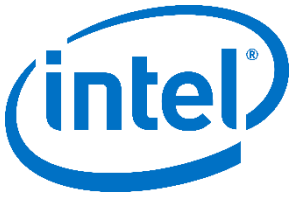
Hewlett Packard Enterprise



Using SQL/MX DBS

Demo using iTP webserver
May 2018

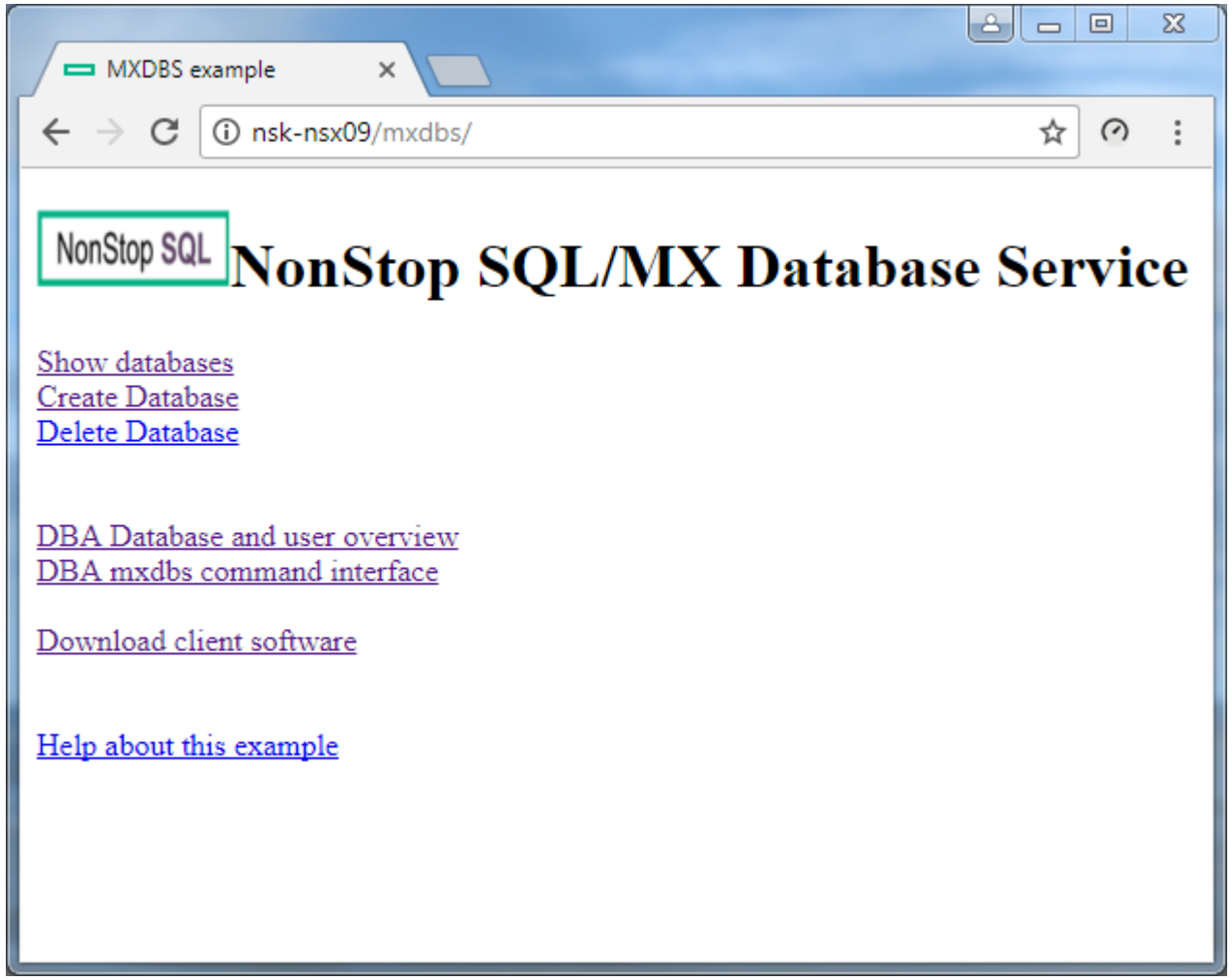
Frans Jongma, Advanced Technology Center

A guided tour through SQL/MX DBS using iTP Webserver

- Requirements
  - NonStop System
  - iTP Webserver running
  - Firefox or Chrome browser
  - SQL/MX DBS 3.5.1 installed and configured for DBS
  - The demopages installed
  - Softlinks for downloadable client software created
    - Use `mxcreatelinks -s -d /usr/tandem/sqlmx/downloads`
  - Navigate your browser to http://your_system/mxdbs

Demo Homepage

- This page sows the basic functions for this demo
- This is a demo, it lacks full access control
  - You could add user control to a system admin in iTP webserver
  - That way, you do not get self-service, but a system admin will be able to quickly provision databases for users that do not require access to the NonStop Operating Environment via bash or TACL
- Included are a few “DBA” pages that require a valid NonStop User-ID for verification
  - Not a complete solution



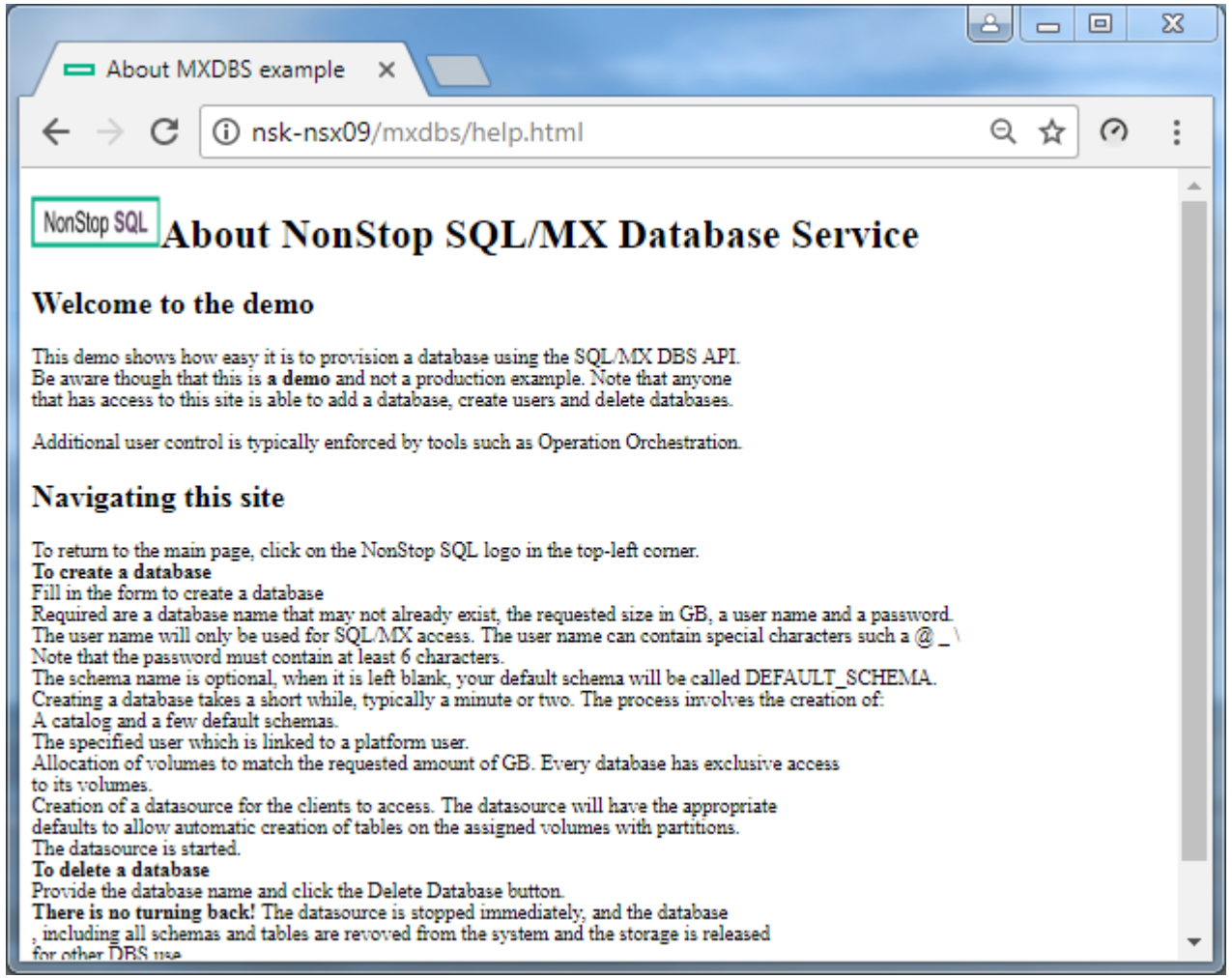
Hewlett Packard Enterprise

intel

A little explanation

– Read this first to learn how to navigate through these pages

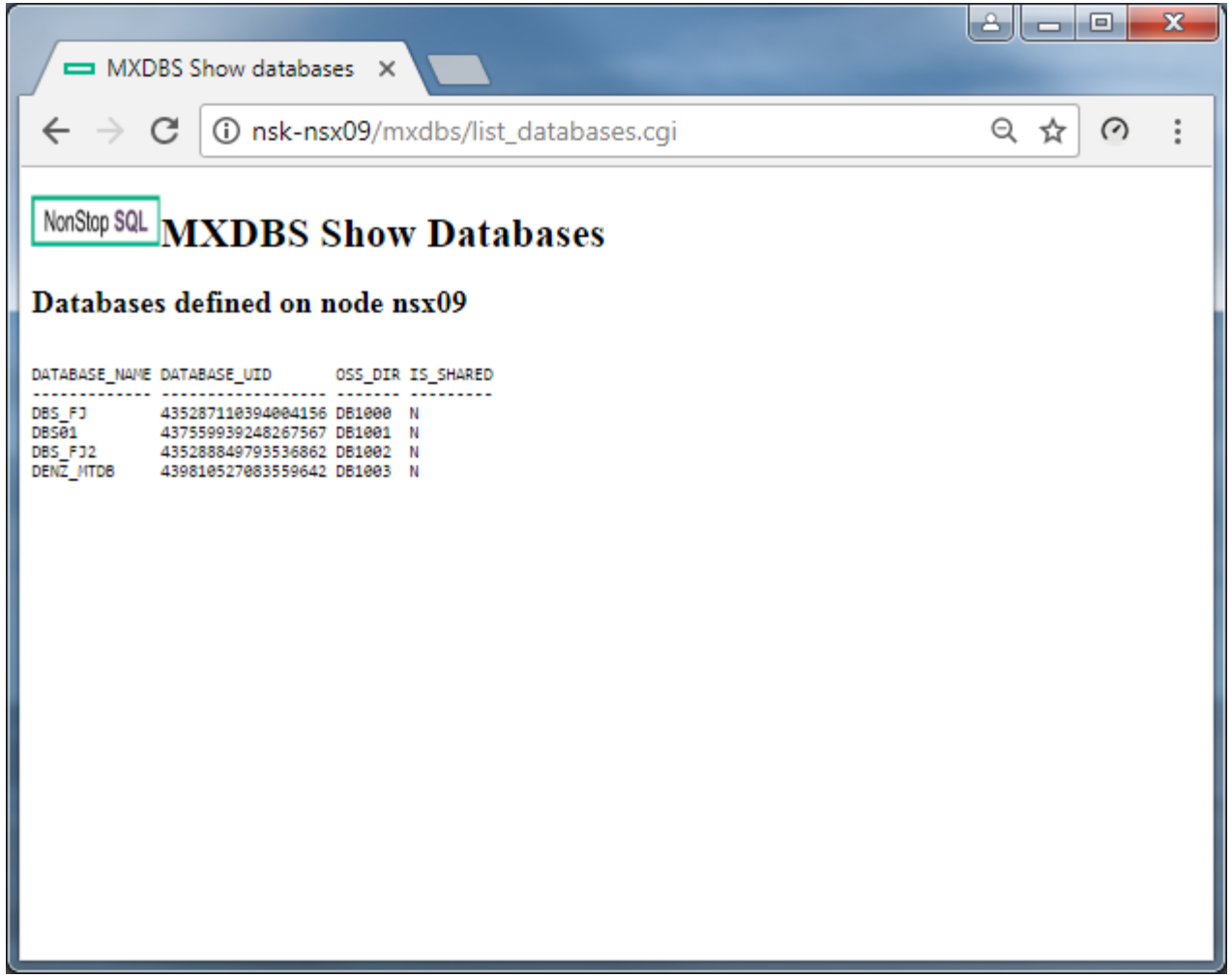
– Click NonStop SQL to return to the main page



See what is out there

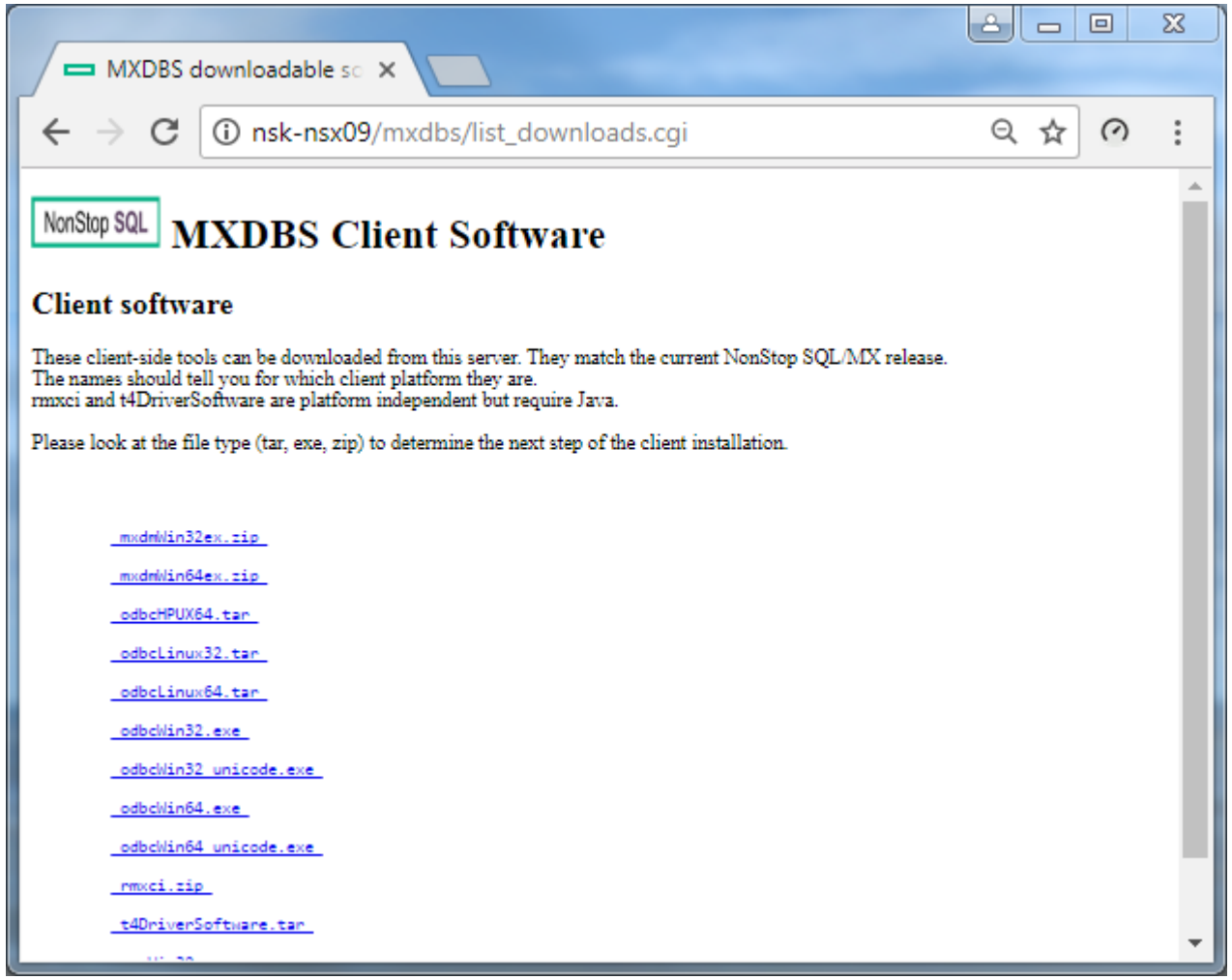
– Databases are unique on the system they are defined

