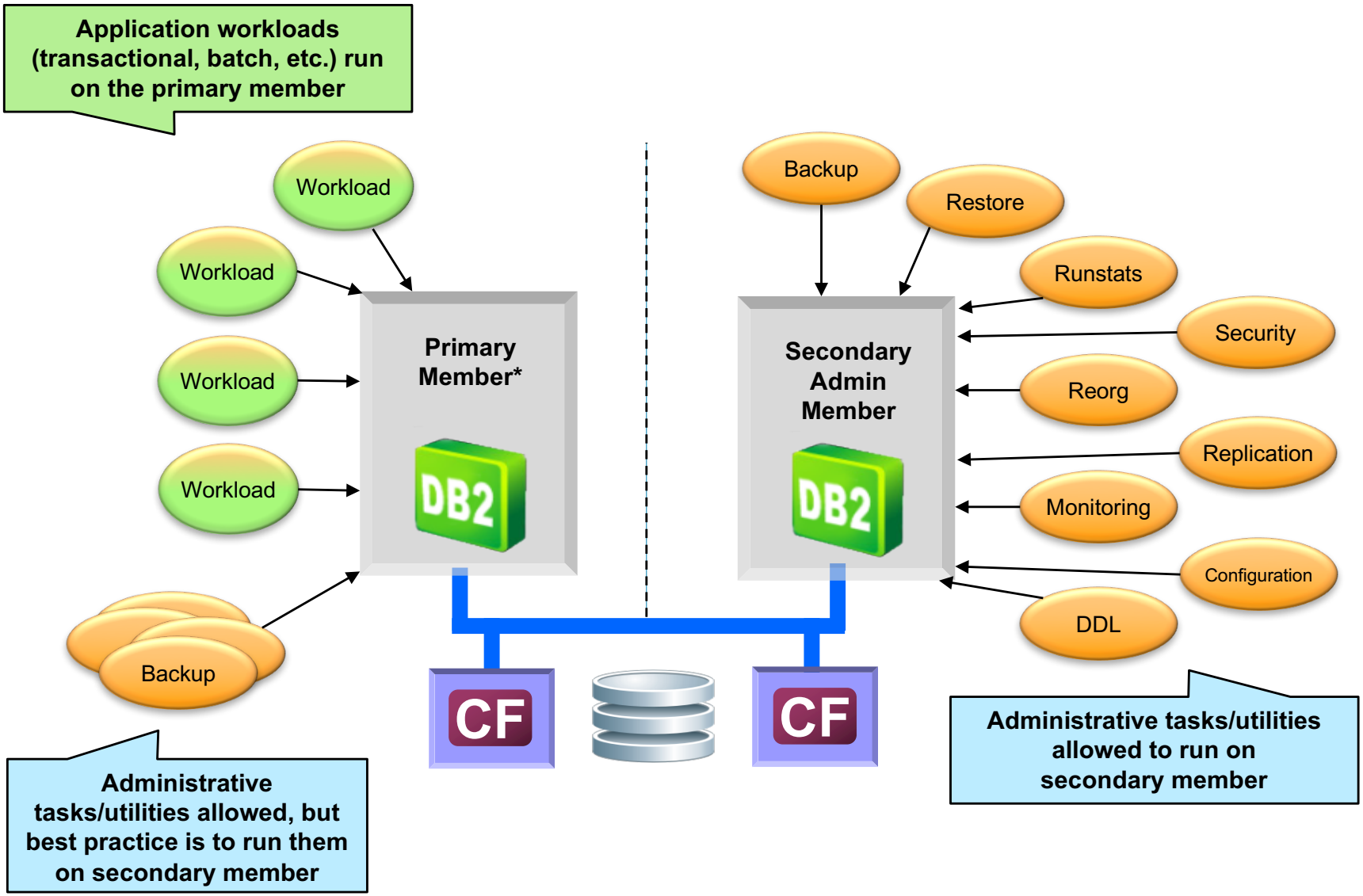


# Db2 pureScale Included in Packaging

- **Low cost active/admin licensing where one Db2 member has minimal licensing and the other Db2 member(s) fully licensed**
  - All application workloads are directed to the primary active member(s)
    - Sometimes referred to as the “primary" member
  - Utilities and admin tasks allowed on the secondary admin member
    - Admin member licensed as warm standby (e.g. 100 PVUs or 1 VPC)
    - Great for off-loading backups from primary members

- **Db2 pureScale active/active available in Db2 Advanced Editions, including new Direct Advanced Edition**

# Licensing - Db2 pureScale Active/Admin Model




*can have up to 127 Primary Members

# Active/Admin - Directing Workloads Using Member Subsets

- **Recommendation is to use member subsets for application routing**
  - However, any method that directs workloads so that compliance is maintained can be used

- **For each database in the cluster, create a member subset for each of the two members**
  - Create subsets as inclusive (default)
  - All databases in cluster must use the same primary member

- **Example:**

```
CALL SYSPROC.WLM_CREATE_MEMBER_SUBSET(
   'SALES_TRANS_SUBSET',
   '<databaseAlias>SALES_T</databaseAlias>',
   '( 0 )')

 CALL SYSPROC.WLM_CREATE_MEMBER_SUBSET(
   'SALES_ADMIN_SUBSET',
   '<databaseAlias>SALES_A</databaseAlias>',
   '( 1 )')
```

**Subset name**

**Database alias**

**Member associated with subset**

# Active/Admin - Directing Workloads Using Member Subsets

- **Typical pureScale client configuration steps are followed**
  - Clients configured to connect to database alias associated with primary member subset on corresponding member host
  - Alternate server points to secondary member host
  - Transaction-level workload balancing recommended
    - With database-level workload balancing, applications may not failback to primary member in a timely manner after maintenance operation completes
  - Configuration via Java properties or db2dsdriver.cfg file depending on type of application

- **Clients are sent to primary member but know how to automatically move to secondary member if failover required**

# Active/Admin - Role Switching

- **Switching of primary and secondary member roles not required, but may be desired in some situations**
  - For example, prior to fixing and restarting a failed primary member
    - Becomes secondary member after started, so no need for apps to be rerouted a second time

