NonStop SQL/MX DBS

Concepts and Architecture

Frans Jongma, NonStop Advanced Technology Center

Agenda

- Definitions
  - Multi-tenant database
  - What defines a database
  - What defines an instance
- High level overview of SQL/MX DBS
  - Introducing new schemas
  - Changes in User management
- Quick overview of provisioning
- Detail: How storage and compute resources are allocated

What is SQL/MX DBS?

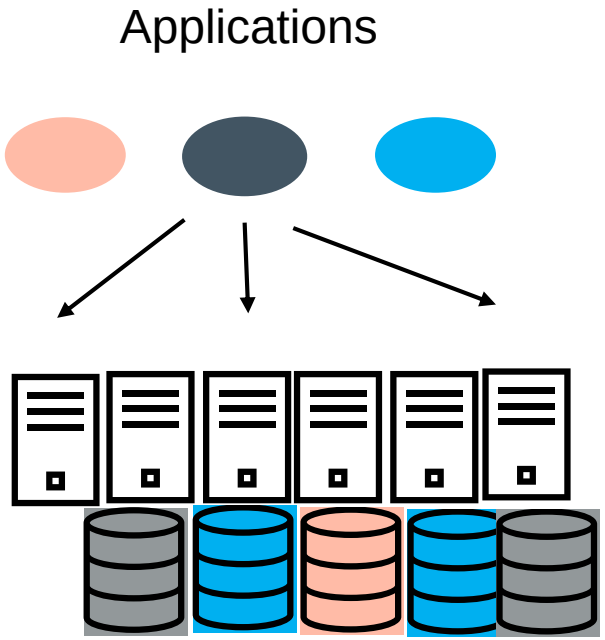
- A means to provide “Database as a Service” (DBaaS)
- SQL/MX DBS provides multi-tenant features with added user isolation

- Important Features
  - Simplifies provisioning of databases
  - Facilitates automation
  - Brings “cloud capabilities” to customers on-premise installations
    - Regular NonStop servers
    - Virtualized NonStop servers

- Requirements
  - L17.02 NonStop SQL/MX 3.5 and onward
    - L17.08 – SQL/MX 3.5.1
    - L18.02 – SQL/MX 3.6

Multi-tenant database

- Virtualized database servers lead to database “sprawl”
  - Add complexity and management efforts
- Multi-tenant databases allow sharing DBMS Software and system data between (isolated) users
- Examples
  - Oracle 12c Pluggable Databases in a Container Database instance
    - Shares SGA, undo, redo space amongst all tenants
  - Microsoft SQL Server Shared database, tenant’s schemas
    - Shares system database and temp database
  - NonStop SQL/MX DBS
    - Shares system software
    - Exclusive use of volumes (=lock space, cache buffers) to tenants
    - Catalog / Datasource represents a database



NonStop SQL scalable database

DBMS database and instance

Definition of terms

– Different products different names

– What does “Database” mean?

– What is an “Instance”?

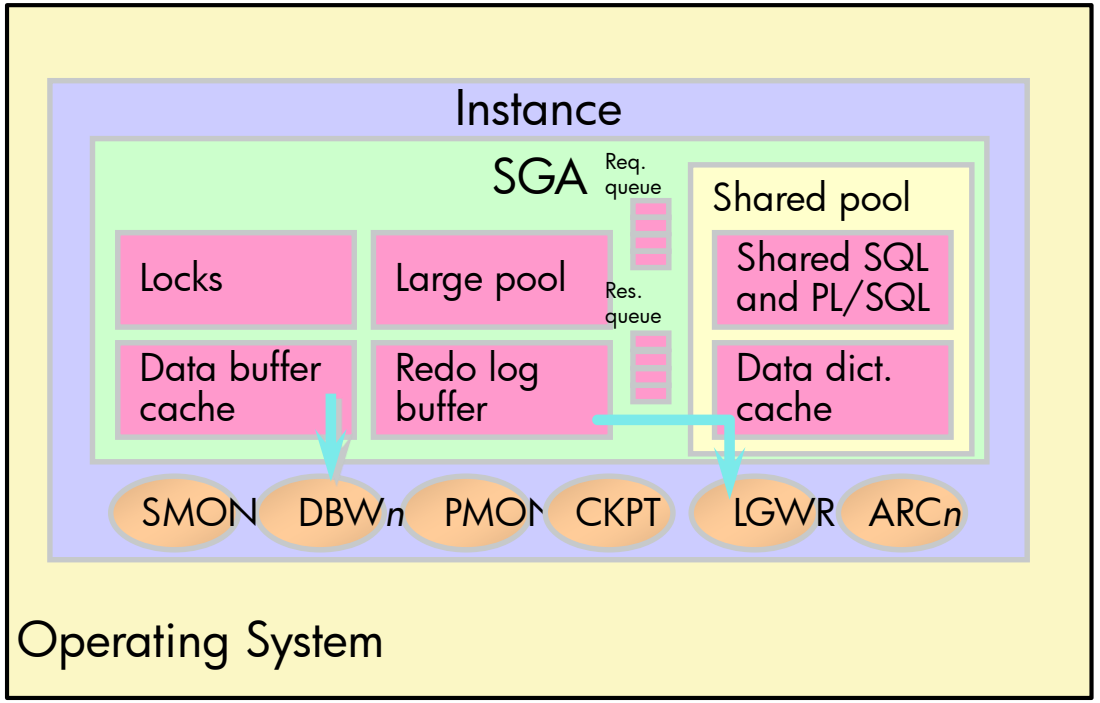
– What is a User?

– How different / similar are
  – Oracle
  – SQL Server
  – NonStop SQL ?

Oracle: Database and Instance

– **Oracle Database**
  – A collection of physical operating system files or disk
  – Database can be opened (mounted) by multiple instances (RAC)
  – Database contains multiple users = schemas
  – Log file per database at instance level

– **Oracle Instance**
  – The set of background processes or threads and a shared memory area that is shared across those running on a single computer
    – Maintains volatile stuff (locks, buffers …)
    – Can exist without storage; can be started / stopped
  – Instance can only mount one database



Oracle Create

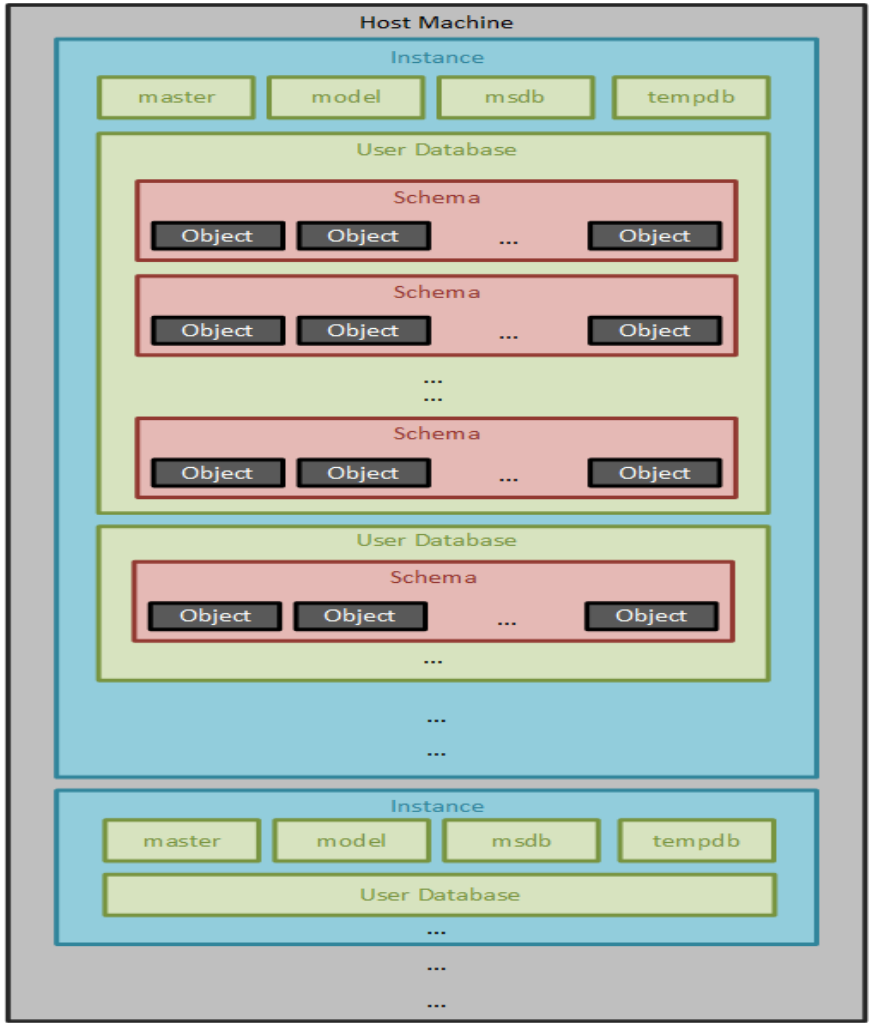
- DATABASE
  - Involves creating an instance, starting it and issue CREATE DATABASE command
    - Includes specification of SYS and SYSTEM users, logfiles, system and other tablespaces
- USER
  - User is an account through which you can log into the database; a way to get access
  - **`CREATE USER IDENTIFIED BY <password> <other attributes>`**
    - Add default table space and default temp tablespace for created objects by this user
    - Quotas for the user per tablespace
    - Grant session to <user> -- to enable the user to create a session to access the data
- SCHEMA
  - Create schema does not really create a schema. (“Schema” is created when you create a user)
  - Create schema is a way to create multiple objects with one statement in one transaction
    - **`CREATE SCHEMA <schema> CREATE TABLE …. CREATE VIEW ….`**
    - <schema> equals your Oracle Database user name