– Before creating your database, see if your choice of name has not been defined already



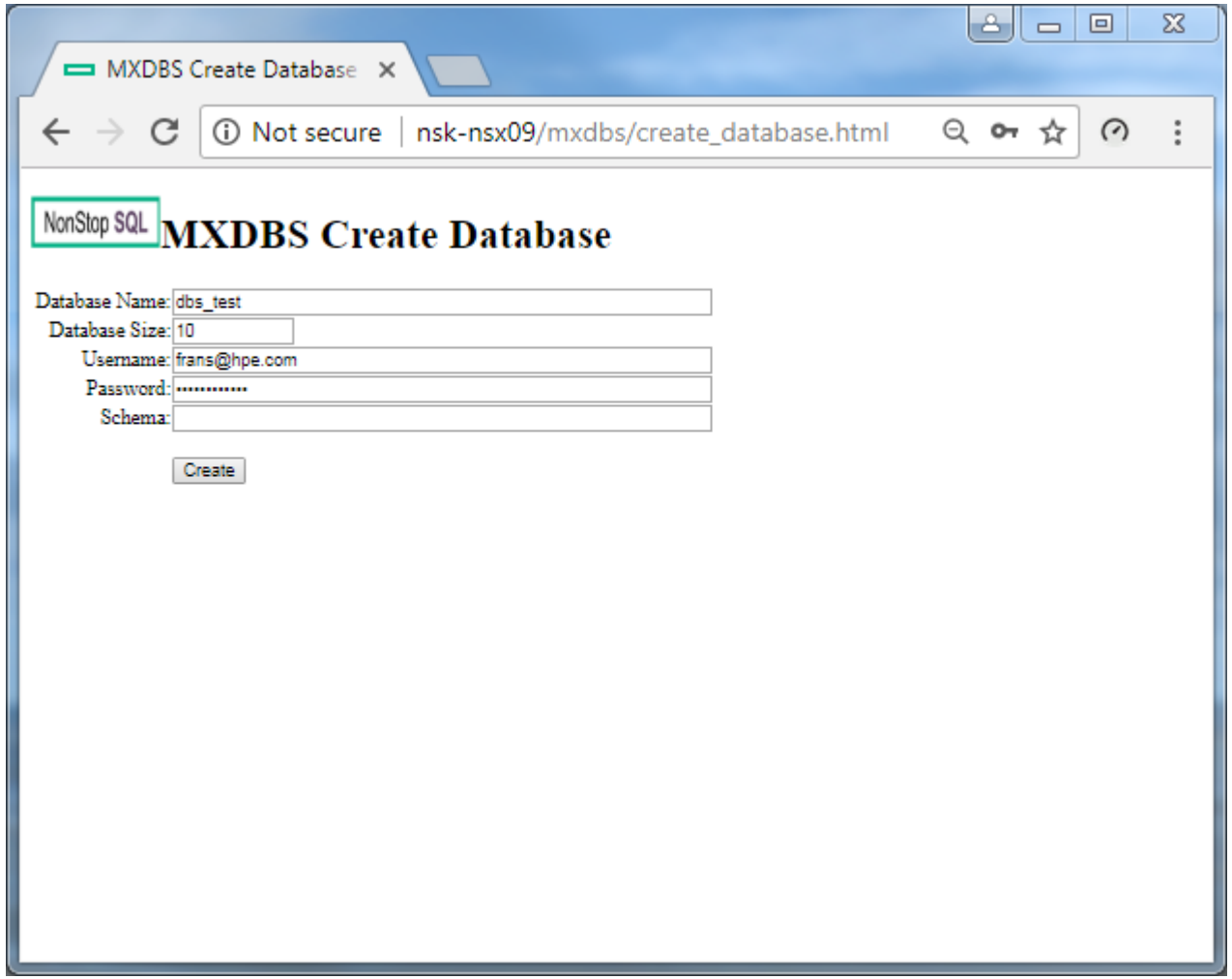
Download page for client software

- HPE documentation is written for system managers, not for end-users
- The same software can be downloaded using this page
  - These are soft links in OSS space that link to Guardian installation files
  - Unlike the Guardian files, the file names are self-explanatory
    - Unzip the zip files, untar the tar files and see what is in them to decide the next step.
    - For example, the mxdm zip files contain the Windows setup executable files.



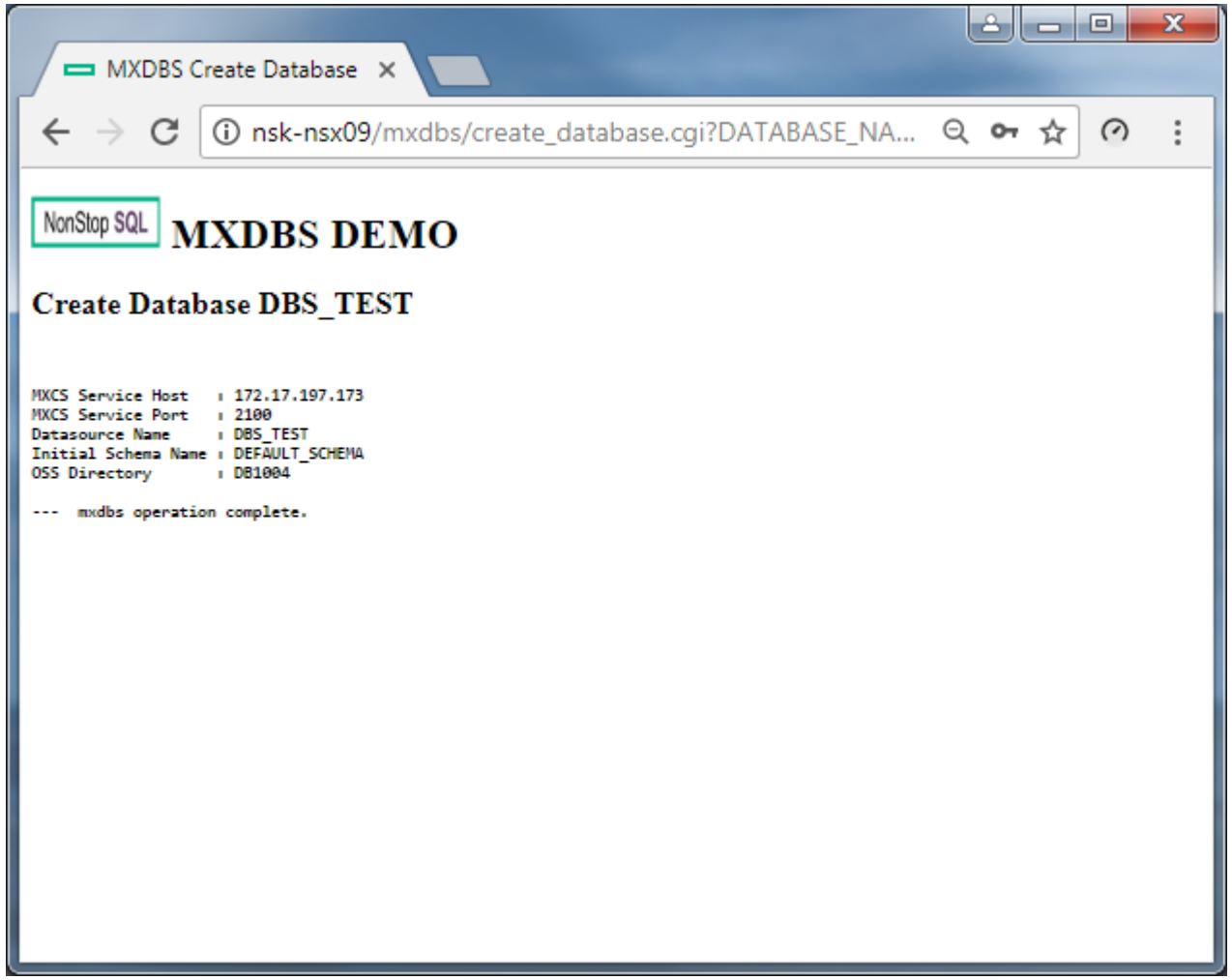
Create a database

– A database defines the user environment. Its name is used for
  – The name of the catalog
  – The name of an MXCS datasource
– For database and schema names, use SQL identifiers (i.e no spaces, no special characters except underscore _)
- Password must adhere to Safeguard rules (typically > 6 characters)
- User name is not case-sensitive, and may include special characters
  - Email addresses are OK (frans@hpe.com)
  - Windows domain names are OK (emea\frans)
  - Users are not case-sensitive unless so defined in the SYSTEM_DEFAULTS table
- Schema is optional, if not defined, DEFAULT_SCHEMA is created
- Database create takes about 1.5 -2 minutes



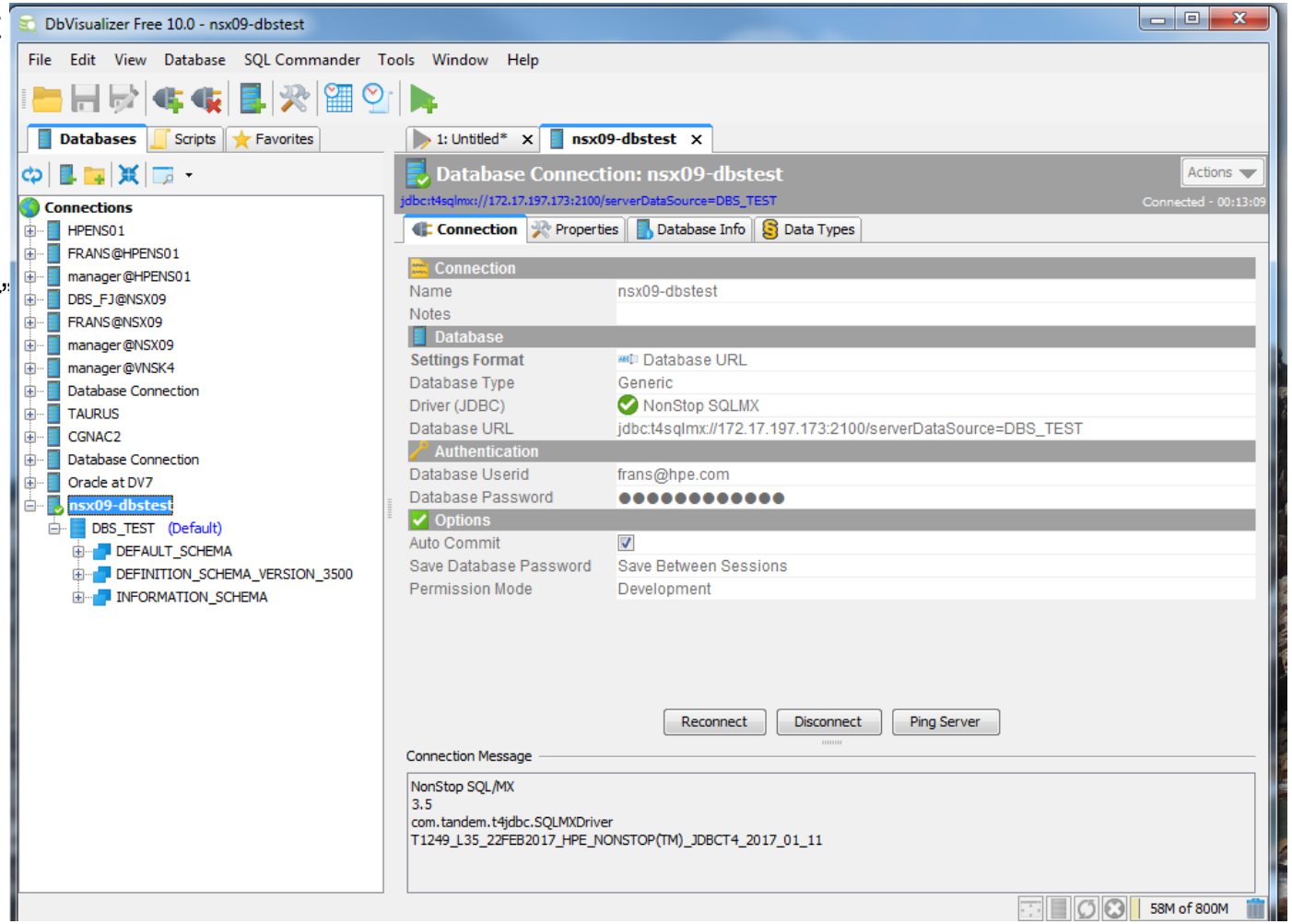
Database create results

- When create is finished, the status is reported
  - Connection information for host
    - Includes the default schema name that is used in MXCS
  - Typically port 2100 is used for MXCS (defined during DBS installation)
  - Datasource name to be used when a connection is made must be in uppercase.
    - ODBC and JDBC names are case-sensitive!