- **Accomplished by switching members in member subsets**
  - Applications drained from dropped member via process of workload balancing
    - Applications not forced off when member dropped from subset

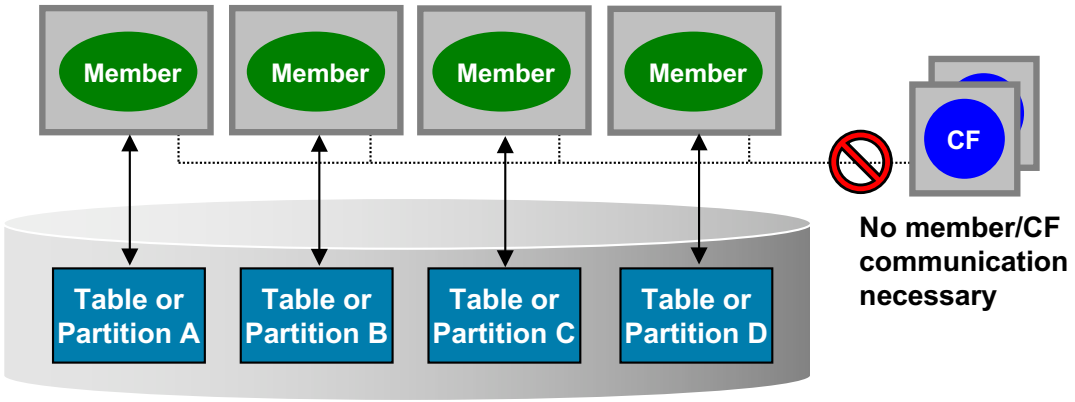
```
CALL SYSPROC.WLM_ALTER_MEMBER_SUBSET
   ('SALES_TRANS_SUBSET', NULL, '(drop 0, add 1)');

CALL SYSPROC.WLM_ALTER_MEMBER_SUBSET
   ('SALES_ADMIN_SUBSET', NULL, '(drop 1, add 0)');
```

**Drop current member for subset and add other member**

- **If role switching done, must be done for all databases**

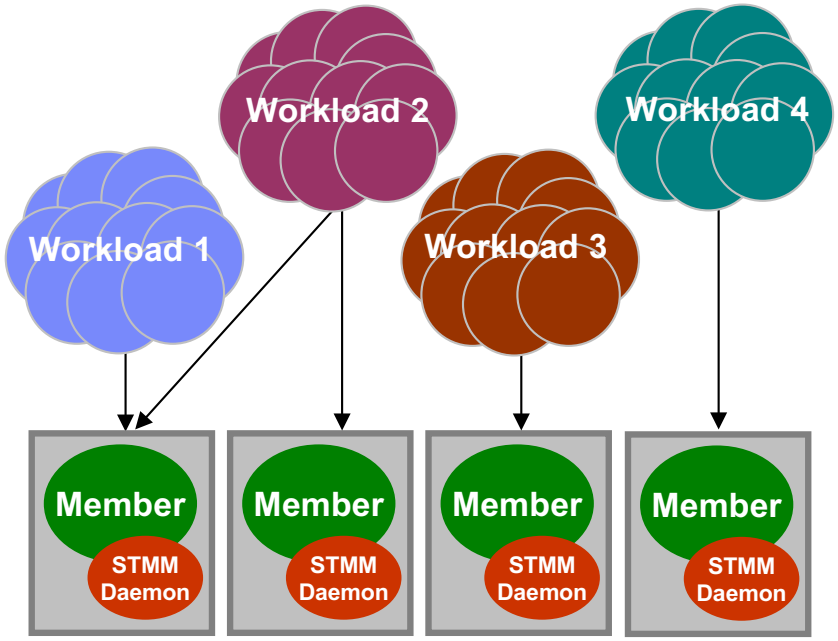
# pureScale Multi-Tenancy - Explicit Hierarchical Locking

- **Designed to remove data sharing costs for tables/partitions that are only accessed by a single member**
  - Avoids CF communication if object sharing not occurring
  - Automatic detection of data sharing access and conversion to normal data sharing mode on a per table / table partition basis
  - Can move back to non-data sharing mode after a period of time with no conflicting interest in table / partition
- **Target scenarios**
  - Workload affinitization or workload consolidation and application affinitization
- **Enabled via `OPT_DIRECT_WRKLD` database configuration parameter**
  - Detection of data access patterns happens automatically and EHL will kick in when data is not being shared after configuration parameter set



# Multi-Tenancy: Self-Tuning Memory Management (STMM)

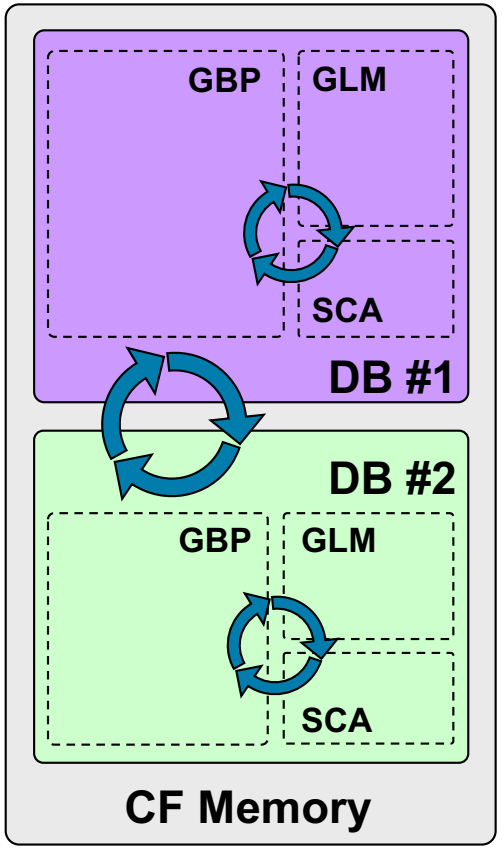
- **Db2 pureScale allows per-member STMM tuning**
  - Workload consolidation
  - Multi-tenancy
  - Batch workloads
  - Affinitized workloads