SQL Server: Database and Instance

– **SQL Server Database**
  – A container for database objects (tables, views)
  – The data file(s) that holds above objects
  – Multiple databases managed by one instance
  – Database contains multiple schemas
  – Database contains multiple users, but schema <> user
  – Log file per database (not per instance)

– **SQL Server Instance**
  – A copy of sqlserver.exe running as an OS service (SQL Server 2016)
  – Manages several system databases and one or more user databases

– SQL Server Schemas and users are not the same (as they are in Oracle)



Hewlett Packard Enterprise

intel

SQL Server Create

- DATABASE
  - Creates a new database within an instance, within the “master database”
  - Specify database name, data file(s) with attributes such as initial and max sizes
  - Specify the log file(s) with file attributes
- USER
  - “there are eleven types of users” (!)
  - Windows user or SQL User with password are two of these user types
  - Create user command can connect the user to a default schema, the schema that will **own** objects created by this user
- SCHEMA
  - There is a default, called dbo (= database owner). (e.g. database.dbo.table)
  - Like Oracle a create schema statement can include several other create statements
  - Schema and schema owner names are different
    - `CREATE SCHEMA Production AUTHORIZATION [Contoso\Mary];`

NonStop SQL: Database and Instance

- **NonStop SQL Database**
  - The Operating System data files that represent database objects (tables, views, indexes etc.)
  - NonStop SQL follows the ANSI model: CATALOG.SCHEMA.<object>
  - In NonStop SQL DBS, catalog maps to Database name defined by tenant when provisioned.

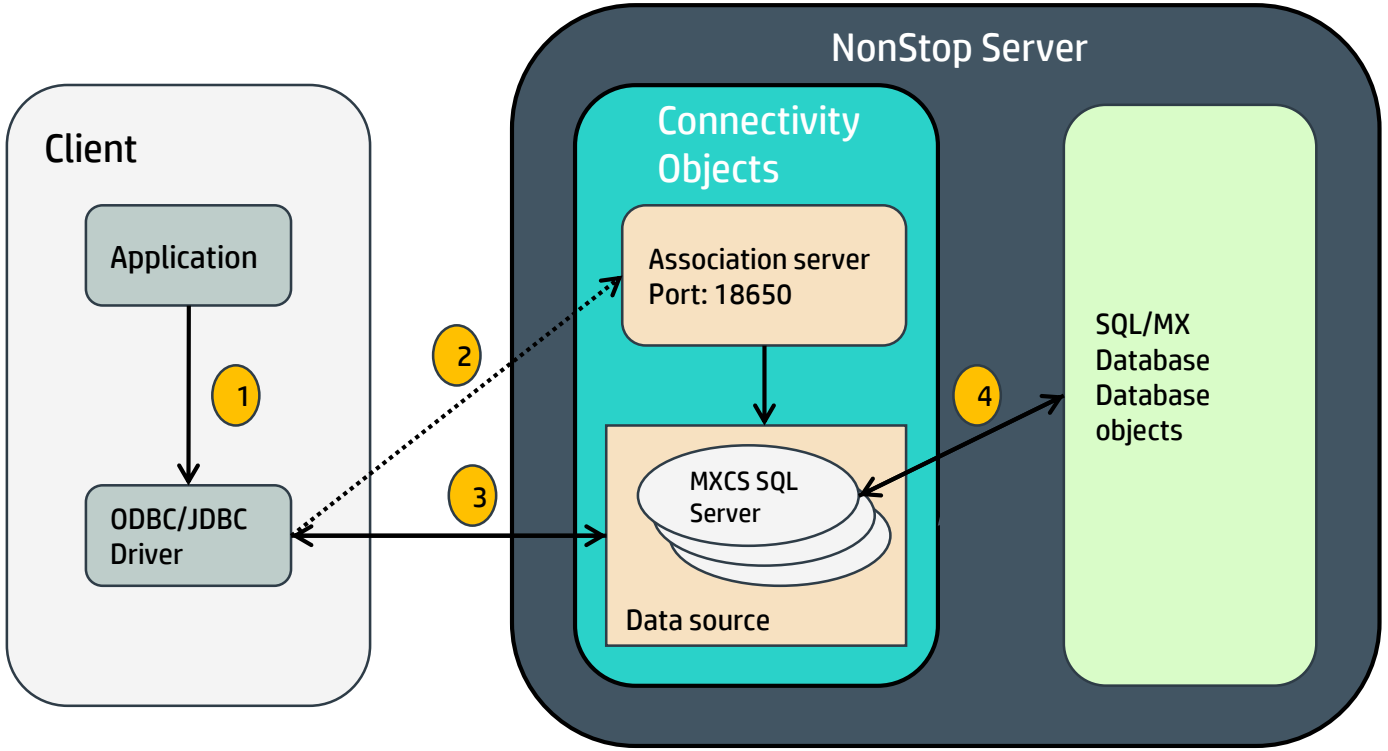
- **NonStop SQL Instance**
  - NonStop SQL is integral part of the NonStop OS
    - Find database engine components in libraries and Disk Access Managers (DAM)
  - If the system is up, the database is up
  - The OS equals “the instance”; all databases on a system are managed by the same version of the software
  - Database locks and cache are managed by the DAMs in a shared-nothing model
    - More processors allow more memory and processing capacity which leads to more volumes and more lock space and cache space
- **NonStop DBS “Instance”**
  - The data source name through which a tenant’s catalog and schema can be accessed
  - Data source name equals the catalog name and “is” the database. Schema names defined by DBA and provisioning portal
  - A data source can be stopped/started by a system administrator. This does not bring a NonStop database “down”

NonStop SQL Create

- DATABASE
  - **`CREATE CATALOG`** comparable to **`CREATE DATABASE`**
    - Catalog is a collection of System and User Schemas
    - In DBS, `CREATE CATALOG` is performed during provisioning process
    - `CREATE CATALOG` does not specify any storage parameters for user data (location for catalog metadata is optional)
- USER
  - Currently, users are created outside the database by the OS
  - In DBS, provisioning scripts attach provisioned users to NonStop user IDs
  - User access to tables done via GRANT/REVOKE
- SCHEMA
  - `CREATE SCHEMA <schema name> [ schema element , schema element , …. ]`
    - Schema element is create table or index or view etc. Similar to Oracle and SQLServer
  - Optional Location clause (ZSDxxxxx subvolume name)