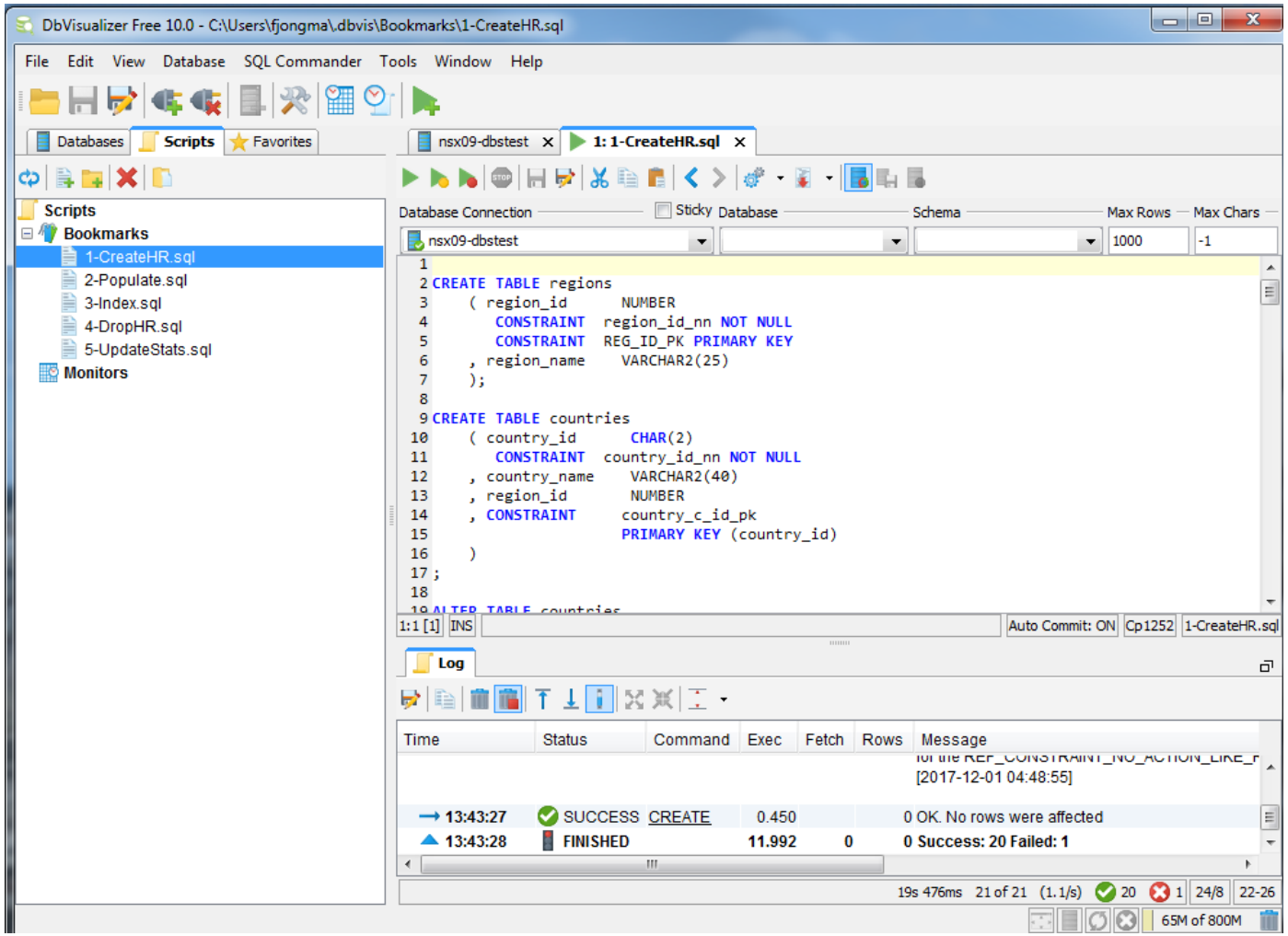
Time to try the connection

- This slide shows DbVisualizer, a popular tool that also has a free version with some limited functionality
- To connect, use the SQL/MX T4 driver
  - Driver is downloaded as part t4DriverSoftware.tar
  - Configure the driver using the DbVis “Driver Manager” window
  - Tip: In Driver Properties, set `java.sql.statement.setFetchSize 50`



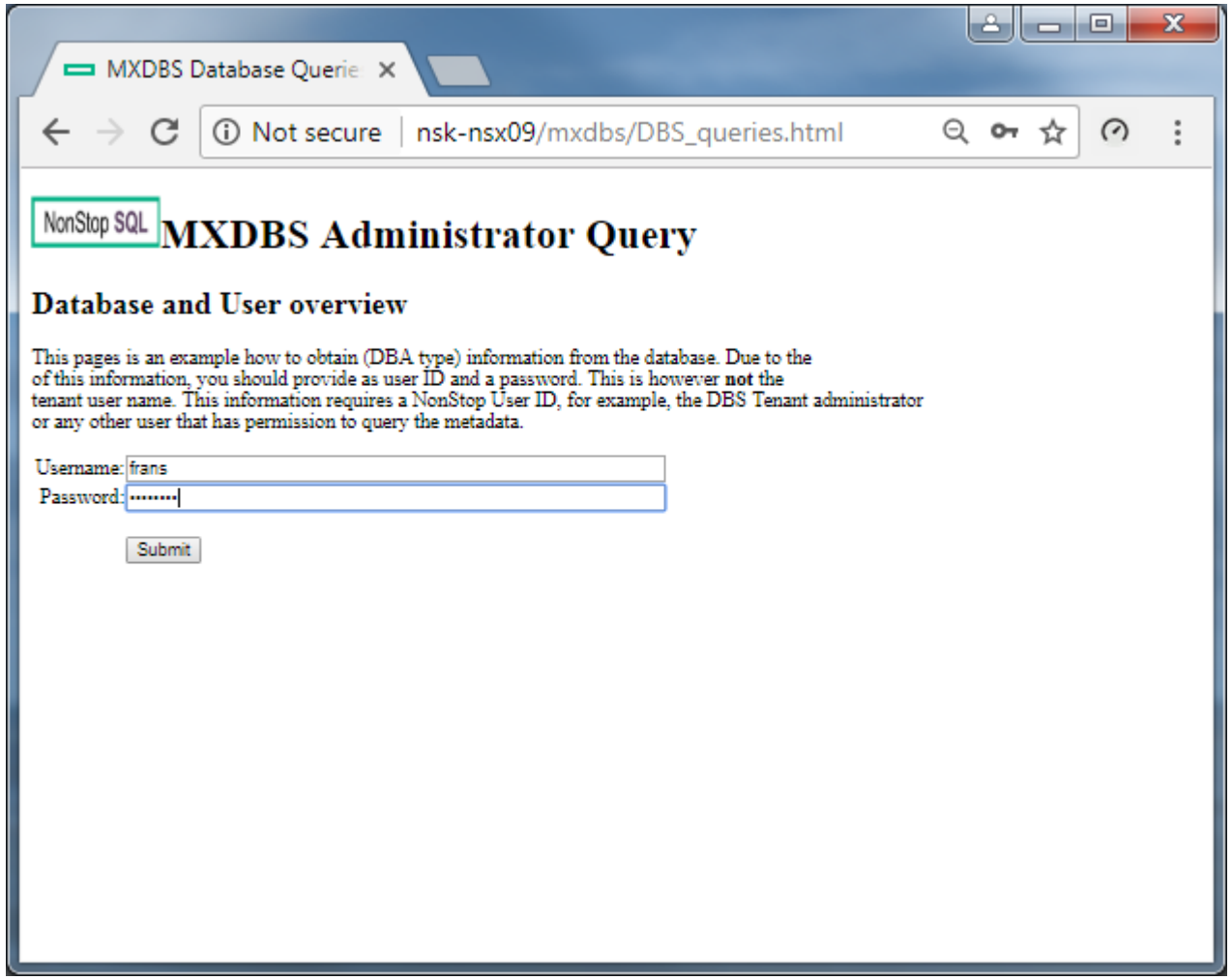
Use the database

- This example shows how tables are created
- Note the use of Database Compatibility features
  - REGION_ID NUMBER
  - REGION_NAME VARCHAR2



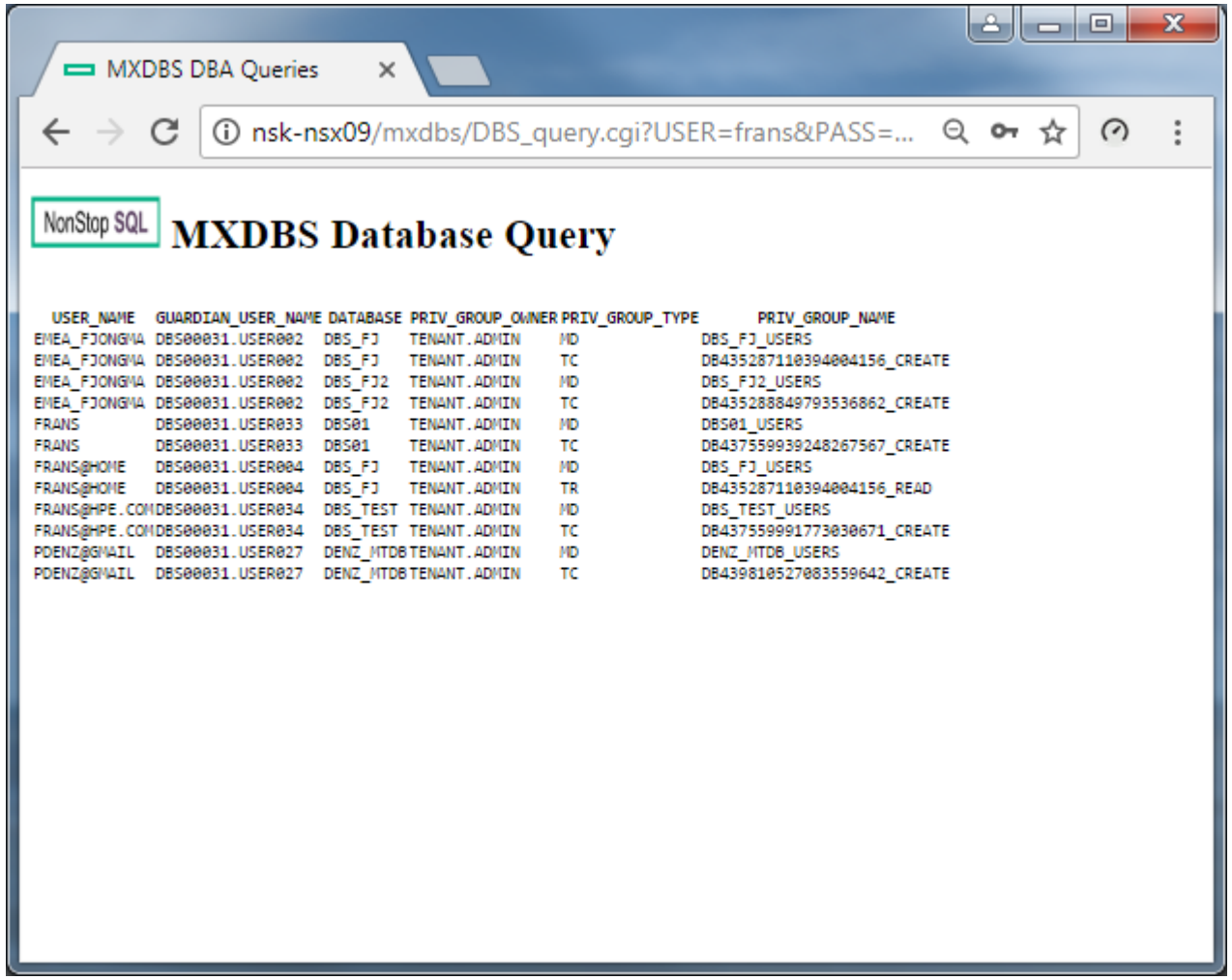
A DBA query (request)

- This is an example of how one might run metadata queries from a controlled user. This page invokes a cgi script on the host, and the script will only execute the query if the user ID and password are valid for system access.
- Tenant users (such as the created frans@hpe.com) cannot run these queries.



A DBA query (response)

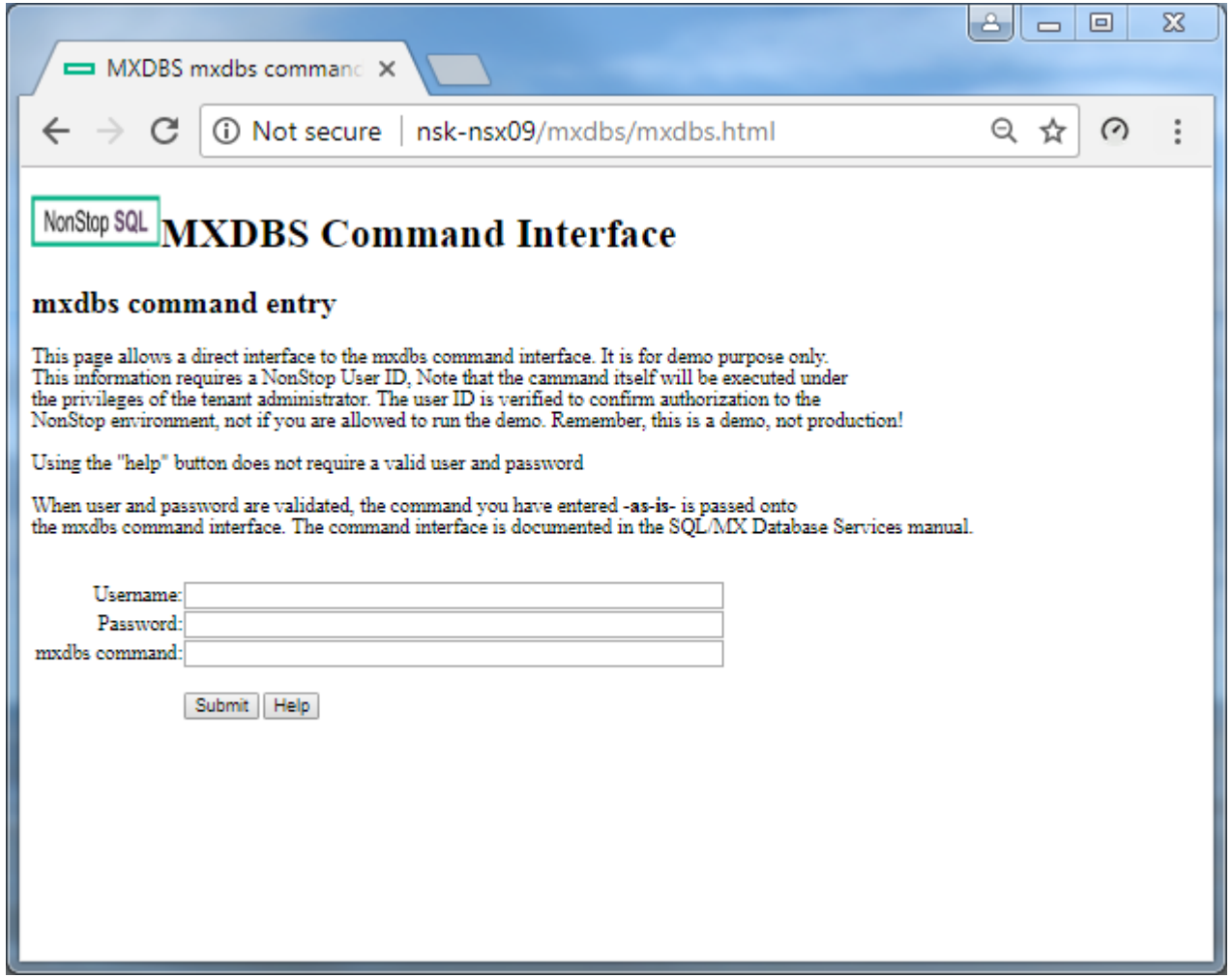
- This is an overview of external users and their Guardian equivalent
- Also shows to which databases and privilege groups they belong