# Multi-Tenancy: CF Self-Tuning Memory

- **CF memory is optimally distributed between consumers based on workload**
  - Less administrative overhead for DBA, with reduction in memory monitoring and management

- **Can function at two levels**
  - Dynamic distribution of CF memory between multiple databases in an instance
  - Dynamic distribution of database's CF memory between its consumers
    - Group buffer pool (GBP)
    - Global lock manager (GLM)
    - Shared communication area (SCA)




- **Beneficial for multi-tenant environments**
  - where multiple databases are consolidated in a Db2 pureScale cluster

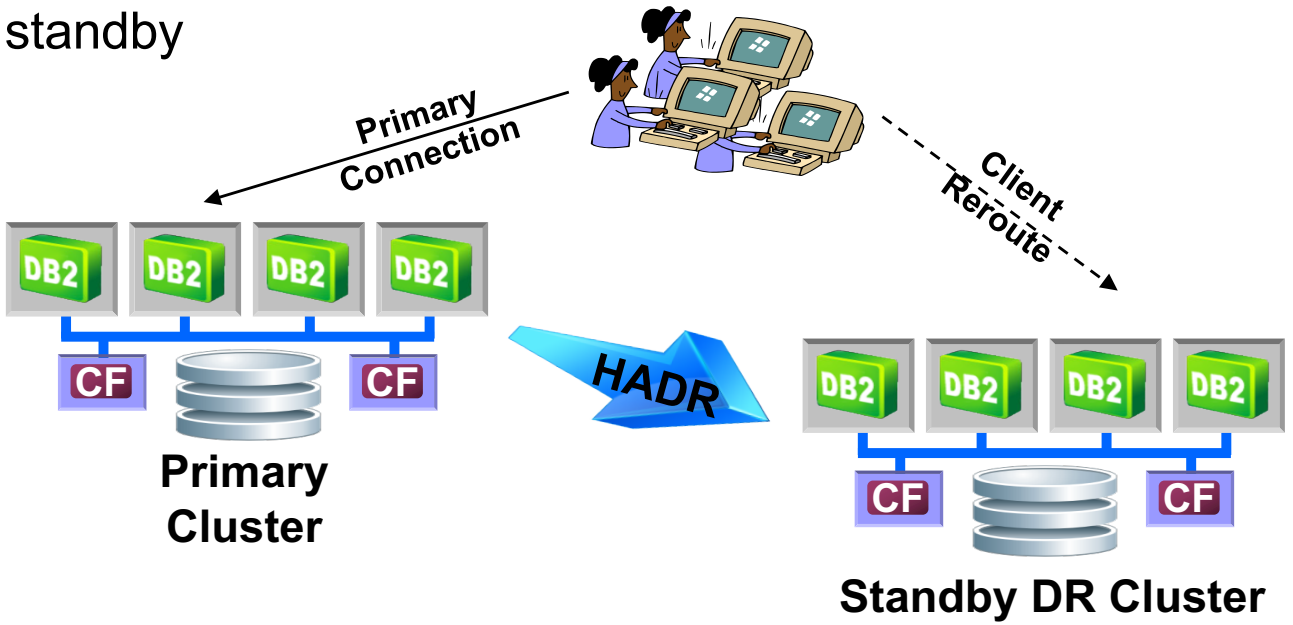
# Disaster Recovery with Db2 pureScale

# Disaster Recovery Options for pureScale

- **Variety of disaster recovery options to meet your needs**
  - HADR
  - Geographically Dispersed pureScale Cluster (GDPC)
  - Storage Replication
  - Q Replication
  - InfoSphere Change Data Capture (CDC)
  - Manual Log Shipping