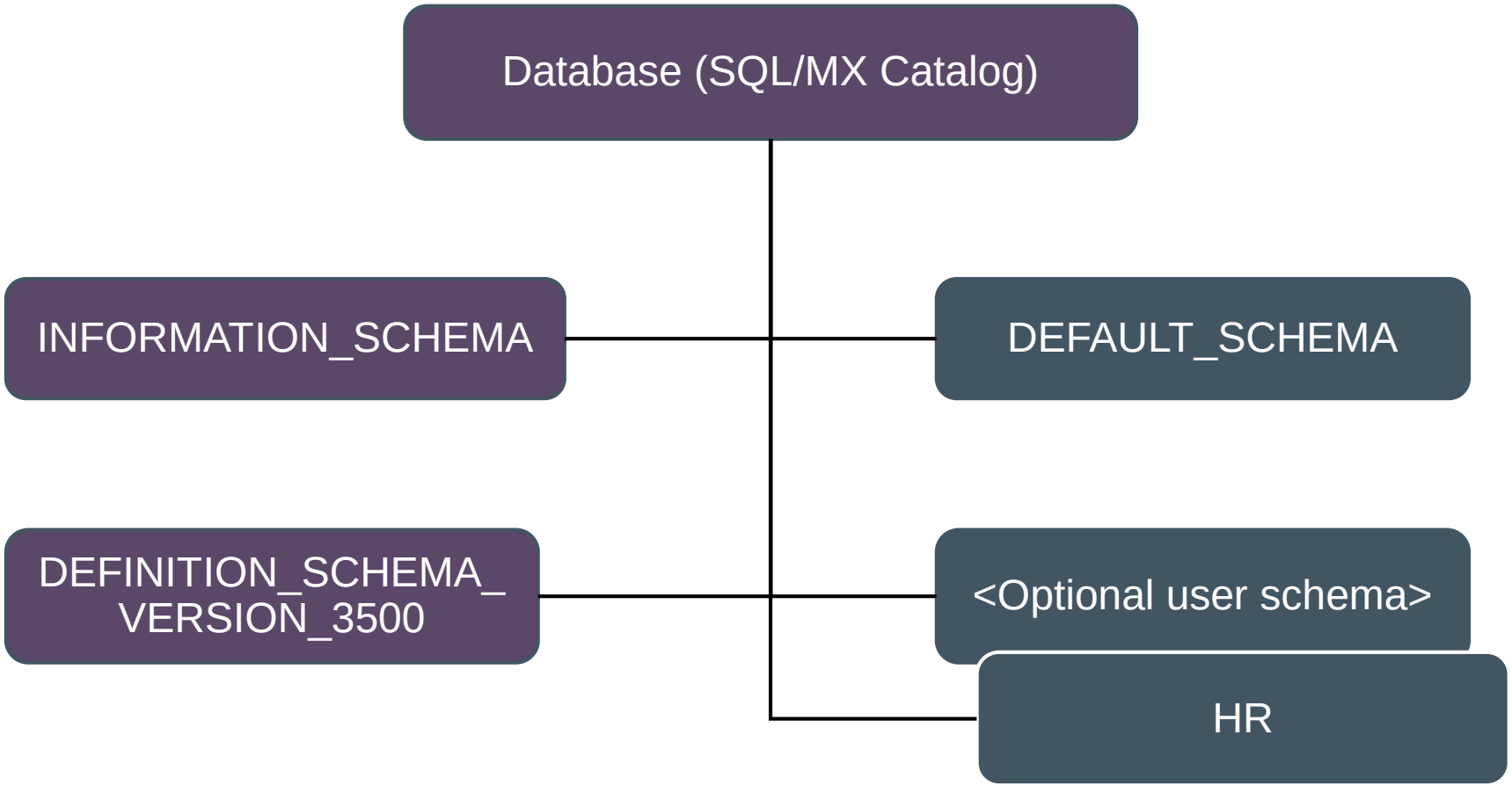
Data Source

- **In our context: Database Services**
  - The Data Source is the port of access to the data
  - Referred to as Database Name (ODBC) or serverDataSource (JDBC)
  - Usually contains address and port of the database listener
  - Often listeners use a well-known port
    - SQL/MX : 18650
    - SQL/MX DBS : 2100
    - Oracle : 1521
    - SQL Server : 1433
    - MySQL : 3306
    - …



NonStop SQL/MX DBS high-level overview

Database schemas in SQL/MX DBS



Two important schemas in a database

INFORMATION_SCHEMA

- Information about the database
  - Datasource
  - Schemas
  - Storage
  - CPUs
  - Users
  - Privilege groups

DEFINITION_SCHEMA_VERSION_*nnnn*

- SQL/MX standard metadata
- Per schema information about
  - Tables
  - Partitions
  - Access paths
  - Constraints
  - Indexes
  - Privileges
  - Partitions
  - Etc.

User management

– Multi-tenant support requires additional user management functionality
  – Allow user names that were defined elsewhere
  – Deny access other other user's metadata
  – Allow an end-user to add other users to access a database

Additions to SYSTEM_SECURITY_SCHEMA

- Allow “external users” access via MXCS
  - Also known as database users
- An email address (Joe@hpe.com)
- Windows user name (ASIAPAC\Senthil)

- Privilege groups are used to assign privileges to multiple users (even to future members of the group)
  - In DBS: all users of a database belong to a group
  - Group is created when a database is provisioned
- Introducing SCHEMA privileges
  - Simplifies management at schema level using privilege groups
  - DDL (manage objects)
  - DML (manage data)

- **DATABASE_USERS**
- **DATABASE_USERS_EXT**
- **PRIVILEGE_GROUPS**
- **PRIVILEGE_GROUP_GRANTS**
- **PRIVILEGE_GROUP_MEMBERSHIP**

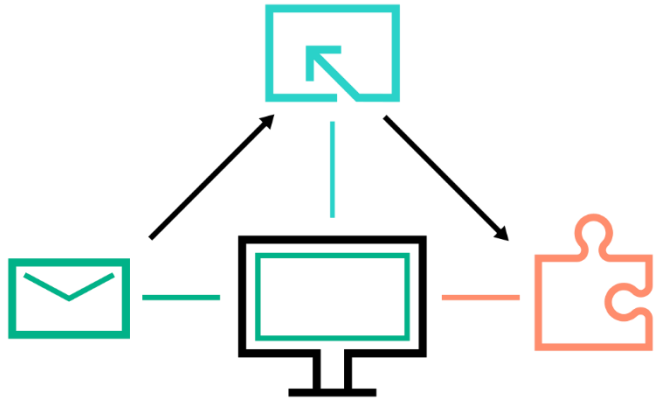
Quickly provision a database

SQL/MX 3.5: Database Services

DBS thin provisioning interface

mxdbs CLI
Create a database Share a database Delete a database Add more storage Add additional users Change user's access level Change user's password Delete a user Show databases

The API should be kept simple, database agnostic
The goal is to cover the life cycle management scope from long term projects to ad-hoc projects.

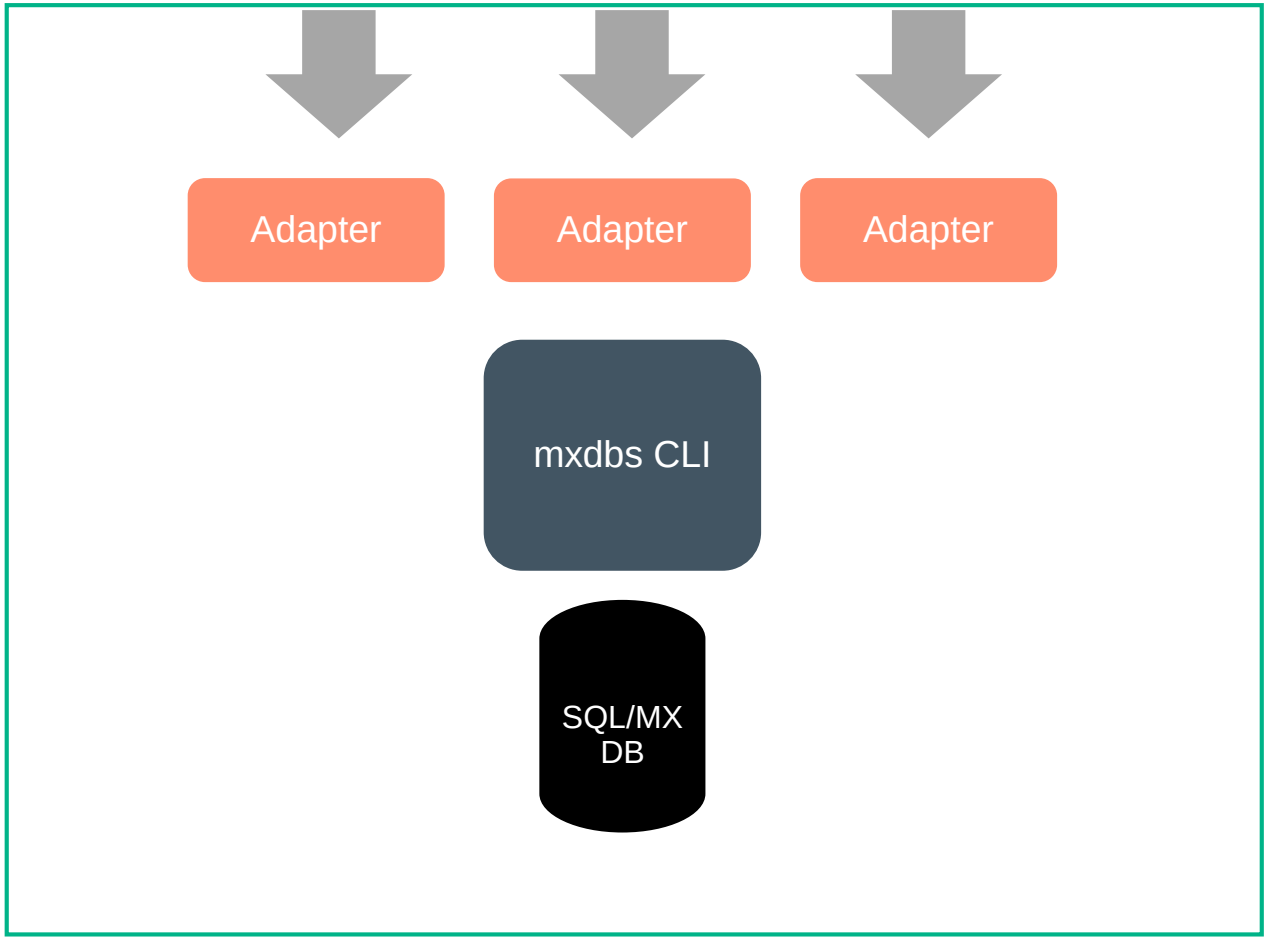


```
usage: mxdbs [-h] [-V]
             { db-create | db-alter-share | db-delete | db-add-user |
             db-remove-user | db-add-storage | db-user-change-access |
             user-change-password | show-databases } ...
```