USER_NAME	GUARDIAN_USER_NAME	DATABASE	PRIV_GROUP_OWNER	PRIV_GROUP_TYPE	PRIV_GROUP_NAME
EMEA_FJONGMA	DBS00031.USER002	DBS_FJ	TENANT.ADMIN	MD	DBS_FJ_USERS
EMEA_FJONGMA	DBS00031.USER002	DBS_FJ	TENANT.ADMIN	TC	DB435287110394004156_CREATE
EMEA_FJONGMA	DBS00031.USER002	DBS_FJ2	TENANT.ADMIN	MD	DBS_FJ2_USERS
EMEA_FJONGMA	DBS00031.USER002	DBS_FJ2	TENANT.ADMIN	TC	DB4352888497935368862_CREATE
FRANS	DBS00031.USER033	DBS01	TENANT.ADMIN	MD	DBS01_USERS
FRANS	DBS00031.USER033	DBS01	TENANT.ADMIN	TC	DB4375599392482675567_CREATE
FRANS@HOME	DBS00031.USER004	DBS_FJ	TENANT.ADMIN	MD	DBS_FJ_USERS
FRANS@HOME	DBS00031.USER004	DBS_FJ	TENANT.ADMIN	TR	DB435287110394004156_READ
FRANS@HPE.COM	DBS00031.USER034	DBS_TEST	TENANT.ADMIN	MD	DBS_TEST_USERS
FRANS@HPE.COM	DBS00031.USER034	DBS_TEST	TENANT.ADMIN	TC	DB4375599917730306671_CREATE
PDENZ@GMAIL	DBS00031.USER027	DENZ_MTDB	TENANT.ADMIN	MD	DENZ_MTDB_USERS
PDENZ@GMAIL	DBS00031.USER027	DENZ_MTDB	TENANT.ADMIN	TC	DB4398105270835596642_CREATE

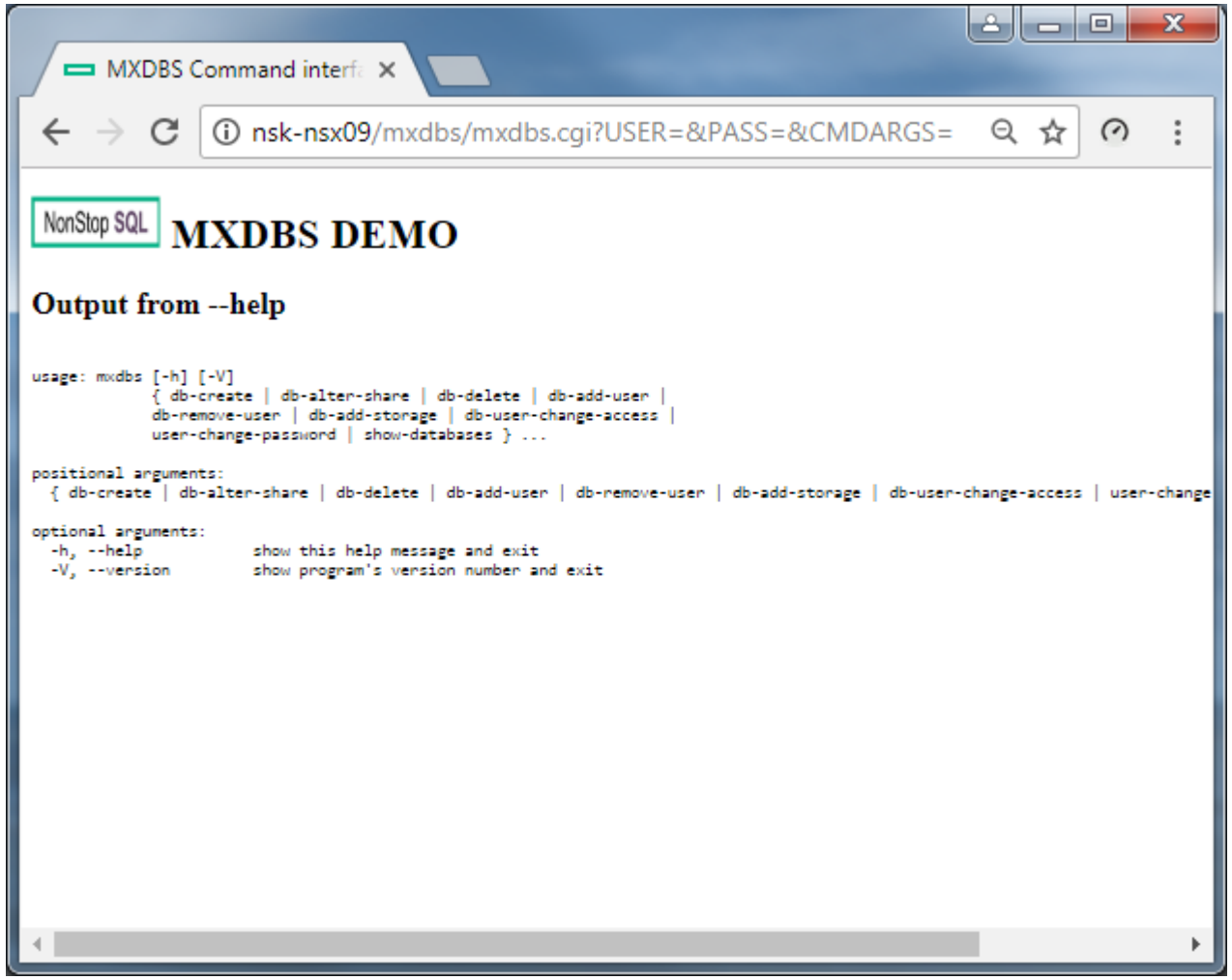
Another DBA function screen

- Just in case you need to invoke another mxdbs command
- User ID and password are required
  - Except if you press the help button

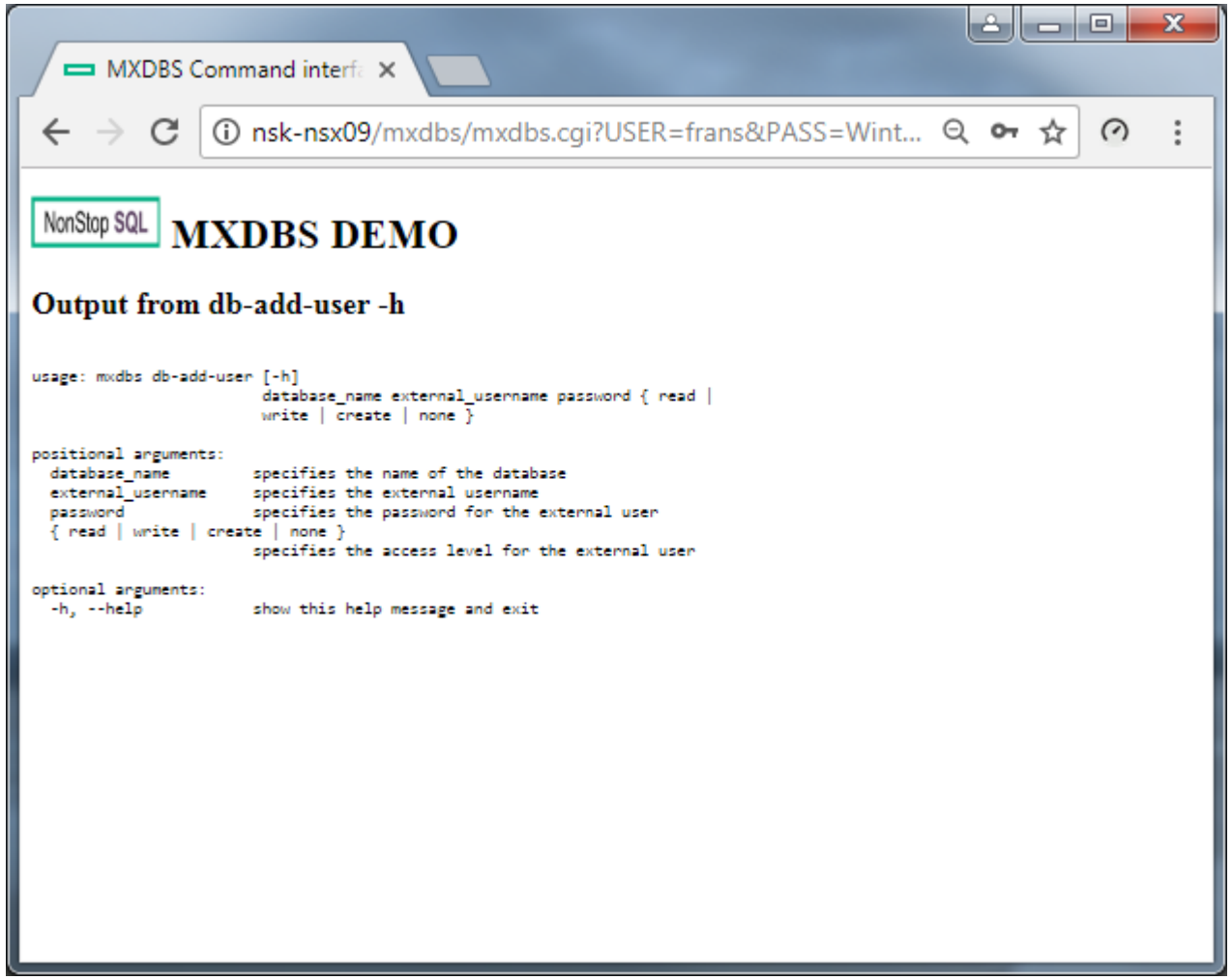


Another DBA function screen (help output)

- The help button invokes the mxdbs –help command and this is the output.
- Alternatively you can enter the command (or specific help commands) on the input screen