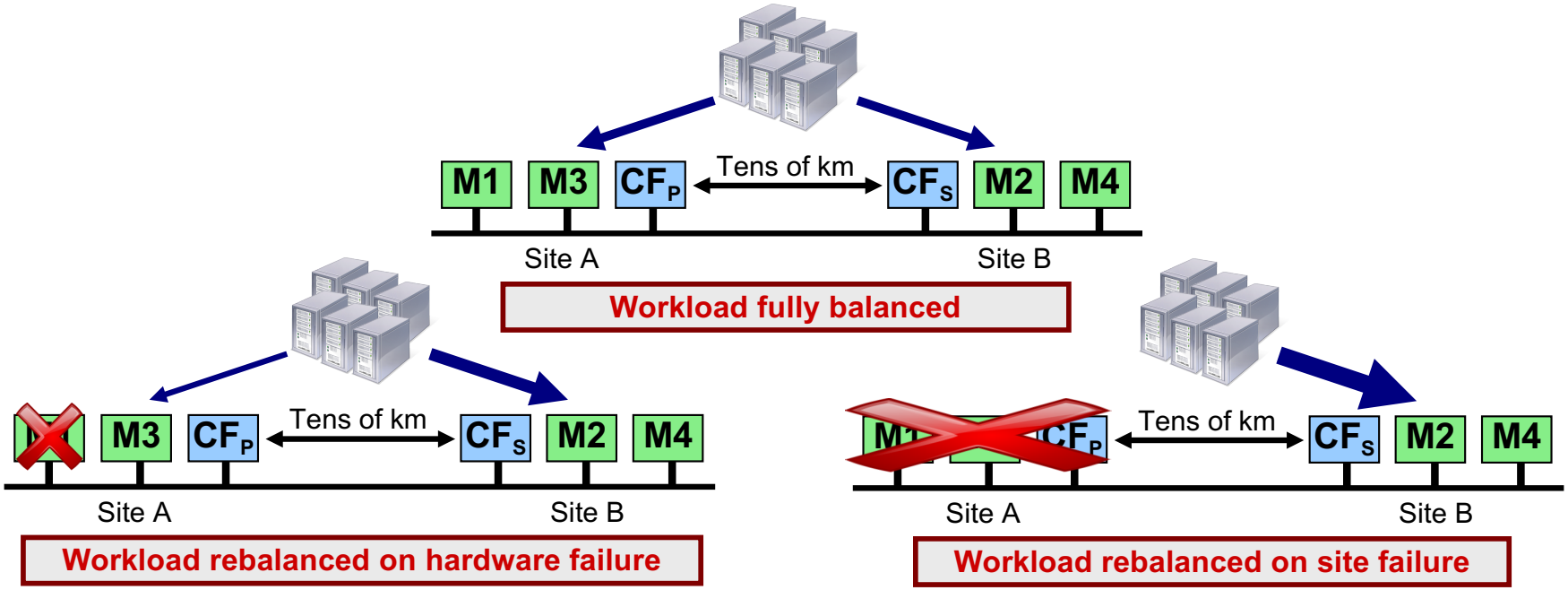
# HADR with Db2 pureScale

- **Integrated disaster recovery solution**
  - Simple to setup, configure, and manage
  - An integrated zero data loss (i.e. RPO=0) disaster recovery solution
- **HADR support with pureScale includes:**
  - All modes - SYNC, NEARSYNC, ASYNC and SUPERASYNC
  - Time delayed apply, Log spooling
  - Both non-forced (role switch) and forced (failover) takeovers
  - Table space recovery – only affected table space(s) need be restored on standby

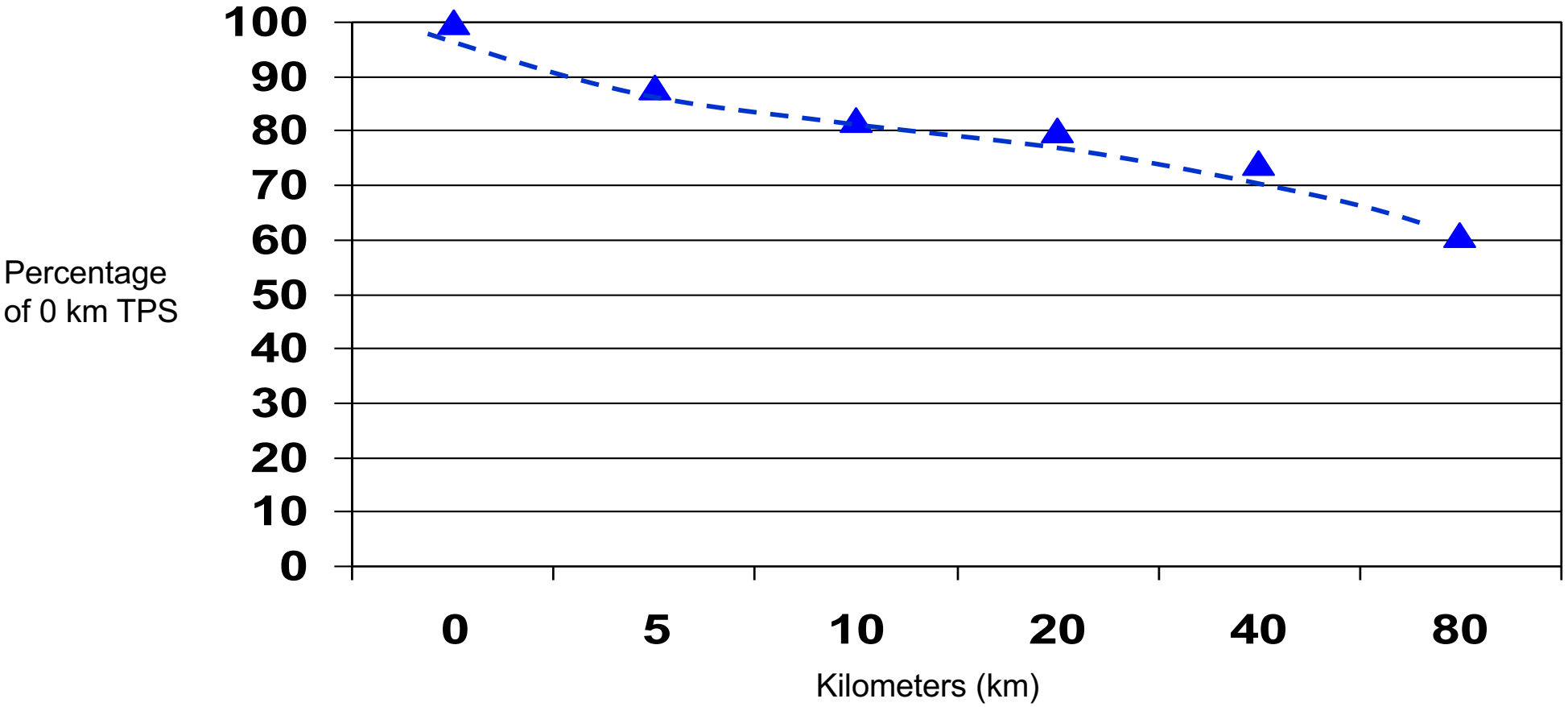


# Geographically Dispersed pureScale Clusters (GDPC)

- **A "stretch" or geographically dispersed pureScale cluster (GDPC) spans two sites**
  - Provides active/active DR for one or more shared databases across the cluster
  - Enables a level of DR support suitable for many types of disasters (e.g. fire, data center power outage)
  - Supported with RDMA (10 GE RoCE) and TCP/IP
- **Both sites active and available for transactions during normal operation**
- **On failures, client connections are automatically redirected to surviving members**
  - Applies to both individual members within sites and total site failure