Hewlett Packard Enterprise intel

One API to support multiple clients and protocols

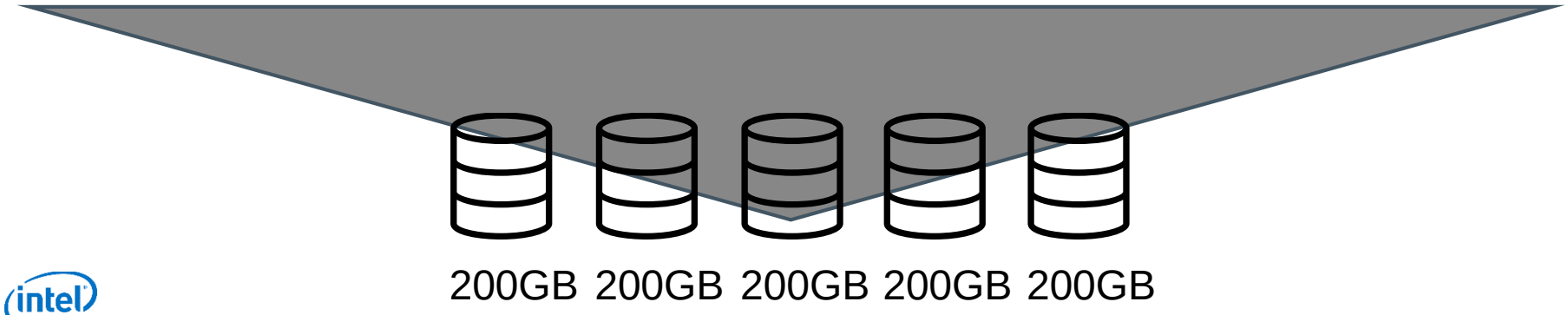
– Protocol examples
  – SSH
  – HTTPS
  – Message bus (AMQP)
  – Local execute/launch
– Client examples
  – HPE Operations Orchestration
  – Ansible
  – Client issuing REST API calls
  – Openstack Trove



Functions of “create-db”

– Based on requested database size, (free) storage is assigned
  – Use physical drives that are partitioned
– Use dedicated volumes for database
– Security ACLs assigned
– Catalog and initial schema created
– Datasource created and started
– Database is now ready to roll
  – All you need is the appropriate driver

```
> mxdbs db-create dbs_fj 1000 emea_fjongma Welcome-1234
```



Use it

```
>rmxci -h 172.17.197.173:2100 -dsn DBS_FJ -u
emea_fjongma -p Welcome-1234
Welcome to the NonStop(TM) SQL/MX Remote
Conversational Interface
(c) Copyright 2015-2016 Hewlett Packard Enterprise
Development Company, LP

Connected to Data Source: DBS_FJ

SQL>CREATE TABLE departments
+>     ( department_id    NUMBER(4) NOT NULL
+>                                  PRIMARY KEY
+>     , department_name  VARCHAR2(30)
+>       CONSTRAINT  dept_name_nn  NOT NULL
+>     , manager_id       NUMBER(6)
+>     , location_id      NUMBER(4)
+>     ) ;

--- SQL operation complete.
```

```
SQL>showddl departments;

CREATE TABLE DBS_FJ.DEFAULT_SCHEMA.DEPARTMENTS
  (
    DEPARTMENT_ID                    NUMERIC(4, 0) NO DEFAULT
      -- NOT NULL NOT DROPPABLE
  , DEPARTMENT_NAME                  VARCHAR2(30) CHARACTER
SET ISO88591
      COLLATE DEFAULT NO DEFAULT -- NOT NULL NOT DROPPABLE
  , MANAGER_ID                       NUMERIC(6, 0) DEFAULT
NULL
  , LOCATION_ID                      NUMERIC(4, 0) DEFAULT
NULL
  , CONSTRAINT
DBS_FJ.DEFAULT_SCHEMA.DEPARTMENTS_486497159_5192 PRIMARY KEY
      (DEPARTMENT_ID ASC) NOT DROPPABLE
  , CONSTRAINT
DBS_FJ.DEFAULT_SCHEMA.DEPARTMENTS_776297159_5192 CHECK
      (DBS_FJ.DEFAULT_SCHEMA.DEPARTMENTS.DEPARTMENT_ID IS NOT
NULL AND
      DBS_FJ.DEFAULT_SCHEMA.DEPARTMENTS.DEPARTMENT_NAME IS NOT
NULL) NOT
      DROPPABLE
  )
  LOCATION \NSX09.$HD0300.ZSDV34TJ.FTDKSC00
  NAME NSX09_HD0300_ZSDV34TJ_FTDKSC00
  ATTRIBUTES BLOCKSIZE 4096
  STORE BY (DEPARTMENT_ID ASC)
  ;

--- SQL operation complete.
```

Hewlett Packard Enterprise

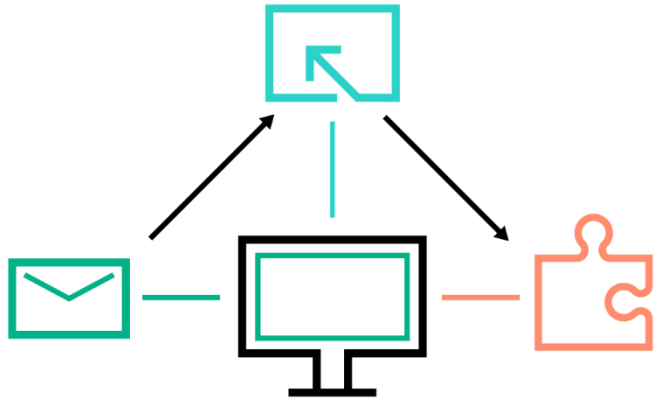
Details Storage allocation

SQL/MX 3.5: Database Services

DBS thin provisioning interface

mxdbs CLI
Create a database Share a database Delete a database Add more storage Add additional users Change user's access level Change user's password Delete a user Show databases

The API should be kept simple, database agnostic
The goal is to cover the life cycle management scope from long term projects to ad-hoc projects.



```
usage: mxdbs [-h] [-V]
             { db-create | db-alter-share | db-delete | db-add-user |
             db-remove-user | db-add-storage | db-user-change-access |
             user-change-password | show-databases } ...
```