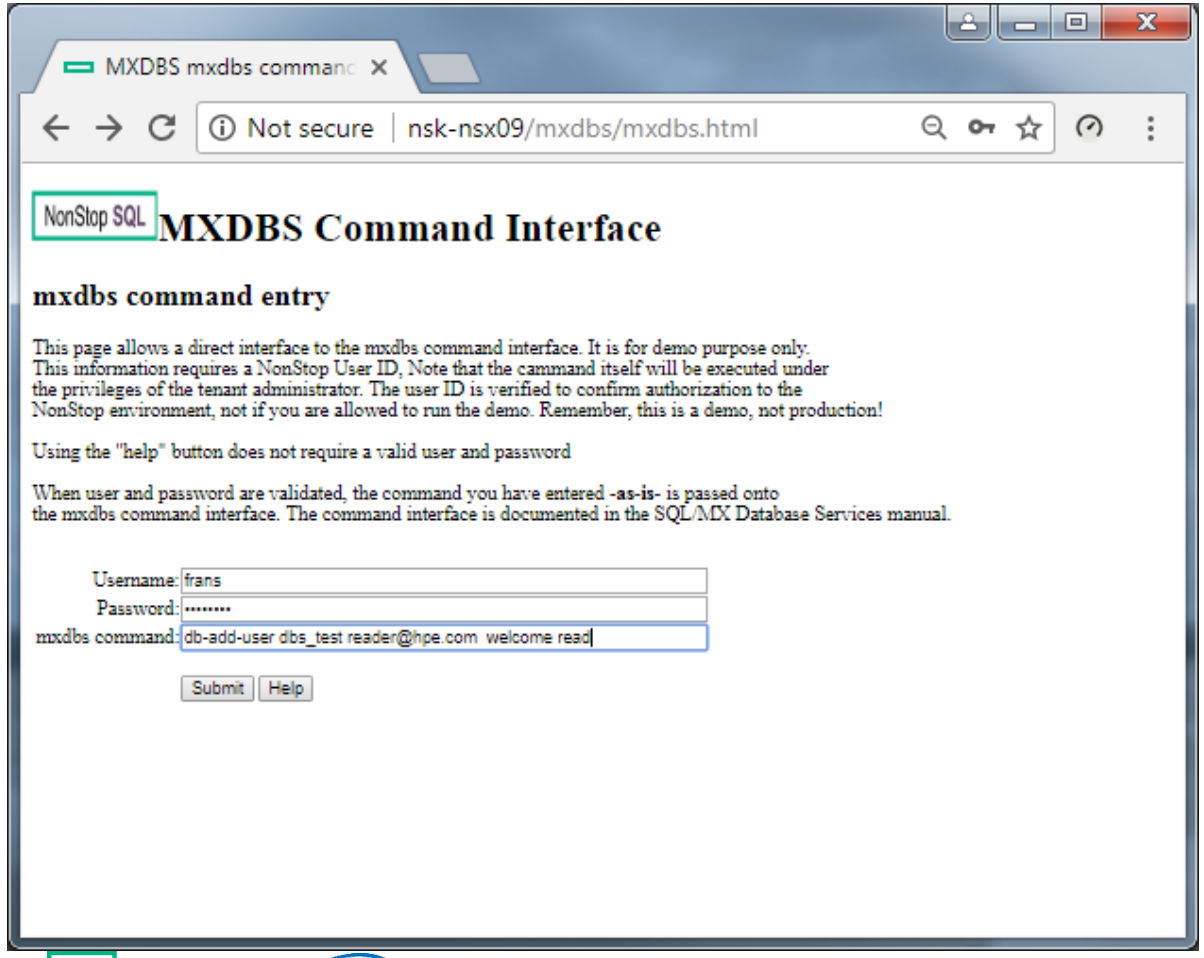


Another DBA function screen



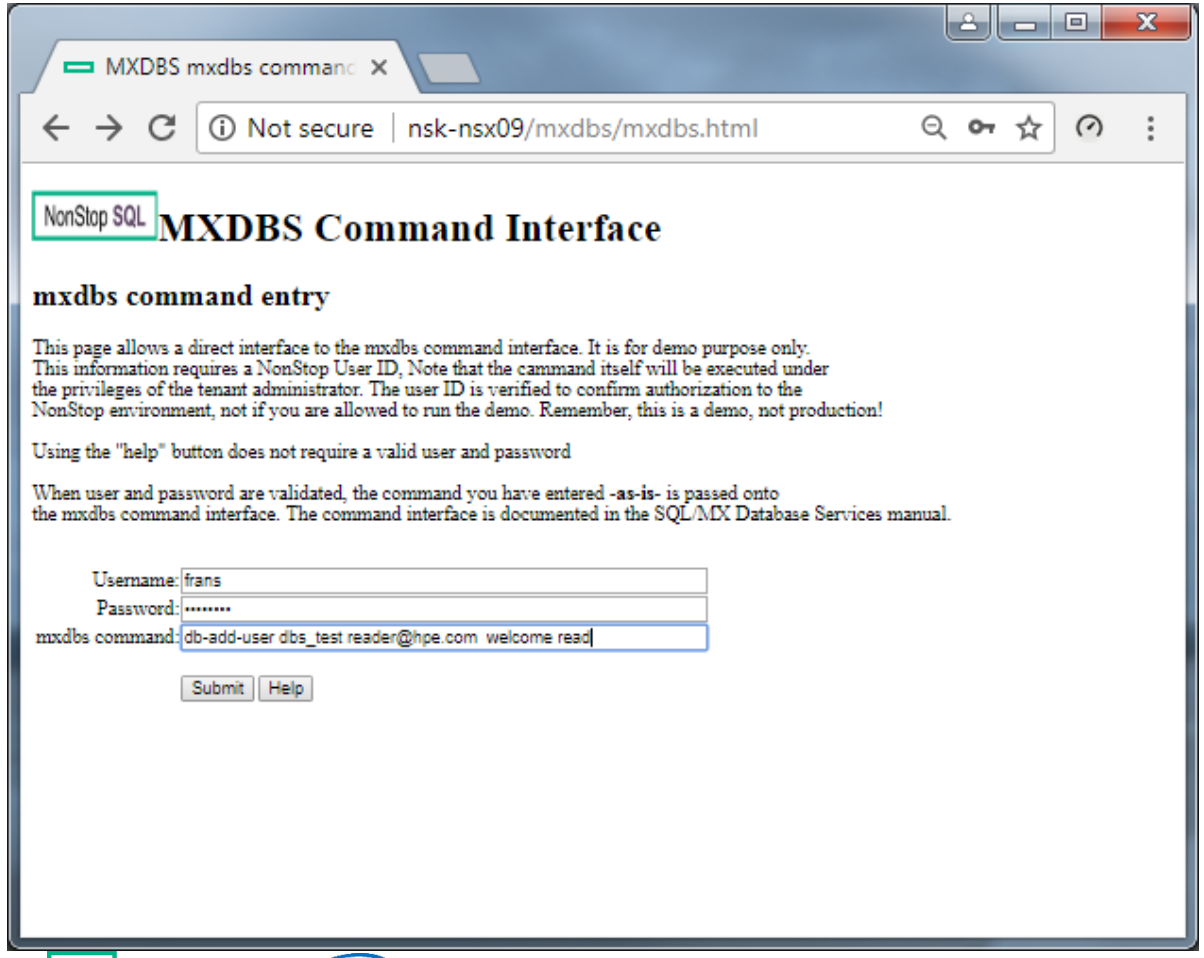
Another DBA function screen

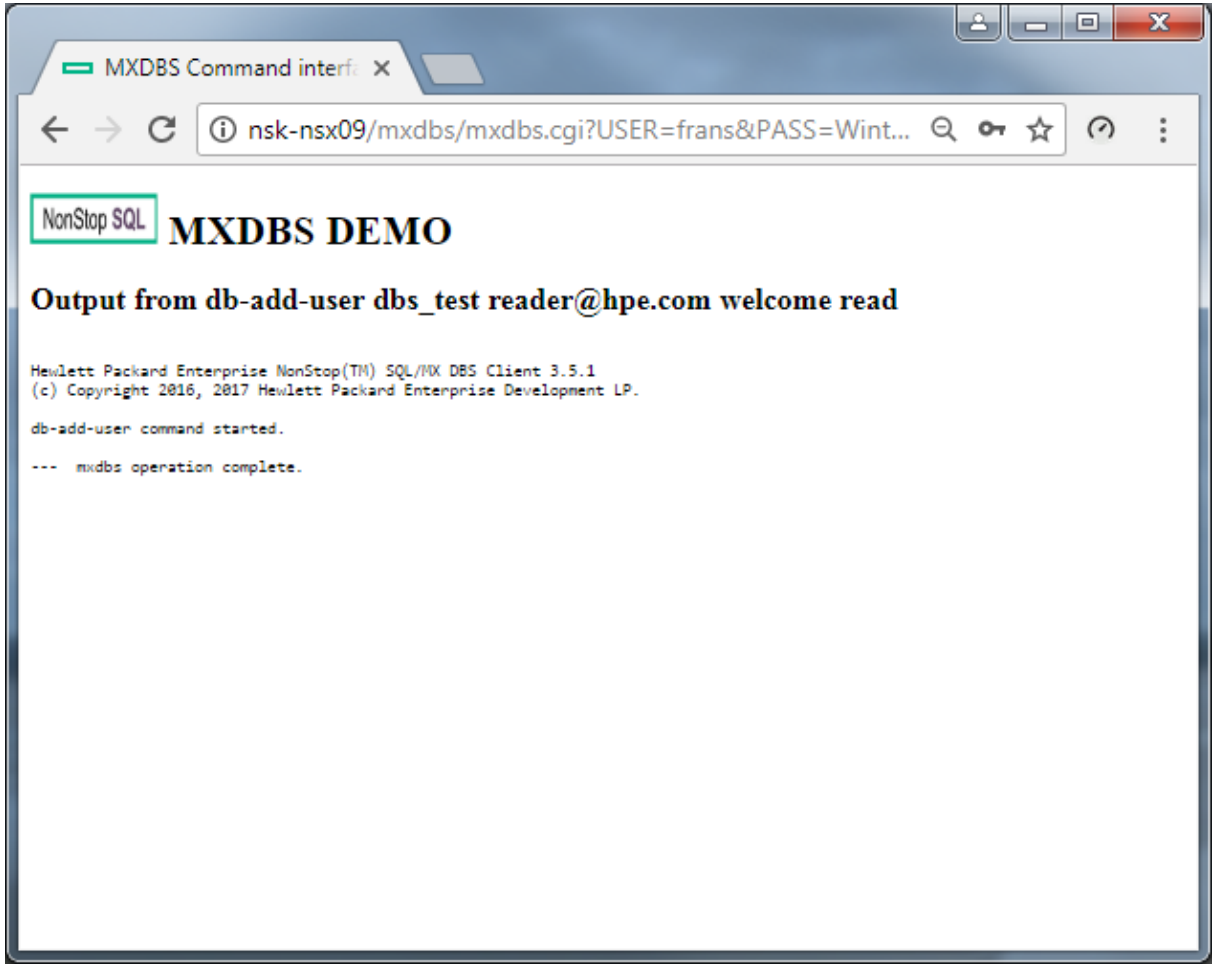
– Add a new user to a database



Another DBA function screen

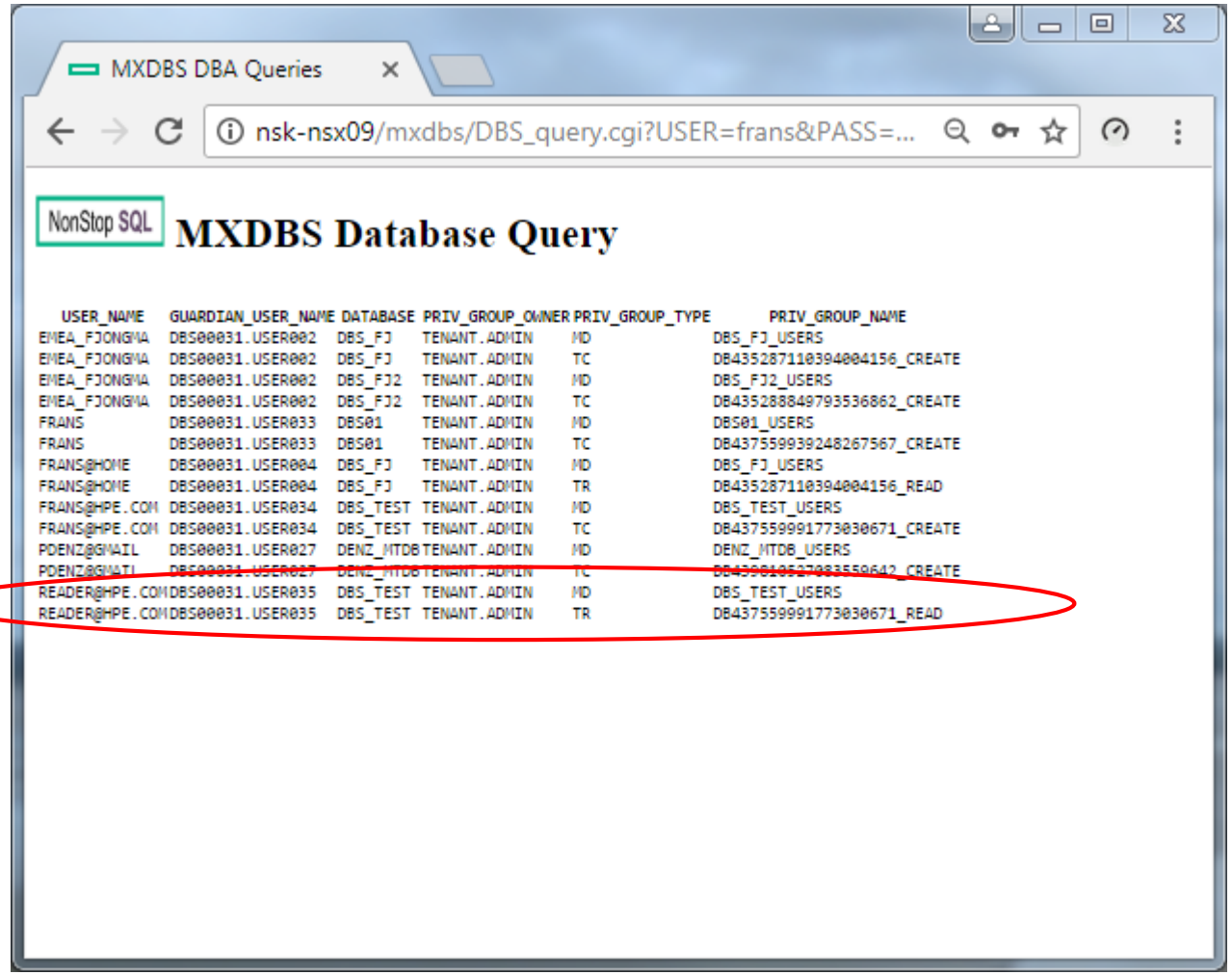
– Add a new user to a database





A DBA query (response)

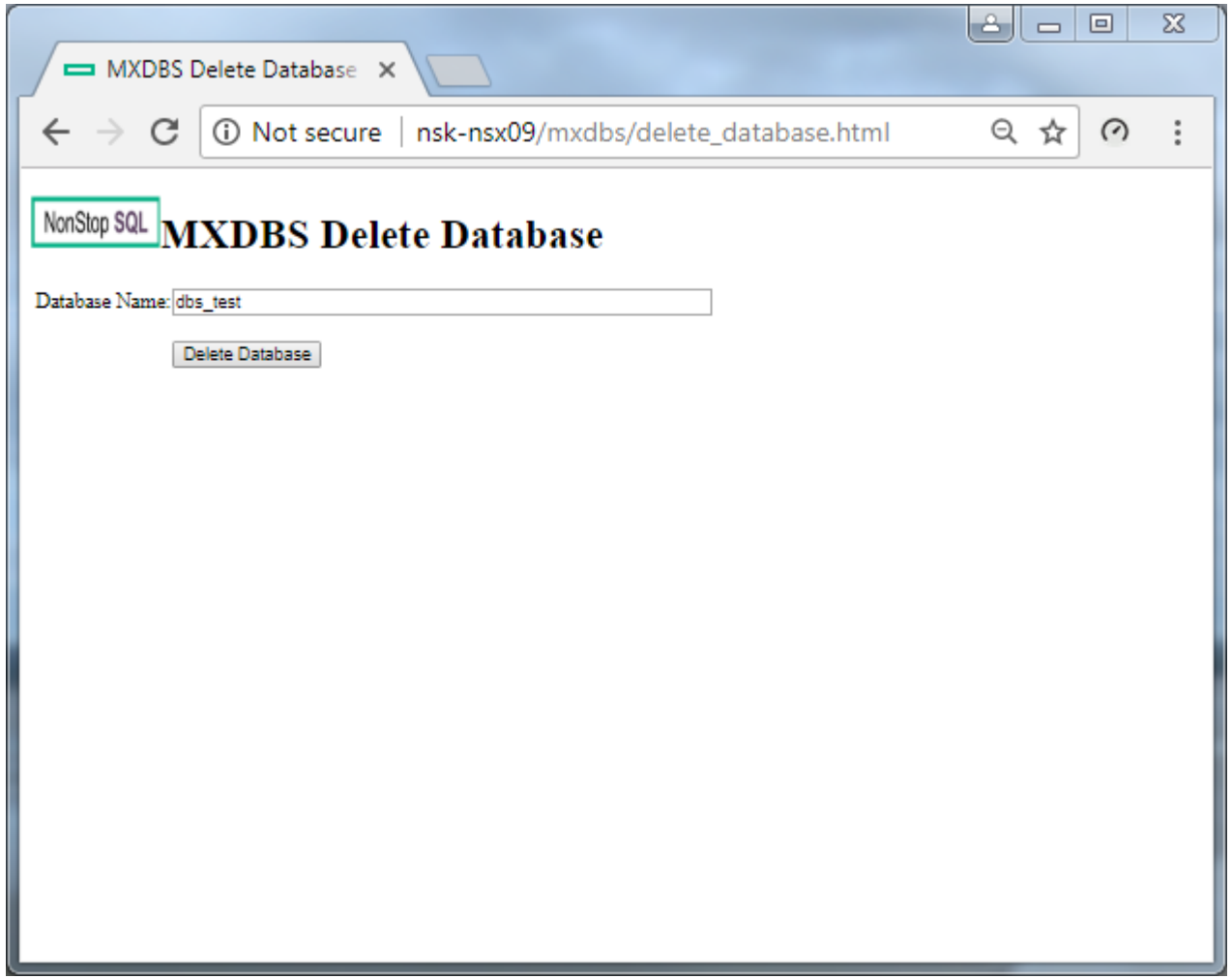
– To show that reader@hpe.com has been added as a user to database DBS_TEST