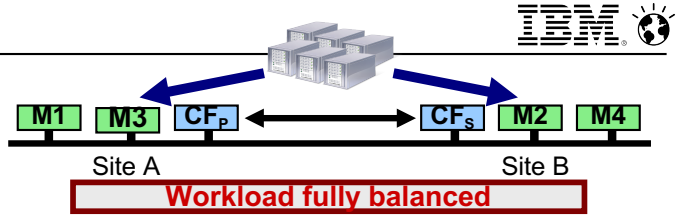


# Performance Sensitivity to Distance: Sample Data

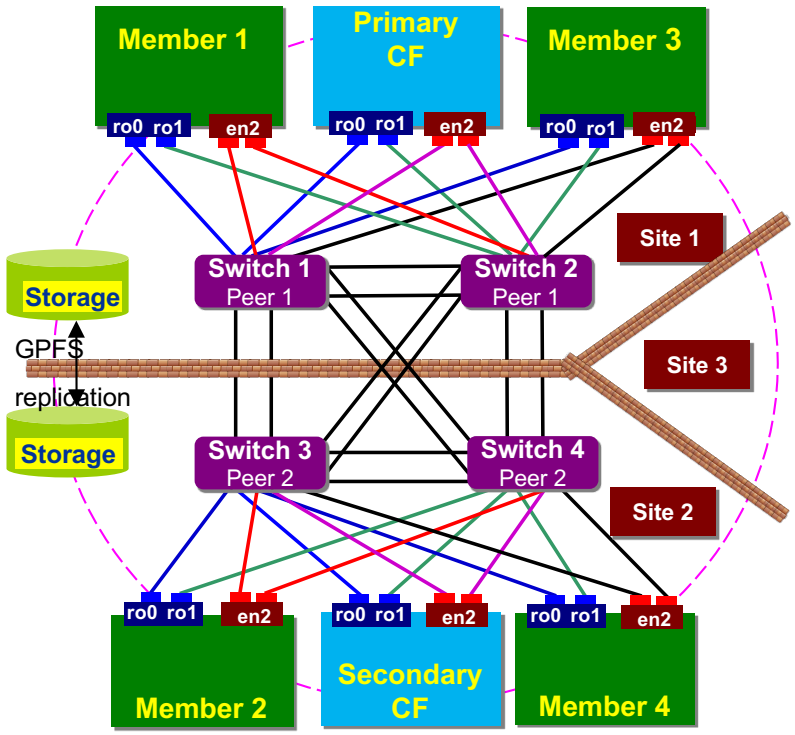



Internal OLTP workload with 70% reads 30% writes

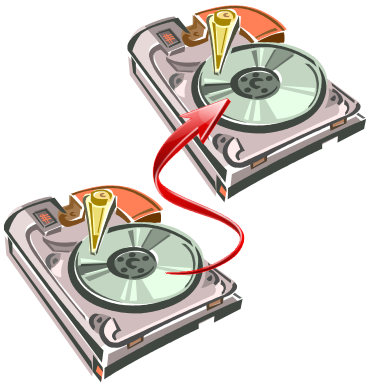
# GDPC Support Enhancements




- **Db2 V11 adds improved high availability for Geographically dispersed Db2 pureScale clusters (GDPC) for both RoCE & TCP/IP**
  - Multiple adapter ports per member and CF to support higher bandwidth and improved redundancy at the adapter level
  - Dual switches can be configured at each site to eliminate the switch as a site-specific single point of failure (i.e. 4-switch configuration)



# Storage Replication



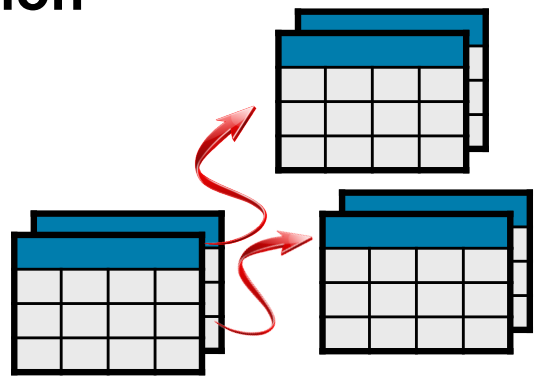
- **Uses remote disk mirroring technology**
  - Maximum distance between sites is typically 100s of km (for synchronous, 1000s of km for asynchronous)
  - For example: IBM Metro Mirror, EMC SRDF

- **Transactions run against primary site only, DR site is passive**
  - If primary site fails, database at DR site can be brought online
  - DR site must be an identical pureScale cluster with matching topology

- **All data and logs must be mirrored to the DR site**
  - Synchronous replication guarantees no data loss
  - Writes are synchronous and therefore ordered, but “consistency groups” are still needed
    - If failure to update one volume, don’t want other volumes to get updated (leaving data and logs out of sync)

# Q Replication/CDC

- **High-throughput, low latency logical data replication**
  - Distance between sites can be up to thousands of km



- **Asynchronous replication**

- **Includes support for:**
  - Delayed apply
  - Multiple targets
  - Replicating a subset of data
  - Data transformation
  - Bi-directional replication - updates on both Primary and DR sites

- **pureScale Member topology does not have to match between Primary and DR sites**

# Manual Log Shipping

- **"Home grown" (user managed) active/passive DR solution**
- **Database on standby system is kept in a perpetual "rollforward in progress" state**
  - Roll forward command is executed repeatedly as log files become available
    - Can choose to incorporate a time delay between the primary and standby
  - `STOP` option is used to bring it out of roll forward state if primary fails
- **Use log files from the archive location and/or use scripts to manage the transfer of log files to the standby site**
  - Recommended that archive location should be geographically separate from the primary site
  - Consider using the `ARCHIVE LOG` command if logs are being filled too slowly
- **Two ways to initialize a standby**
  - Restore of a backup image taken on the primary
  - Using the `db2inidb` command with the `STANDBY` option against a split mirror copy of the primary
- **Operations that are not logged will not be replayed on the standby database**

# Comparison of pureScale Disaster Recovery Options

| | HADR | Storage Replication | | GDPC | Q Replication / CDC | Manual Log Shipping |
|---|---|---|---|---|---|---|
| | | Sync | Async | | | |
| **Active/active DR** | No | No | No | Yes | Yes | No |
| **Synchronous** | No | Yes | No | Yes | No | No |
| **Requires matching pureScale topology at DR site** | Yes | Yes | Yes | n/a | No | Yes |
| **Delayed apply** | Yes | No | No | No | Yes | Yes |
| **Multiple DR target sites** | No | No | No | No | Yes | Yes |
| **Maximum distance between sites** | 1000s km | 100s km | 1000s km | 10s km | 1000s km | 1000s km |