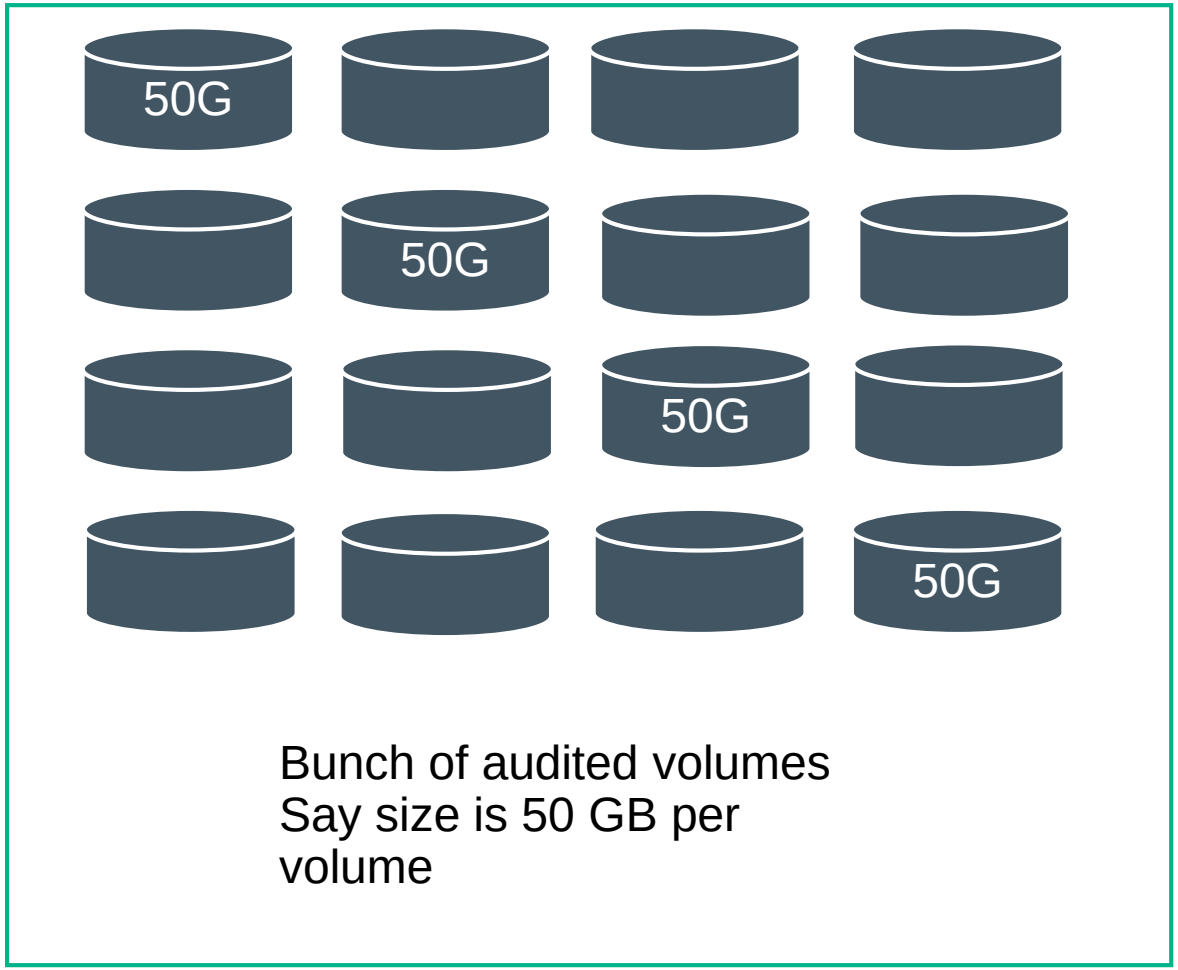
Hewlett Packard Enterprise intel

Starting point

– OS: L17.02 or higher

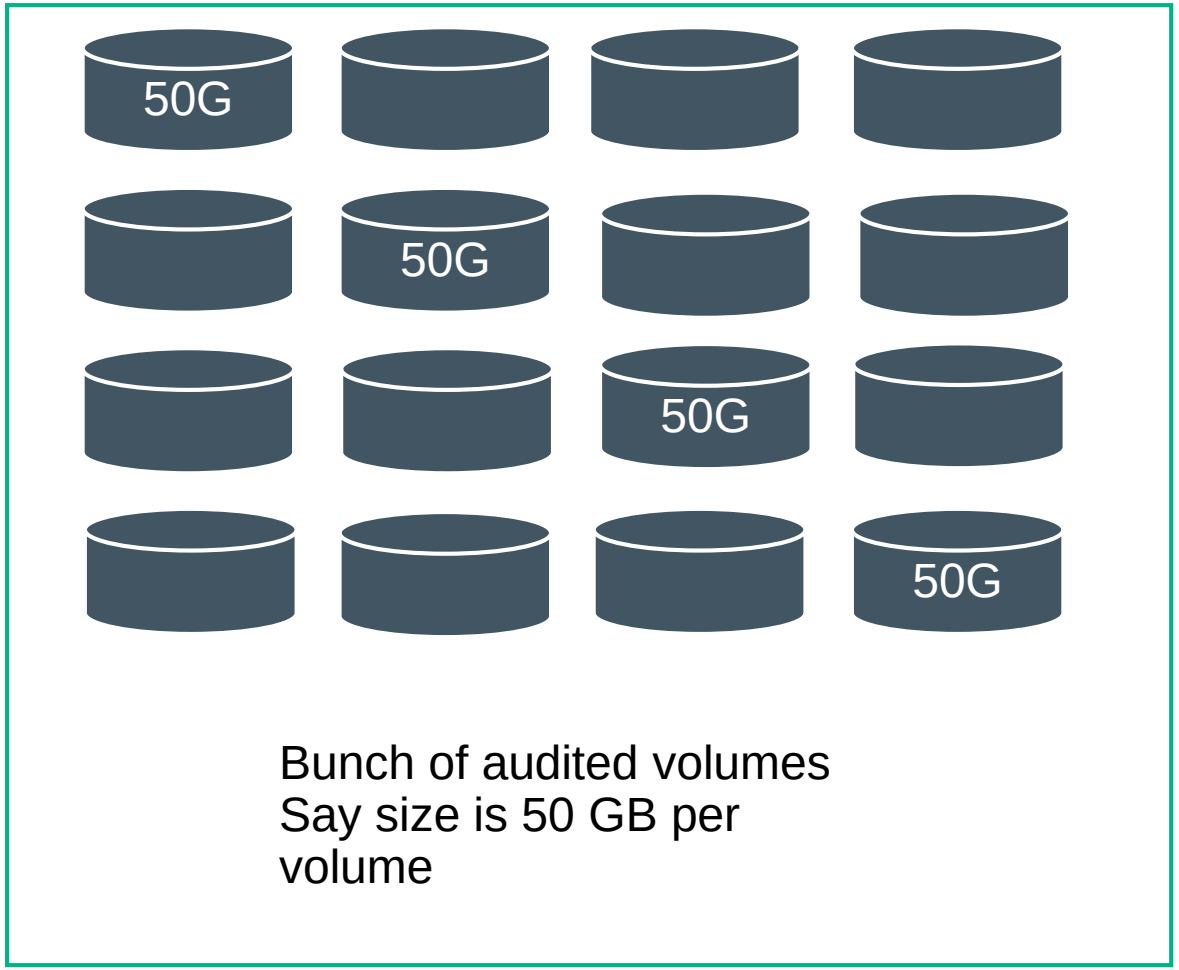
– SQL/MX 3.5 with DBS initialized
  – Defines the available volumes
  – Assigns ranges of Guardian users and Safeguard groups
  – Defines port numbers for MXCS DBS data sources

– Bunch of Storage for DBS



User requests a database (1)

– User request comes in
  – User: frans@hpe.com
  – Size request: 10GB
  – Database name: DB_FRANSJ
  – User password: Welcome



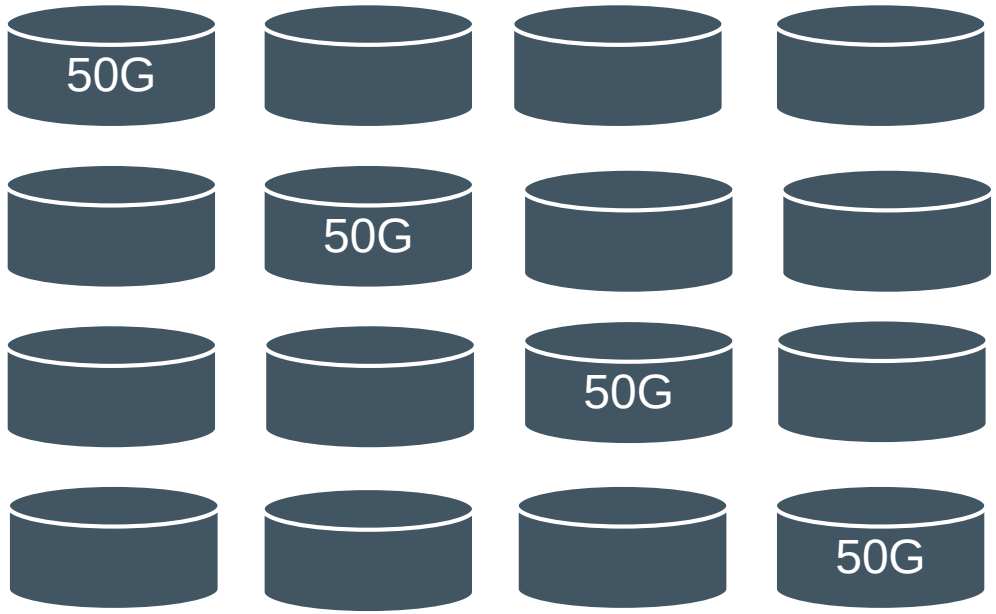
User requests a database (2)

– Command required:
  – On platform by TENANT.ADMIN

```
> mxdbs db-create db_fransj 10 frans@hpe.com welcome
```

  – Off platform by e.g ssh command
  – Could even be a web-site invoking ssh

```
> ssh mx-nsx09 /usr/tandem/sqlmx/bin/mxdbs db-create
db_fransj 10 frans@hpe.com welcome
```

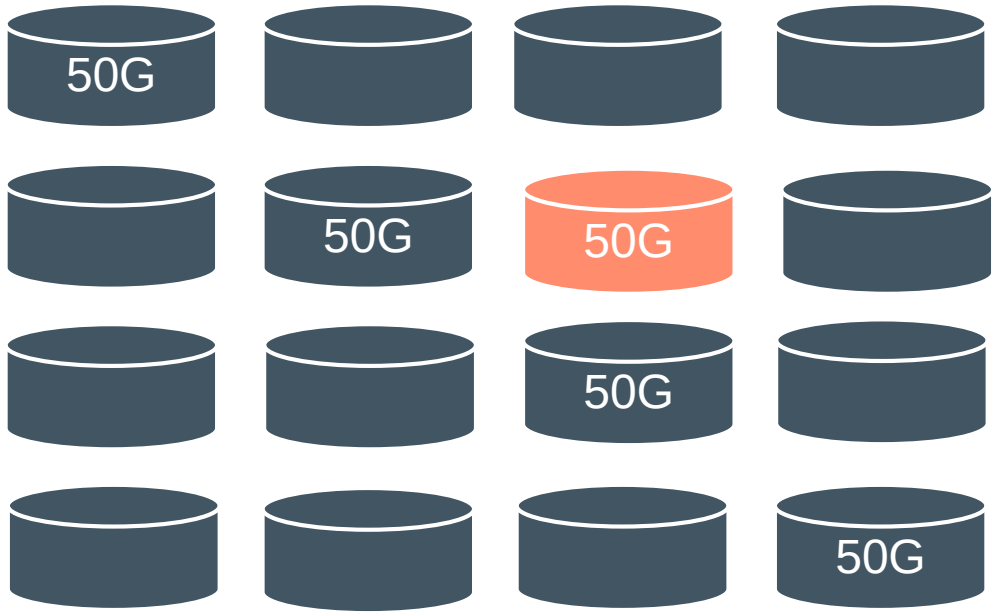


Bunch of audited volumes
Say size is 50 GB per volume

User requests a database (3)