USER_NAME	GUARDIAN_USER_NAME	DATABASE	PRIV_GROUP_OWNER	PRIV_GROUP_TYPE	PRIV_GROUP_NAME
EMEA_FJONGMA	DBS00031.USER002	DBS_FJ	TENANT.ADMIN	MD	DBS_FJ_USERS
EMEA_FJONGMA	DBS00031.USER002	DBS_FJ	TENANT.ADMIN	TC	DB435287110394004156_CREATE
EMEA_FJONGMA	DBS00031.USER002	DBS_FJ2	TENANT.ADMIN	MD	DBS_FJ2_USERS
EMEA_FJONGMA	DBS00031.USER002	DBS_FJ2	TENANT.ADMIN	TC	DB435288849793536862_CREATE
FRANS	DBS00031.USER033	DBS01	TENANT.ADMIN	MD	DBS01_USERS
FRANS	DBS00031.USER033	DBS01	TENANT.ADMIN	TC	DB437559939248267567_CREATE
FRANS@HOME	DBS00031.USER004	DBS_FJ	TENANT.ADMIN	MD	DBS_FJ_USERS
FRANS@HOME	DBS00031.USER004	DBS_FJ	TENANT.ADMIN	TR	DB435287110394004156_READ
FRANS@HPE.COM	DBS00031.USER034	DBS_TEST	TENANT.ADMIN	MD	DBS_TEST_USERS
FRANS@HPE.COM	DBS00031.USER034	DBS_TEST	TENANT.ADMIN	TC	DB437559991773030671_CREATE
PDENZ@GMAIL	DBS00031.USER027	DENZ_MTDB	TENANT.ADMIN	MD	DENZ_MTDB_USERS
PDENZ@GMAIL	DBS00031.USER027	DENZ_MTDB	TENANT.ADMIN	TC	DB439610517083559642_CREATE
READER@HPE.COM	DBS00031.USER035	DBS_TEST	TENANT.ADMIN	MD	DBS_TEST_USERS
READER@HPE.COM	DBS00031.USER035	DBS_TEST	TENANT.ADMIN	TR	DB437559991773030671_READ

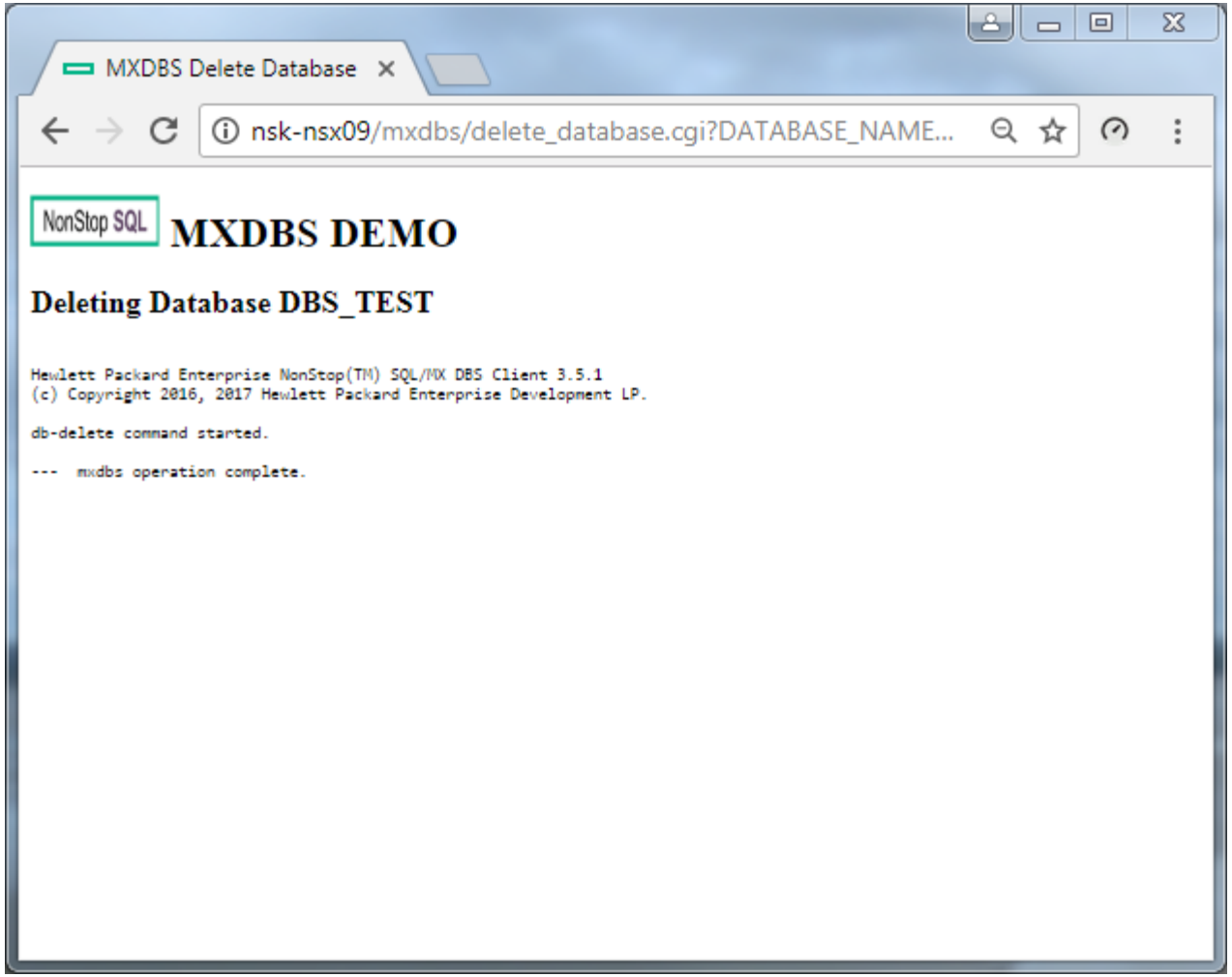
Cleaning up the database

- Free al the resources with a single command
- Drops the catalog and all the objects in it
- Stops and deletes the datasource
- Releases the storage back to the DBS storage pool
- Delete action takes about a minute



Database delete results

– After delete action finishes, the result of the action is displayed.



Summary

– SQL/MX DBS makes life easy for a DBA

– The demo shows “how” easy

– Remember, these HTML pages are not secure enough for production purpose

– They show the principle and making access secure is not rocket-science

– HTML and scripts are available

  – Working on packaging – send me an email

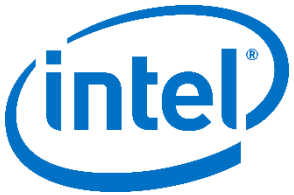
  – No warranty!

Hewlett Packard
Enterprise

Thank you

Frans.Jongma@hpe.com

Hewlett Packard Enterprise



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?

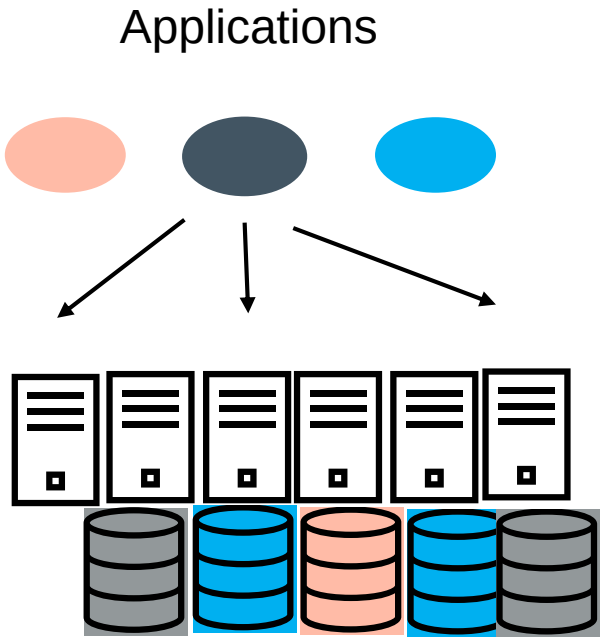
- A means to provide “Database as a Service” (DBaaS)
- SQL/MX DBS provides multi-tenant features with added user isolation

- 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

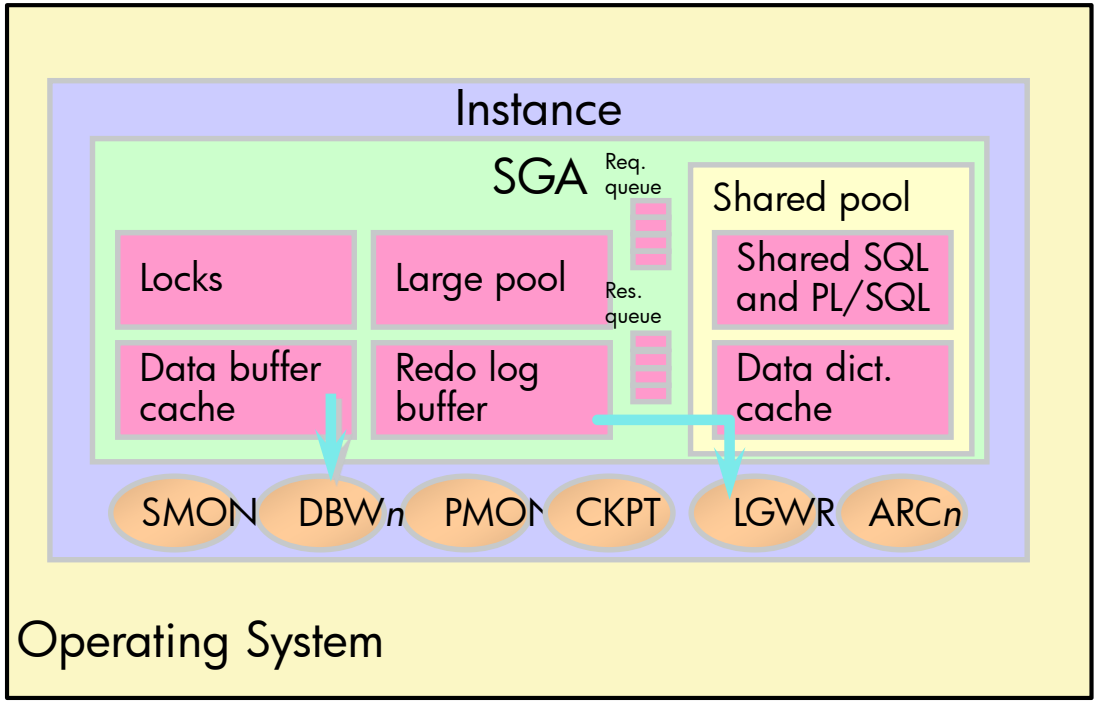
- Different products different names
- What does “Database” mean?
- What is an “Instance”?
- What is a User?

- 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

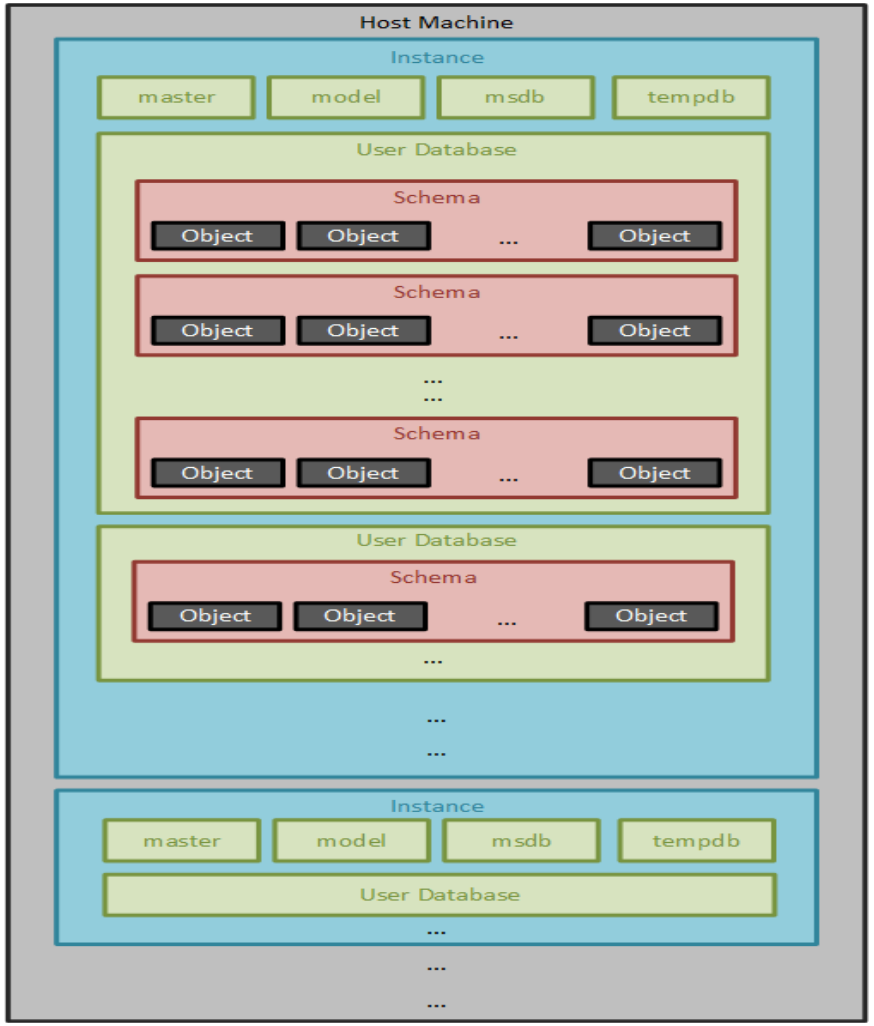
- DATABASE
  - Involves creating an instance, starting it and issue CREATE DATABASE command
    - Includes specification of SYS and SYSTEM users, logfiles, system and other tablespaces
- USER
  - User is an account through which you can log into the database; a way to get access
  - **`CREATE USER IDENTIFIED 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)



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.