# Advanced Application Capabilities with Db2 pureScale

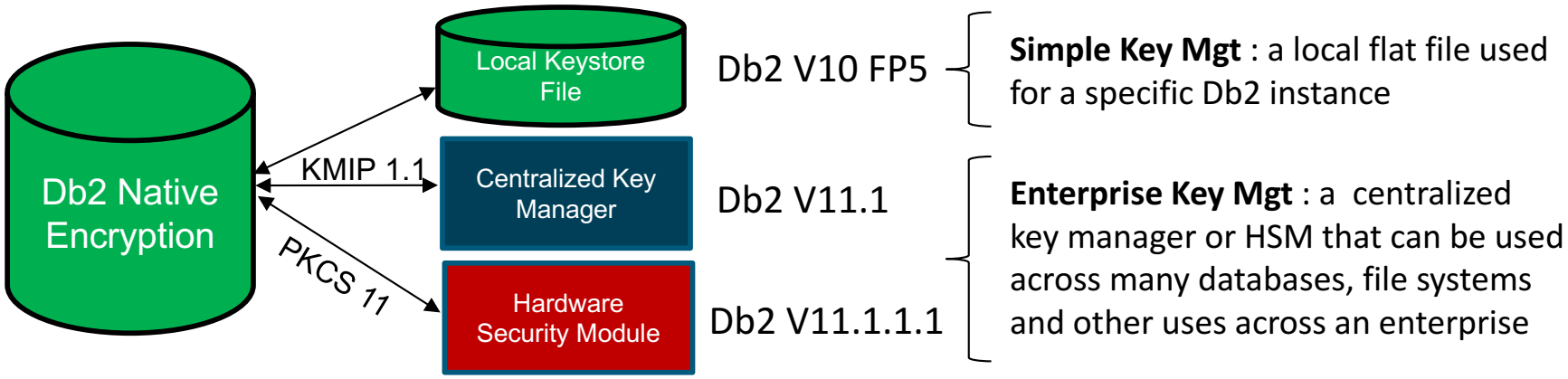
# pureScale Applications – Advanced Security

- **All advanced security is supported with pureScale**
  - Roles
  - Separation of duties
  - Label Based Access Control
  - Row and Column Access Control
  - Native Encryption
  - Trusted Context
  - Identity Assertion

# Encryption Included in Packaging

- **Db2 Native encryption for data-at-rest and Backups**
- **Industry compliant** ***(meets the requirements of NIST SP 800-131 compliant cryptographic algorithms and utilizes FIPS 140-2 certified cryptographic libraries)***

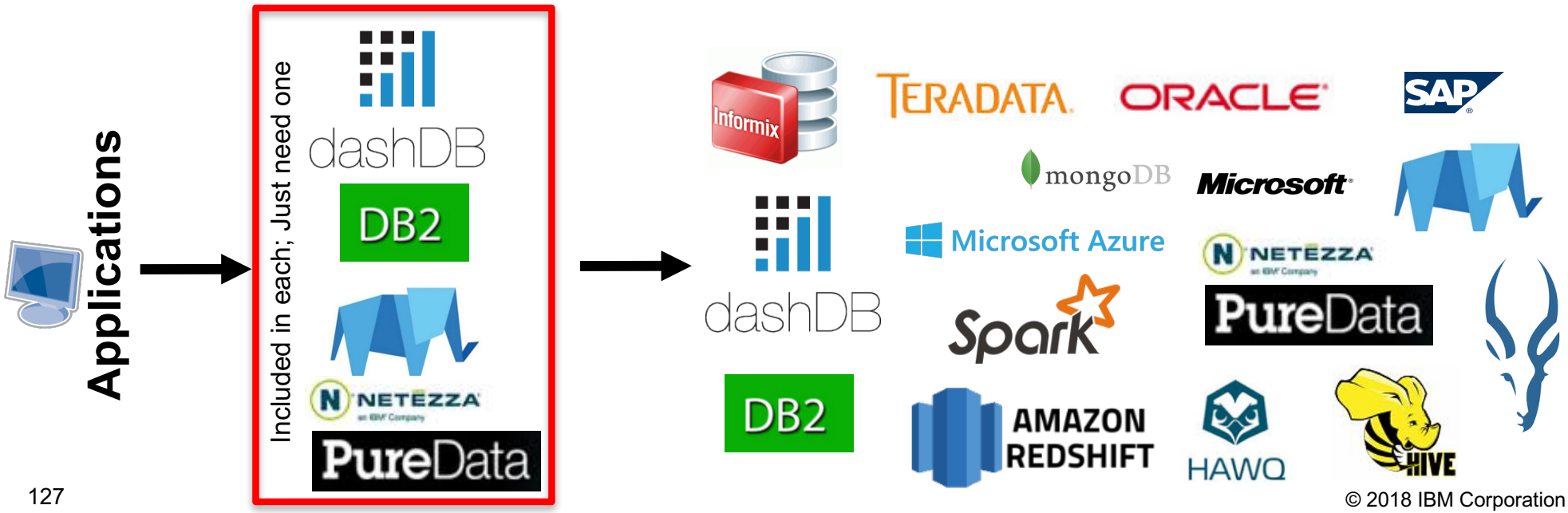
- **V11.1 adds support for KMIP 1.1 complaint centralized key managers**
  - Validated on IBM's Security Key Lifecycle Manager (ISKLM)




- **Direct support for PKCS11 Hardware Security Modules (HSMs)**
  - Validation to include SafeNet Luna & Thales nShield Connect+