– Off platform by e.g ssh command

```
~> ssh mx-nsx09 /usr/tandem/sqlmx/bin/mxdbs db-
create db_fransj 10 frans@hpe.com welcome
Hewlett Packard Enterprise NonStop(TM) SQL/MX DBS
Client 3.5
(c) Copyright 2016 Hewlett Packard Enterprise
Development LP.

db-create command started.

MXCS Service Host    : 172.17.197.173
MXCS Service Port    : 2100
Datasource Name      : DB_FRANSJ
Initial Schema Name  : "DEFAULT_SCHEMA"
OSS Directory        : DB1001

---  mxdbs operation complete.
```

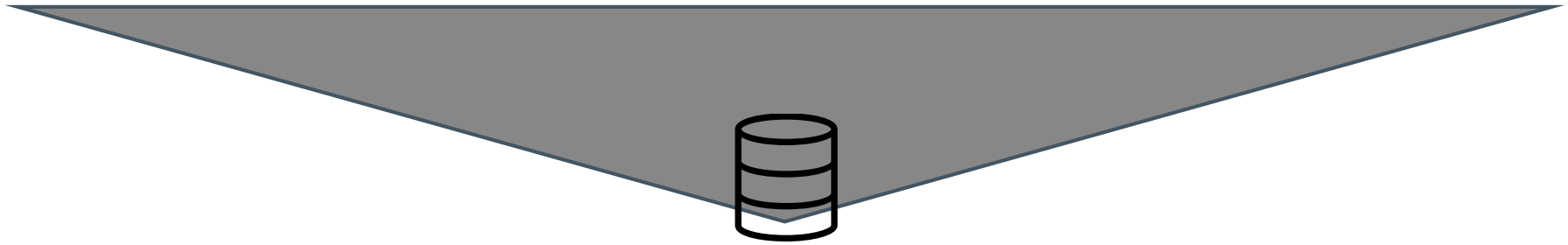


One volume (50GB)
assigned to DB_FRANSJ

Functions of “create-db”

– Based on requested database size, (free) storage is assigned
  – Use physical drives that are partitioned
– Use dedicated volumes for database
– Security ACLs assigned
– Catalog and initial schema created
– Datasource created and started
– Database is now ready to roll
  – All you need is the appropriate driver

```
> mxdbs db-create db_fransj 10 frans@hpe.com welcome
```



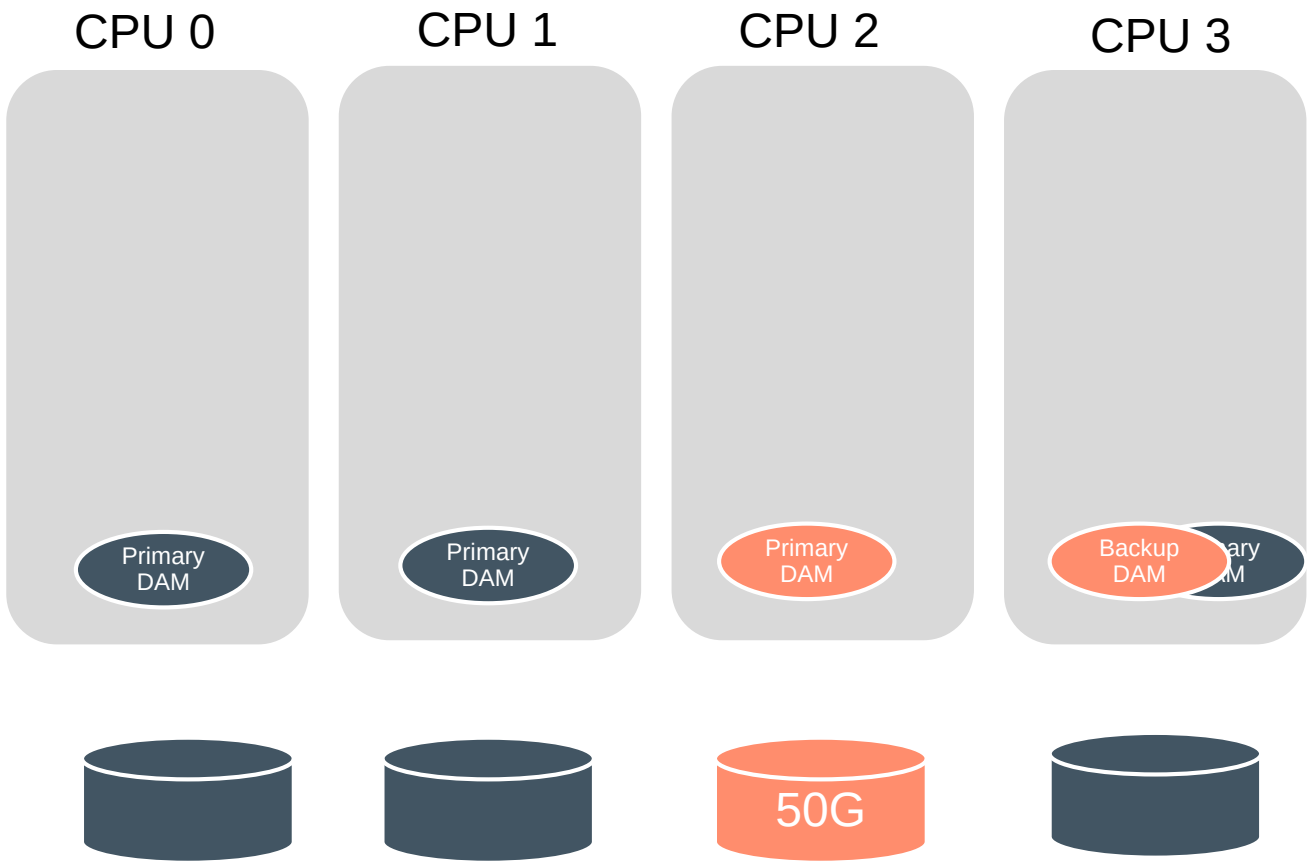
50GB

Details Compute assignment

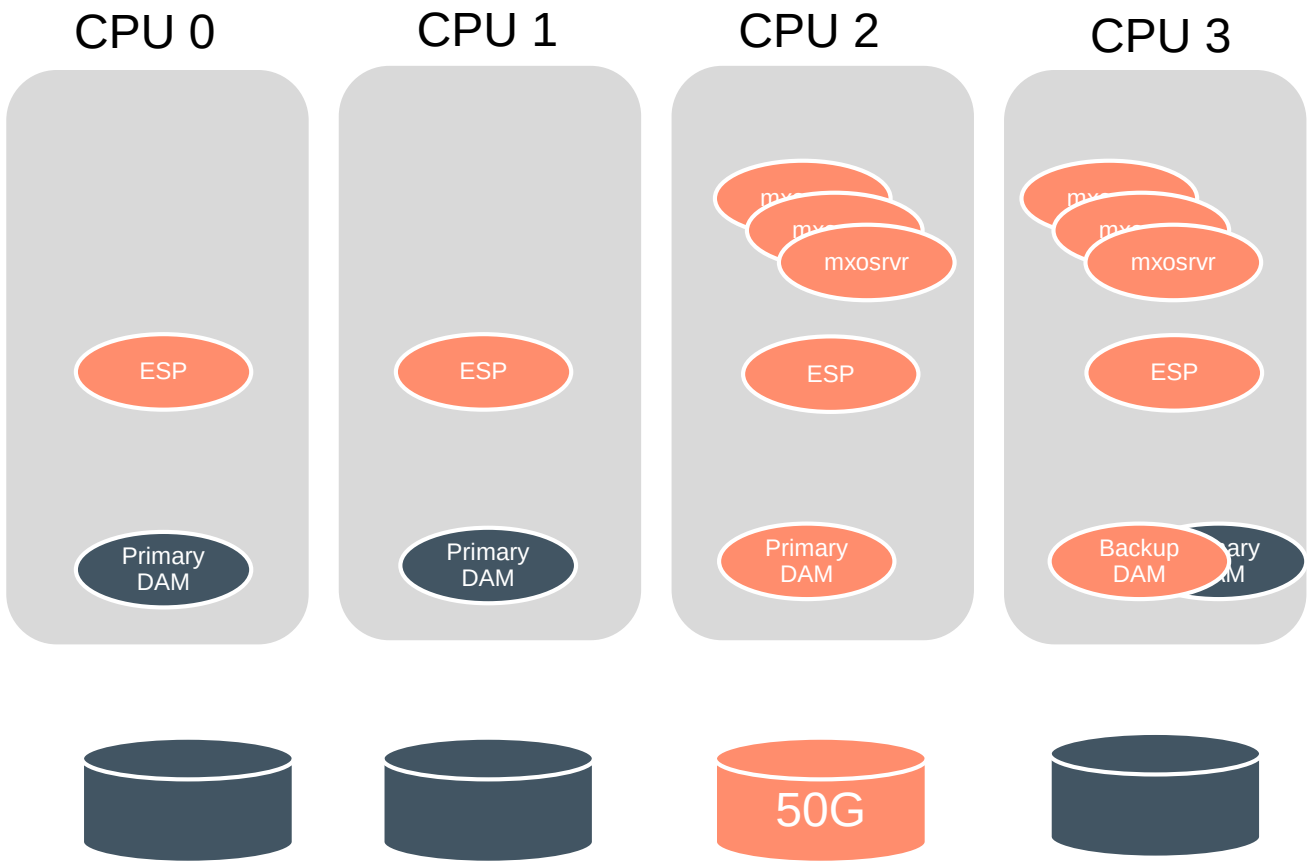
SQL/MX DBS Compute resources

- Disk process or Data Access Manager
  - Primary and backup process
  - Exclusive use for tenant
  - Cache memory
  - Lock space
  - SQL area
- MXOSRVR process
  - One for every connection
  - Compiler process
- Executor Server Process (ESP)
  - Used for sequence generators
  - Used for certain types of parallel queries
- DAMs are assigned based on amount of storage space requested
- DAMs for a tenant are distributed across the system
  - To provide as much compute power as possible
  - Fault-tolerance is standard: ever DAM has a backup processor
  - The processors that are used by DAMs are also assigned to connections to the database.
    - Minimal two processors
- ESPs can run in any processor of the system