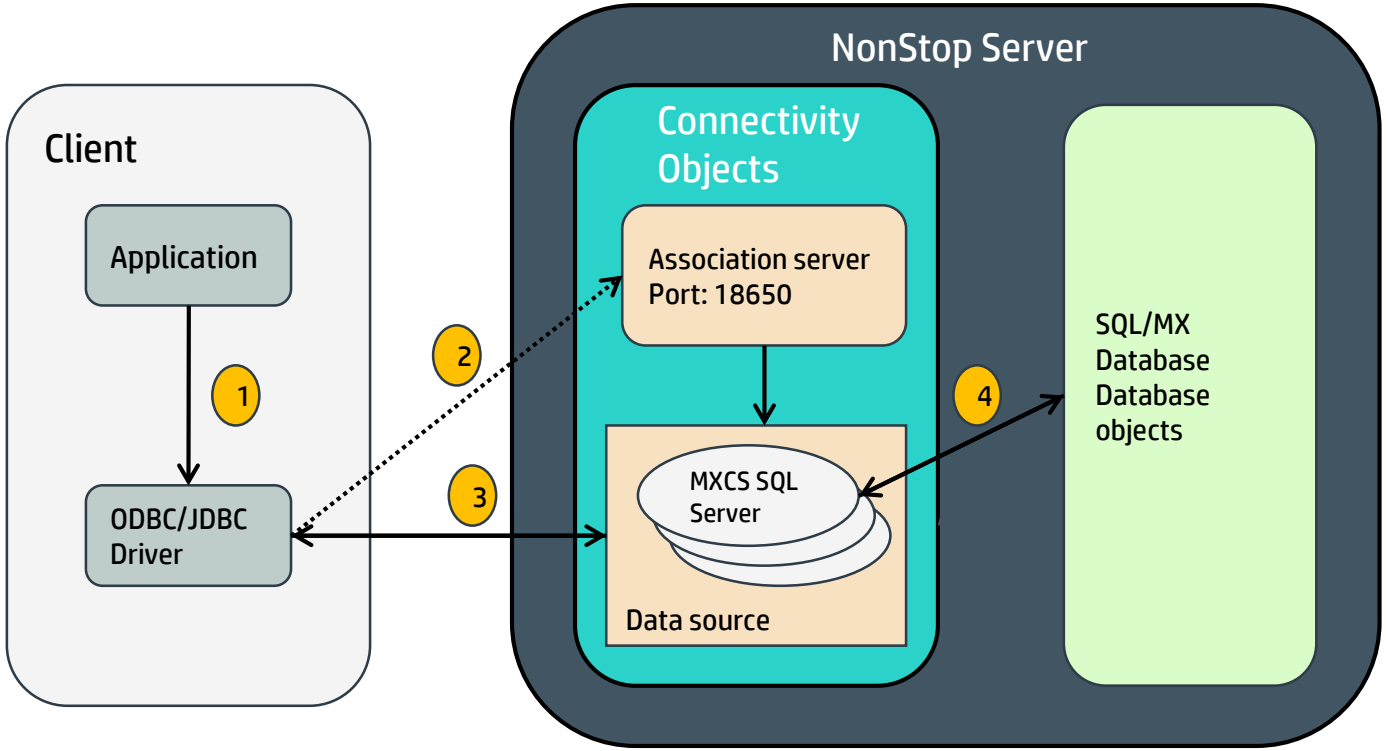
- **NonStop SQL Instance**
  - NonStop SQL is integral part of the NonStop OS
    - Find database engine components in libraries and Disk Access Managers (DAM)
  - If the system is up, the database is up
  - The OS equals “the instance”; all databases on a system are managed by the same version of the 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)

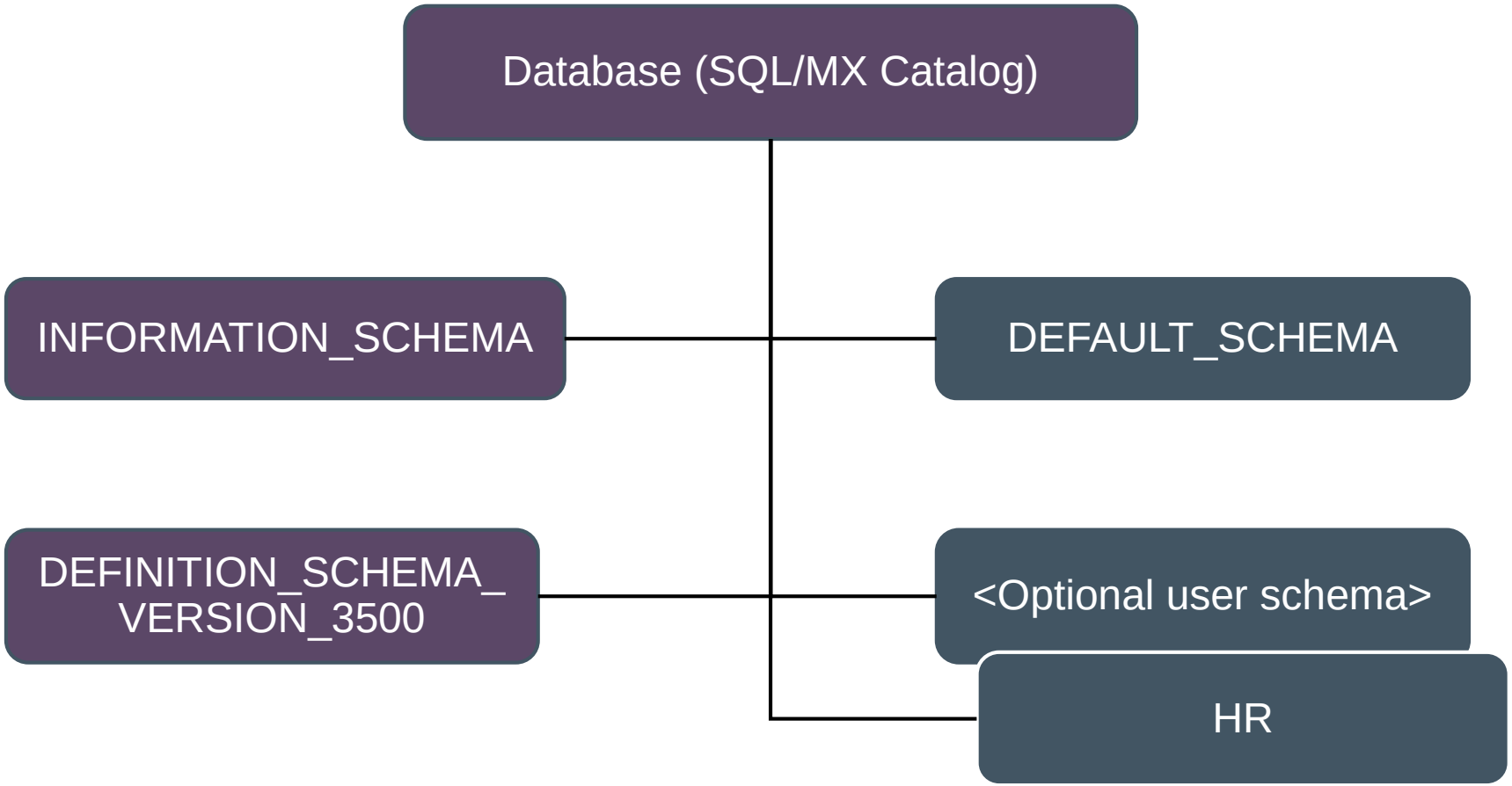
Data Source

- **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**

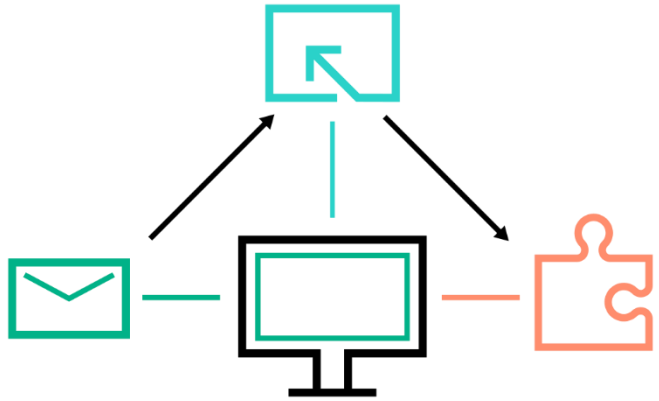
Quickly provision a database

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

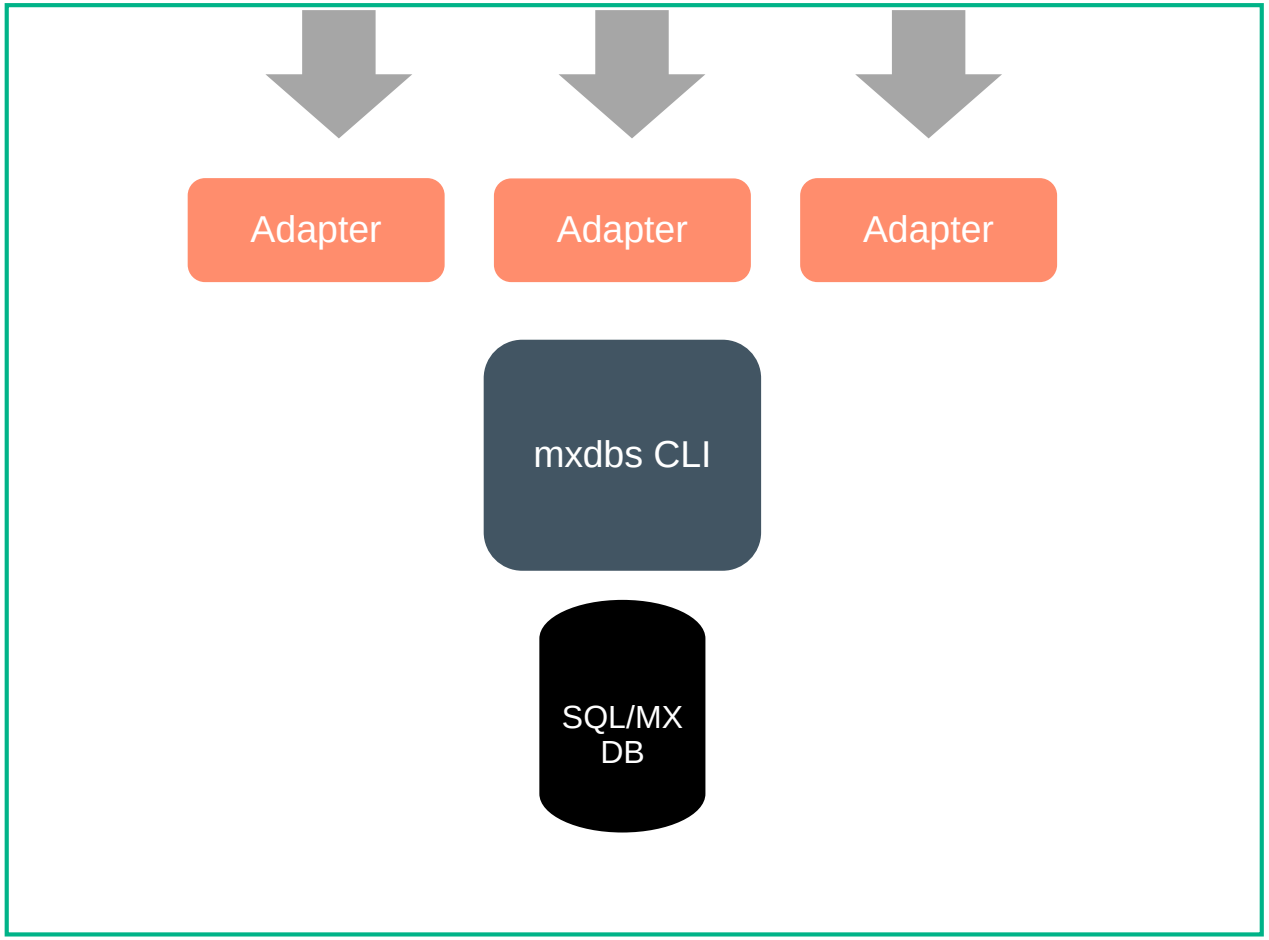
The API should be kept simple, database agnostic
The goal is to cover the life cycle management scope from long term projects to ad-hoc projects.



```
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 } ...
```

One API to support multiple clients and protocols

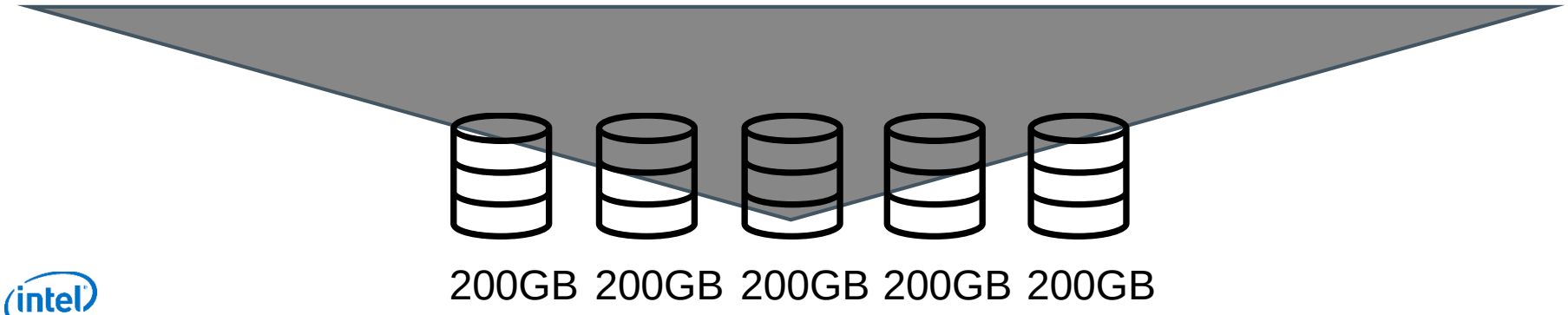
– Protocol examples
  – SSH
  – HTTPS
  – Message bus (AMQP)
  – Local execute/launch
– Client examples
  – HPE Operations Orchestration
  – Ansible
  – Client issuing REST API calls
  – Openstack Trove



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.
```

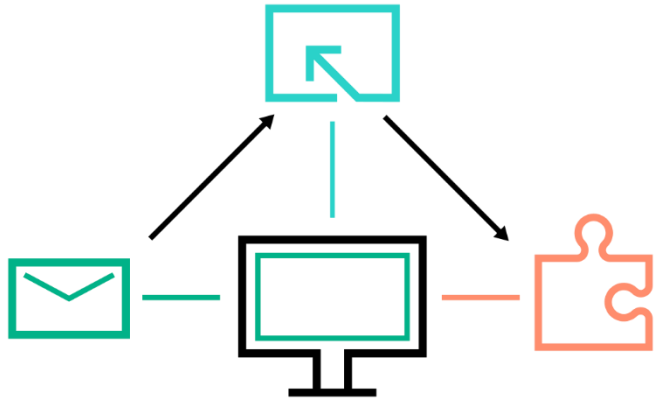
Details Storage allocation

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 } ...
```