# Federation Included in Advanced/Developer Packaging

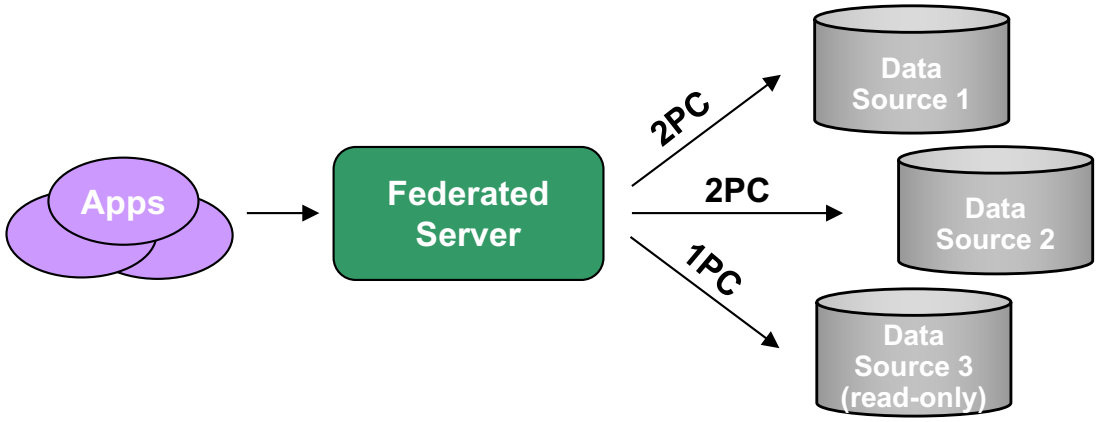
- **Integrated support for homogeneous federation**
  - Single install replacing any prior separate Infosphere Federation Server install
  - Support for upgrading from either a Db2 database product or Infosphere Federation Server
- **Additional Wrappers in Advanced Editions**
  - Db2, PureData System for Analytics (PDA), Oracle, Informix, dashDB, SQLServer, BigSQL, SparkSQL, Hive, Impala, and other Big Data sources.



# pureScale Applications - Federated Two Phase Commit

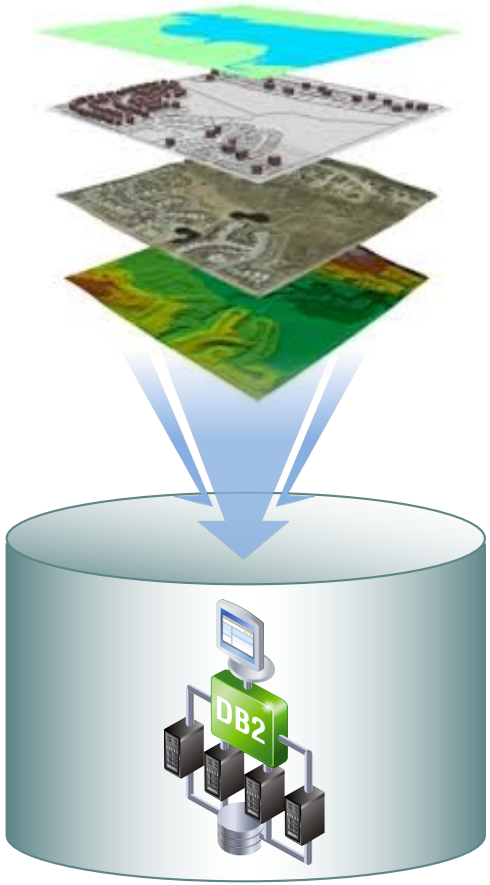
- **Federated two phase commit (F2PC) allows for insert/update/delete against multiple remote data sources within a single transaction**

- **F2PC supported with Db2 pureScale database server acting as a federated server**
  - Data source support for DB2 family (LUW, z/OS, i), Oracle, and Informix
  - Automatic recovery/resync of F2PC transactions during member/group recovery
  - Owning member of F2PC transaction can do resync even if home host not available
  - Ability to manually resolve from owner or non-owner member



# pureScale Applications - Db2 Spatial Extender

- You can store, manage, and analyze spatial data in a Db2 pureScale environment
- No difference between pureScale and non-pureScale in terms of setting up Spatial Extender and creating a project that uses spatial data



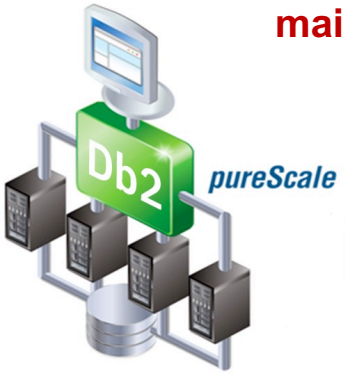
# V11.1 Summary of pureScale Enhancements

## New Support

- SYNC and Near-SYNC HADR modes
- Power Linux LE
- Text Search
- Failover Priority for member subsets
- GDPC multi-switch
- SLES 12, RHEL 6.8 on x86
- RESTORE REBUILD
- Online CREATE INDEX (allow write)
- Online ADD/DROP CF
- Multiple hosts in maintenance mode
- AIX and Linux OS level updates

## Improvements

- Install process and steps
- GPFS replication setup
- TRUNCATE TABLE
- Unified workload balancing
- Online management
- Member crash recovery (MCR)
- MCR improvements by default (no regvar)
- TCP/IP socket performance improvements
- Chrony auto-setup
- Single db2cluster commands to list hosts in maintenance mode and collecting perf data