Process overview at-a-glance (1)

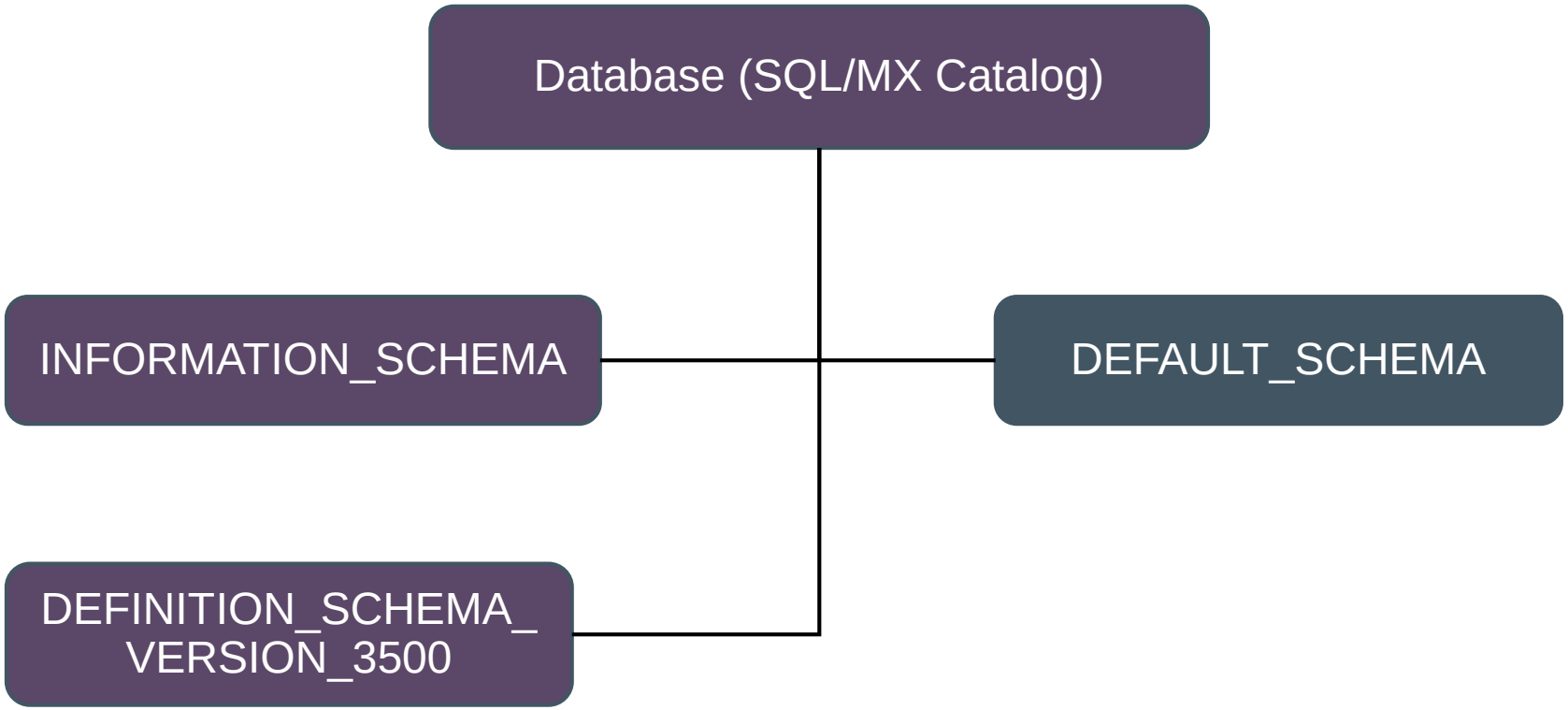


Process overview at-a-glance (2)



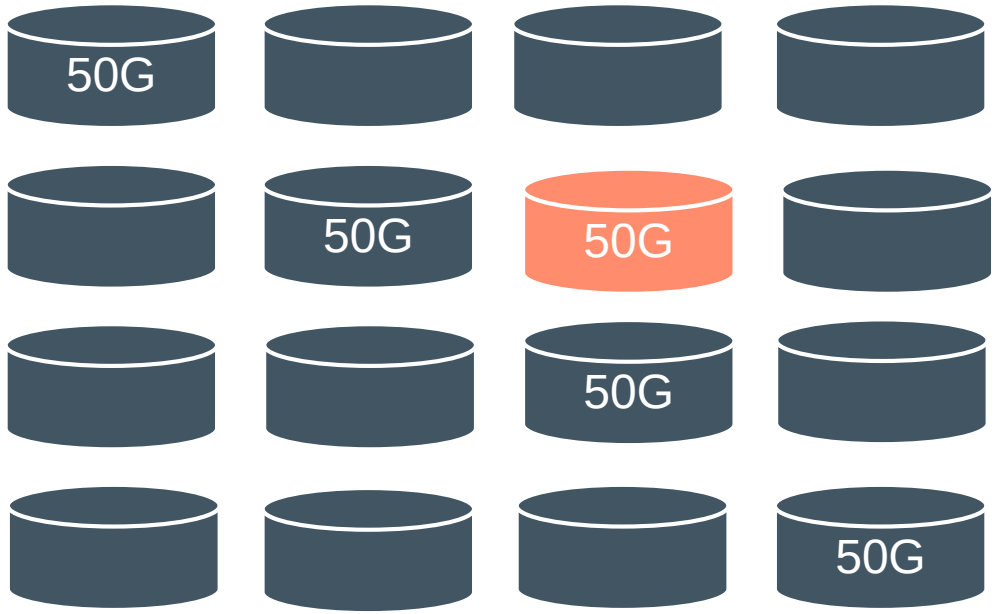
Using the database

Initial schemas in SQL/MX DBS



User uses database (1)

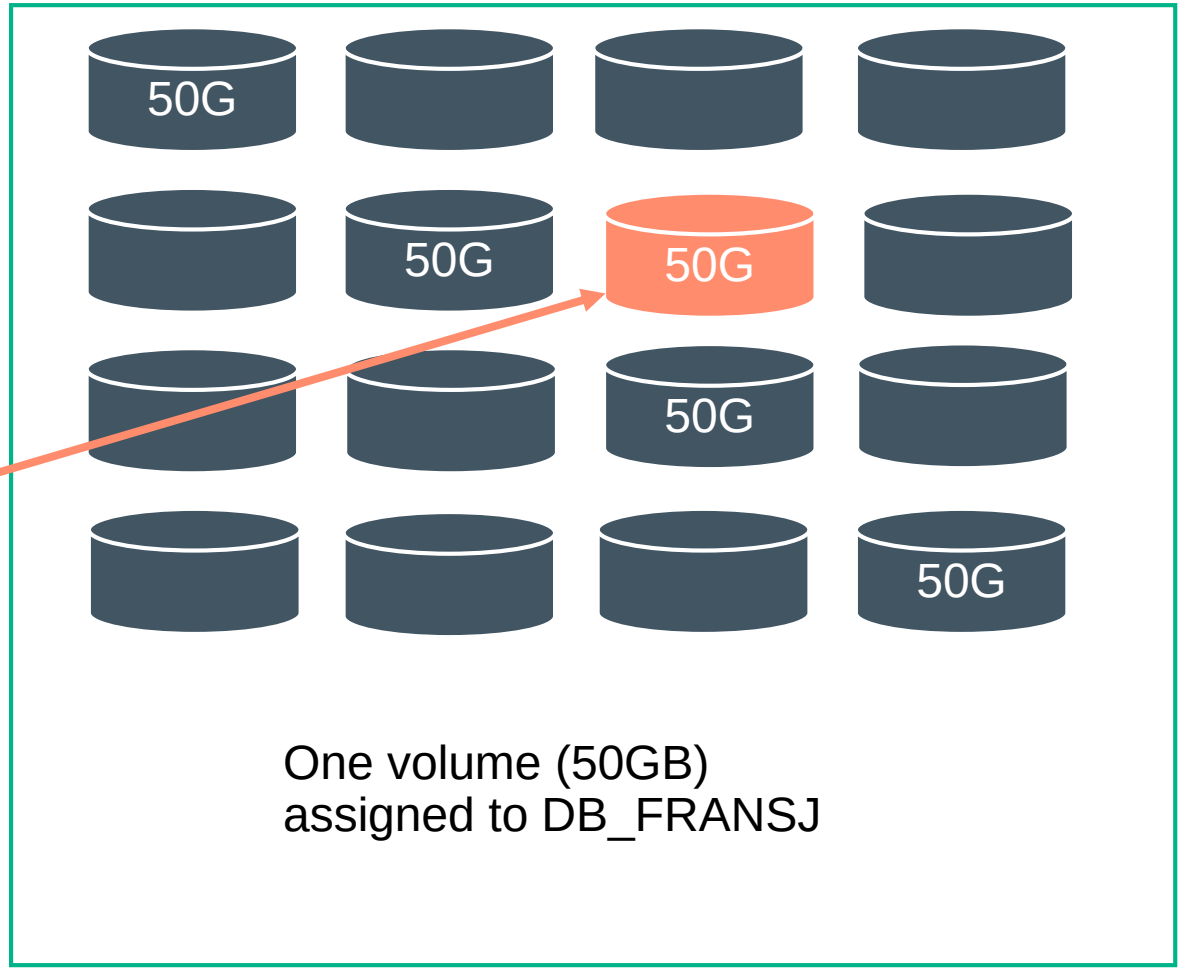
```
~> rmxci -h nsk-nsx09:2100 -dsn DB_FRANSJ -u
frans@hpe.com -p welcome
Welcome to the NonStop(TM) SQL/MX Remote
Conversational Interface
(c) Copyright 2015-2016 Hewlett Packard Enterprise
Development Company, LP


Connected to Data Source: DB_FRANSJ

SQL>set sqlprompt "%catalog.%schema %server SQL>";

DB_FRANSJ."DEFAULT_SCHEMA" nsk-nsx09:2100
SQL>create table T (a int not null primary key);

--- SQL operation complete.

DB_FRANSJ."DEFAULT_SCHEMA" nsk-nsx09:2100
SQL>showddl t;
```



One volume (50GB)
assigned to DB_FRANSJ

User uses database (2)

```
DB_FRANSJ."DEFAULT_SCHEMA" nsk-nsx09:2100
SQL>showddl t;

CREATE TABLE DB_FRANSJ.DEFAULT_SCHEMA.T
  (
    A                                     INT NO
DEFAULT -- NOT NULL NOT DROPPABLE
  , CONSTRAINT
DB_FRANSJ.DEFAULT_SCHEMA.T_222988845_8526 PRIMARY
KEY (A ASC)
      NOT DROPPABLE
  , CONSTRAINT
DB_FRANSJ.DEFAULT_SCHEMA.T_132688845_8526 CHECK
      (DB_FRANSJ.DEFAULT_SCHEMA.T.A IS NOT NULL)
NOT DROPPABLE
  )
  LOCATION \NSX09.$HD0002.ZSDF9Q63.JMKS8H00
  NAME NSX09_HD0002_ZSDF9Q63_JMKS8H00
  ATTRIBUTES BLOCKSIZE 4096
  STORE BY (A ASC)
  ;

--- SQL operation complete.
```

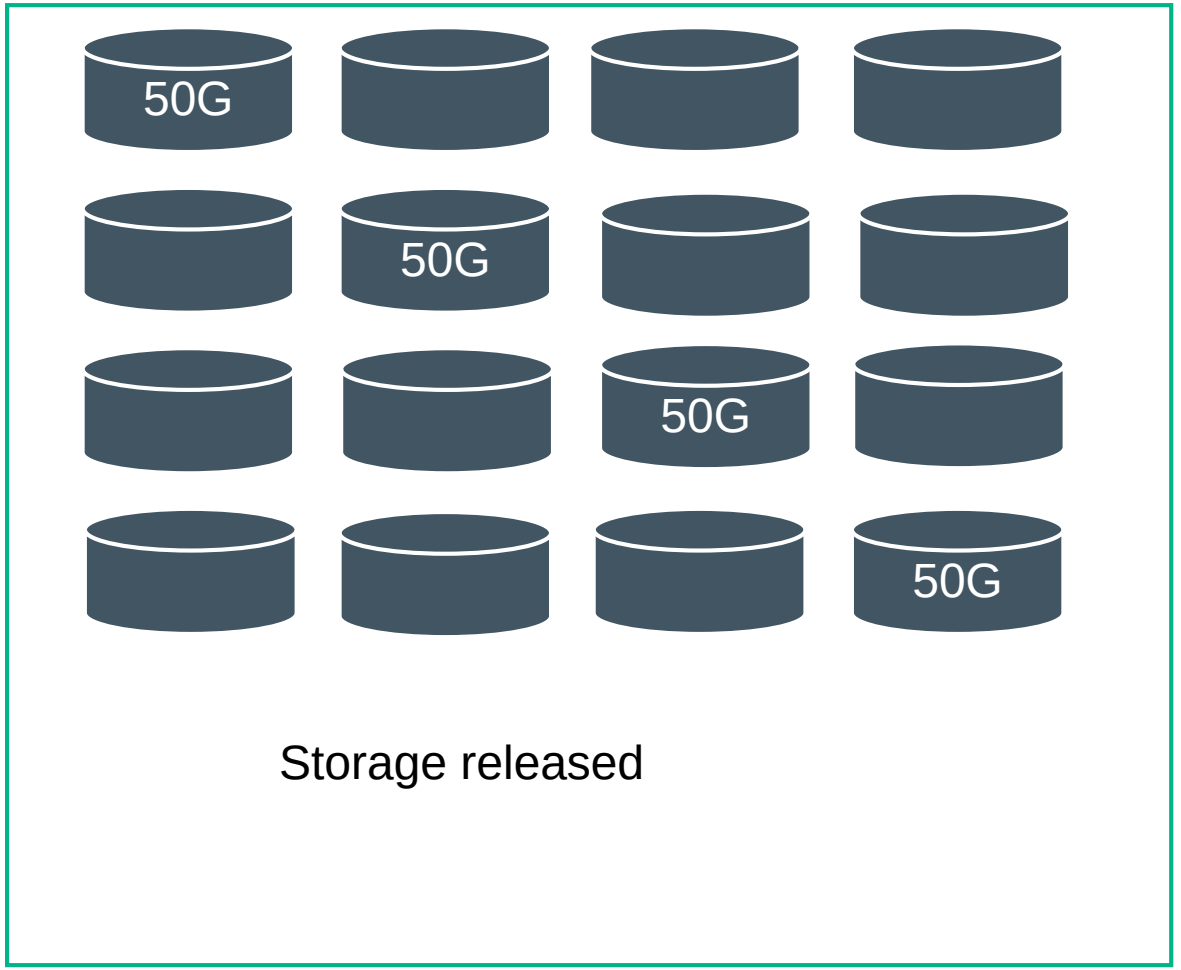


Hewlett Packard Enterprise

intel

User finished, deprovision

```
~> ssh mx-nsx09 /usr/tandem/sqlmx/bin/mxdbs show-
databases
Hewlett Packard Enterprise NonStop(TM) SQL/MX DBS
Client 3.5
(c) Copyright 2016 Hewlett Packard Enterprise
Development LP.

show-databases command started.

DATABASE_NAME DATABASE_UID       OSS_DIR IS_SHARED
------------- ------------------ ------- ---------
DBS_FJ        4352871103940041 56 DB1000  N
DB_FRANSJ     4352880674411114 94 DB1001  N

---  mxdbs operation complete.
~> ssh mx-nsx09 /usr/tandem/sqlmx/bin/mxdbs db-
delete db_fransj
Hewlett Packard Enterprise NonStop(TM) SQL/MX DBS
Client 3.5
(c) Copyright 2016 Hewlett Packard Enterprise
Development LP.

db-delete command started.
---  mxdbs operation complete.
```



EMS messages

```
2017-04-25 05:07:21 \NSX09.$ZAS02 TANDEM.ODBCMX.G06 021008 MXCS data source DB_FRANSJ is started. Event
Type: 4 Component Name: ODBC/MX Service Object Reference: TCP:$ZTC0/2100:NonStopODBC
```

```
2017-04-25 05:24:32 \NSX09.$ZAS02 TANDEM.ODBCMX.G06 021023 MXCS data source DB_FRANSJ is stopping abruptly
for DBS deprovision request.
Event Type: 4 Component Name: ODBC/MX Service Object Reference: TCP:$ZTC0/2100:NonStopODBC
2017-04-25 05:24:32 \NSX09.$ZAS02 TANDEM.ODBCMX.G06 021021 MXCS data source DB_FRANSJ stopped abruptly
 for DBS deprovision request.
Event Type: 4 Component Name: ODBC/MX Service Object Reference: TCP:$ZTC0/2100:NonStopODBC
```

Summary

– Useful for off-platform clients

– Tenants isolated from each other via volume assignments

– Clients cannot use volumes outside their assignments

– Datasource automatically created and removed after deprovision

– External user-IDs cannot be used to access the system using sh or TACL

– TDM_Default_DataSource is not activated. Users must use their assigned data sources

Hewlett Packard
Enterprise

Thank you

Frans.Jongma@hpe.com