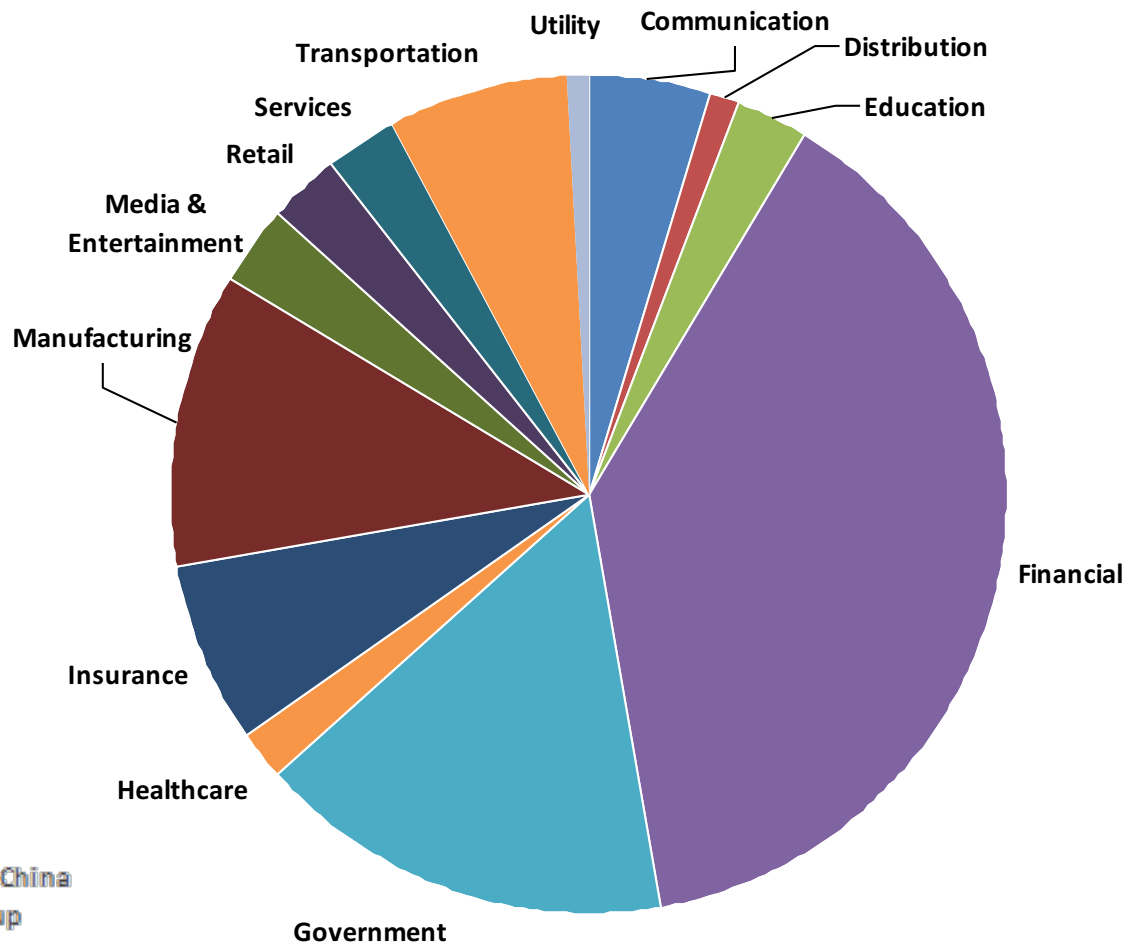



Db2 v11.1 GA | Db2 v11.1.1.1 | Db2 v11.1.2.2 | Db2 v11.1.3.3

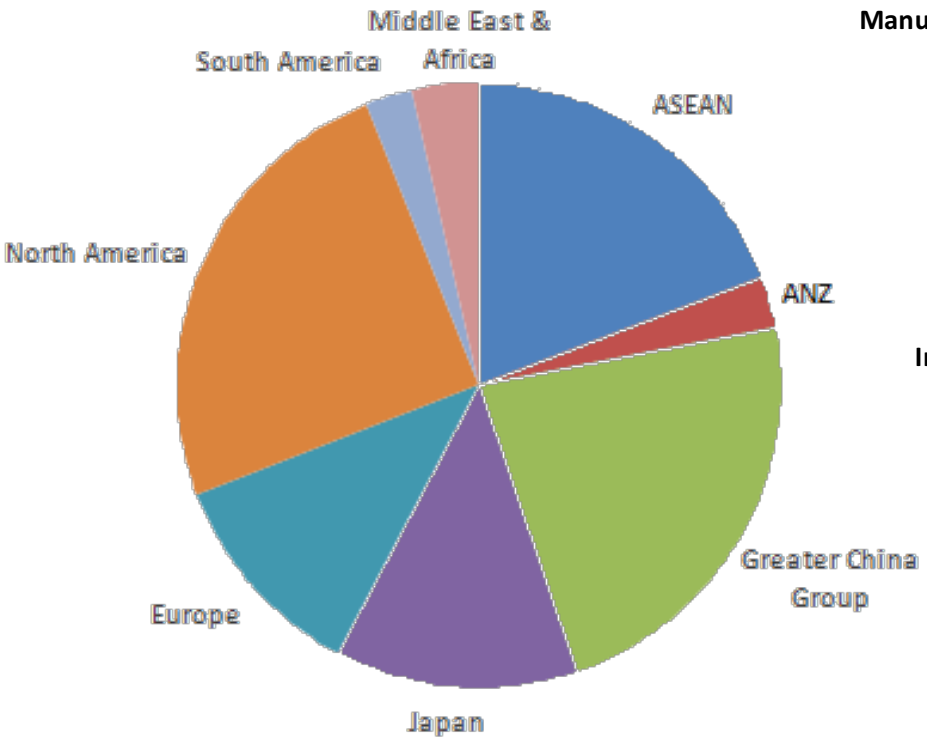
# Db2 pureScale – Successes in Every Industry and Every Geography

# Db2 pureScale Production Systems Worldwide*

## Industries




## Geographies




132 *not exhaustive as pureScale is included with Db2

# pureScale References – A selection of them ….










# pureScale Overall Value - Why clients choose pureScale

- **Business Drivers**
  - 24x7x365 Continuous Availability for Critical Business Applications
    - Need for 5-9s high availability with automatic recovery from failures
    - Maintenance (system and database environment) without an application outage
    - Management activities on-line ( on-line utilities )
  - Consolidation of infrastructure to provide shared services
    - One Db2 environment for multiple OLTP applications
  - Total Cost of Ownership
    - Cost per transaction
    - Application and Infrastructure investment protection
- **Technical Drivers**
  - Horizontal & Vertical near-linear scalability
    - Need to be able to grow and shrink capacity seamlessly to meet application workload
    - Capacity on demand, only paying for what you need when you need it
    - Application transparency through workload balancing
  - Ease of management
    - Integrated installation, configuration and management of all components

# Db2 pureScale – Achieving 24x7x365 Availability

- **Built into architecture**
  - High Availability
  - Workload Balancing
  - Application Transparency
- **Avoiding Planned Outages**
  - On-line OS and Hardware upgrades
  - On-line rolling Db2 fix pack updates
  - On-line add member
  - On-line utilities
- **Avoiding Unplanned Outages**
  - On-line recovery
  - Some disaster recovery capabilities with GDPC
  - Rich disaster recovery capabilities with HADR
- **Other Capabilities**
  - Data-at-Rest and Backup Encryption
  - Flexible server support and pricing models
  - Multi-tenancy capabilities
  - WLM capabilities