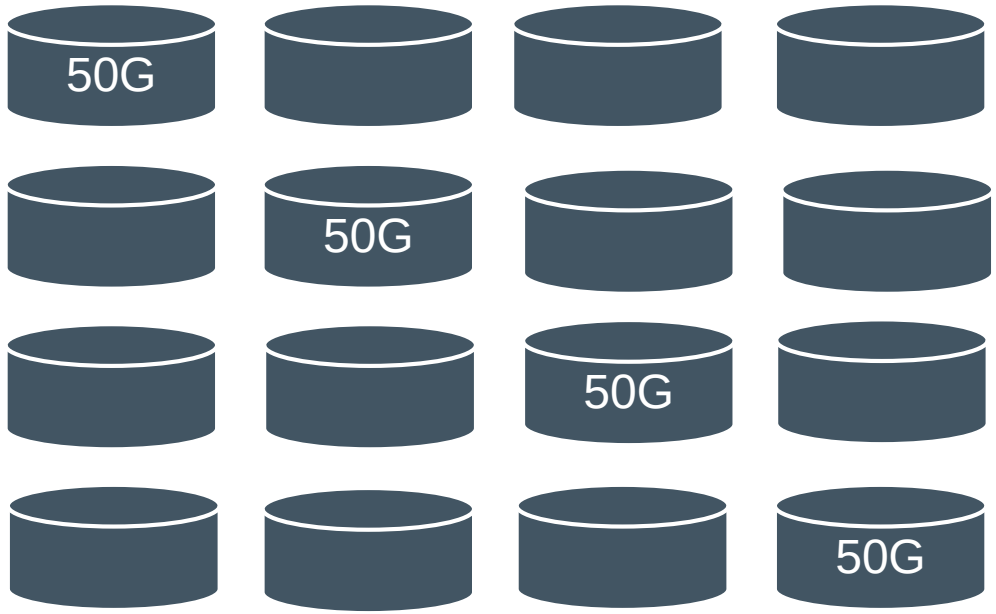
Hewlett Packard Enterprise intel

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

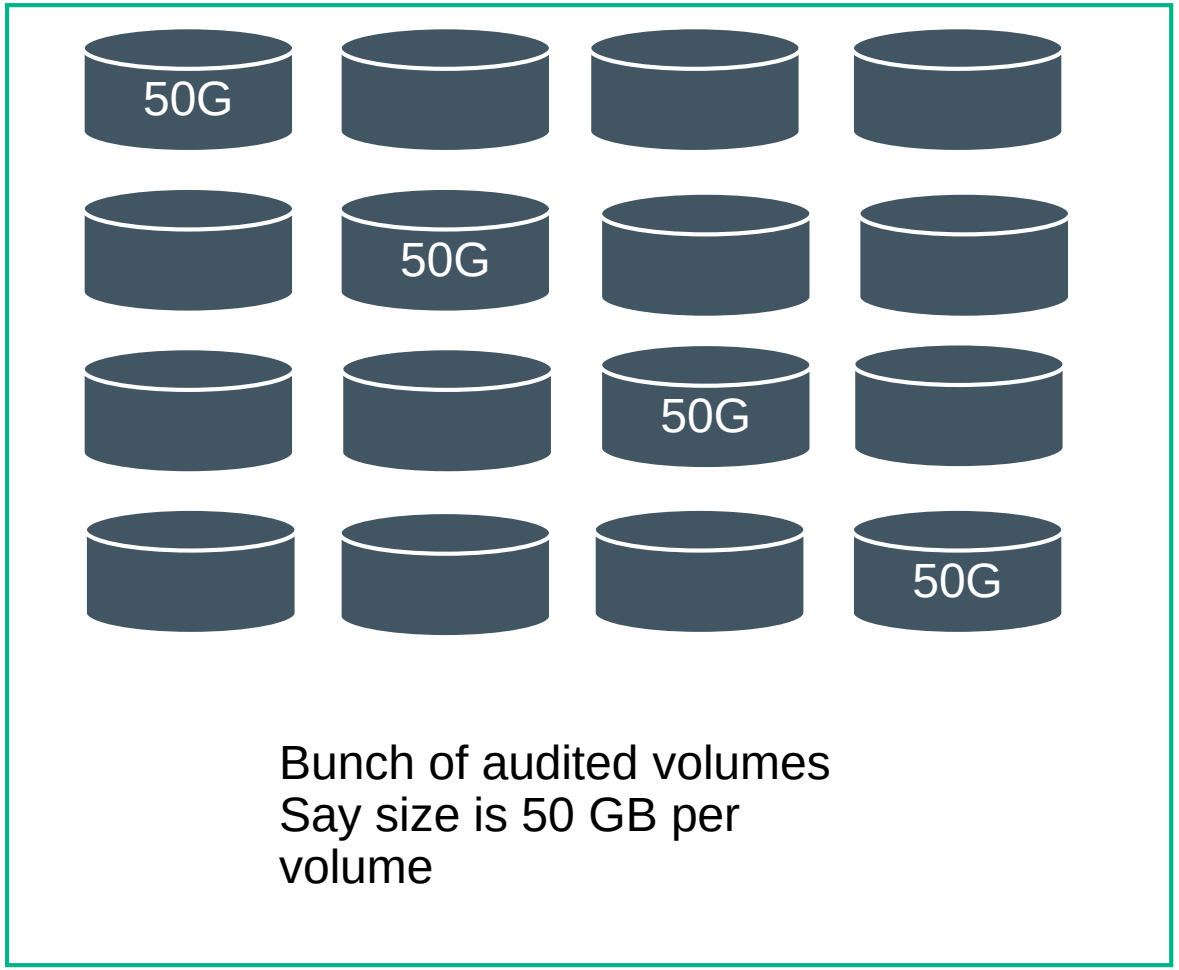
– Bunch of Storage for DBS



Bunch of audited volumes
Say size is 50 GB per volume

User requests a database (1)

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



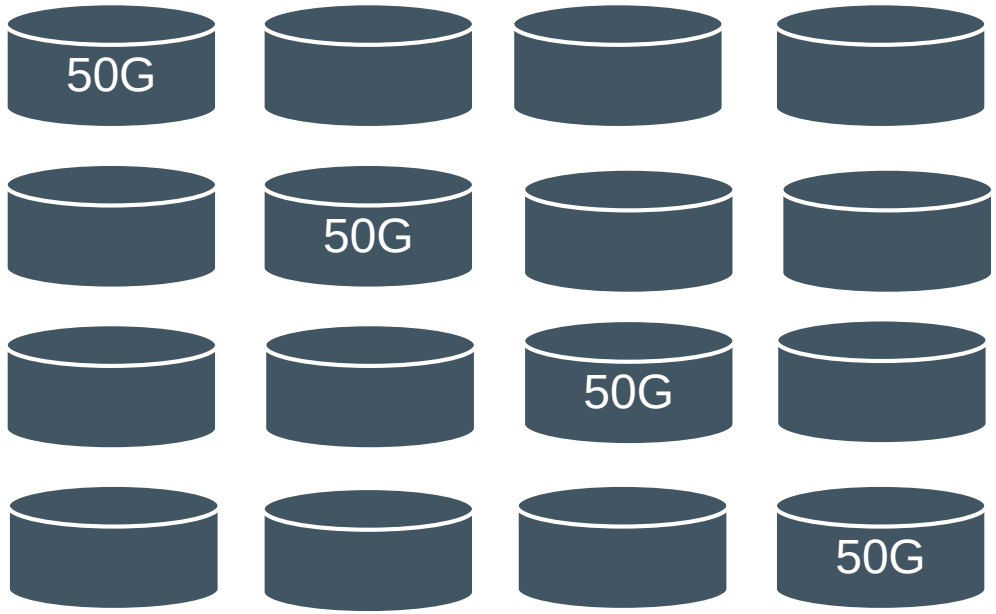
User requests a database (2)

– 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
```

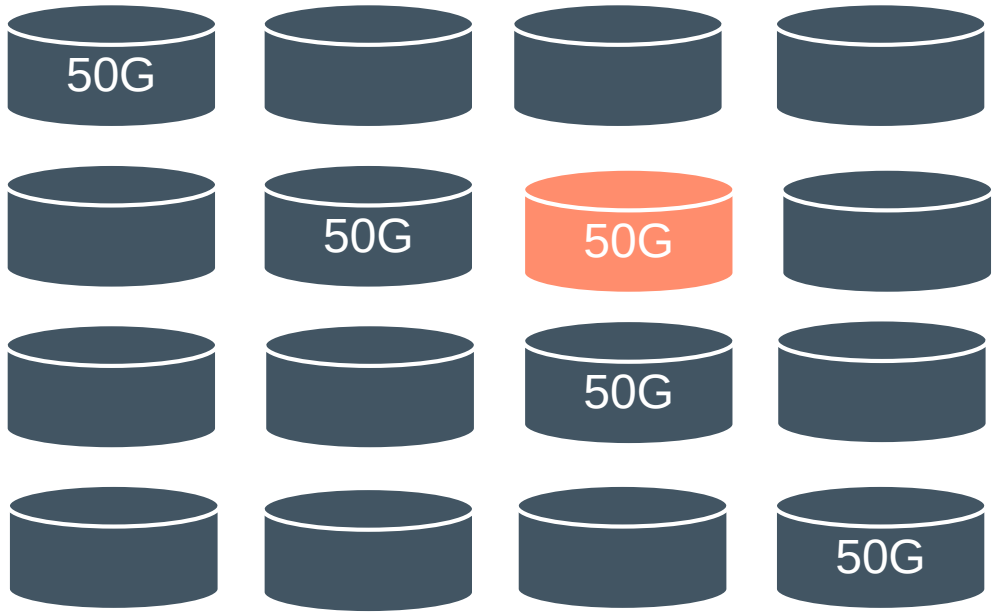


Bunch of audited volumes
Say size is 50 GB per volume

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.
```

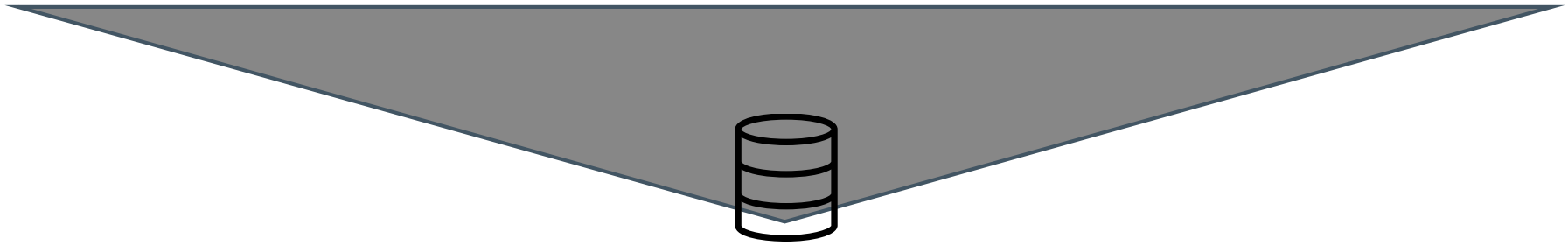


One volume (50GB)
assigned to DB_FRANSJ

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
```



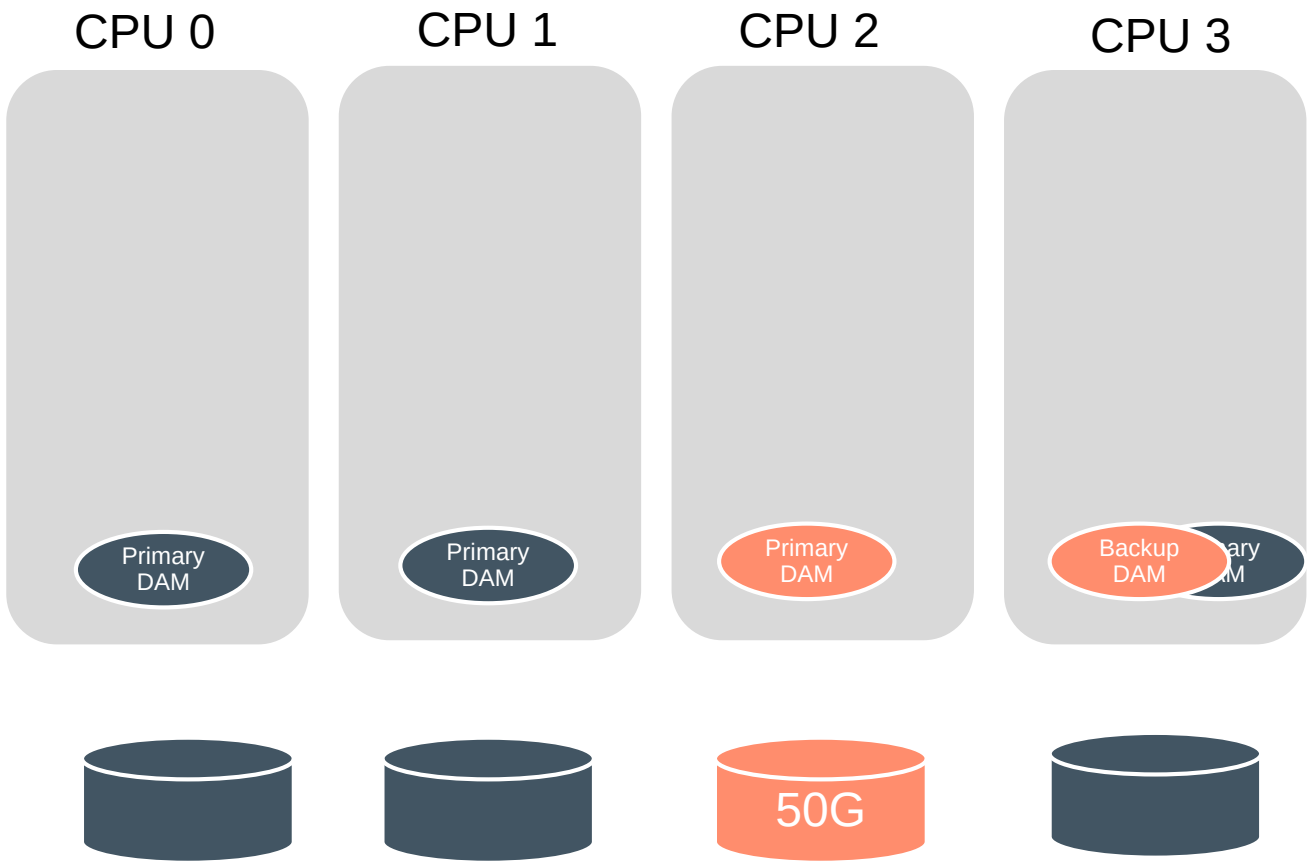
50GB

Details Compute assignment

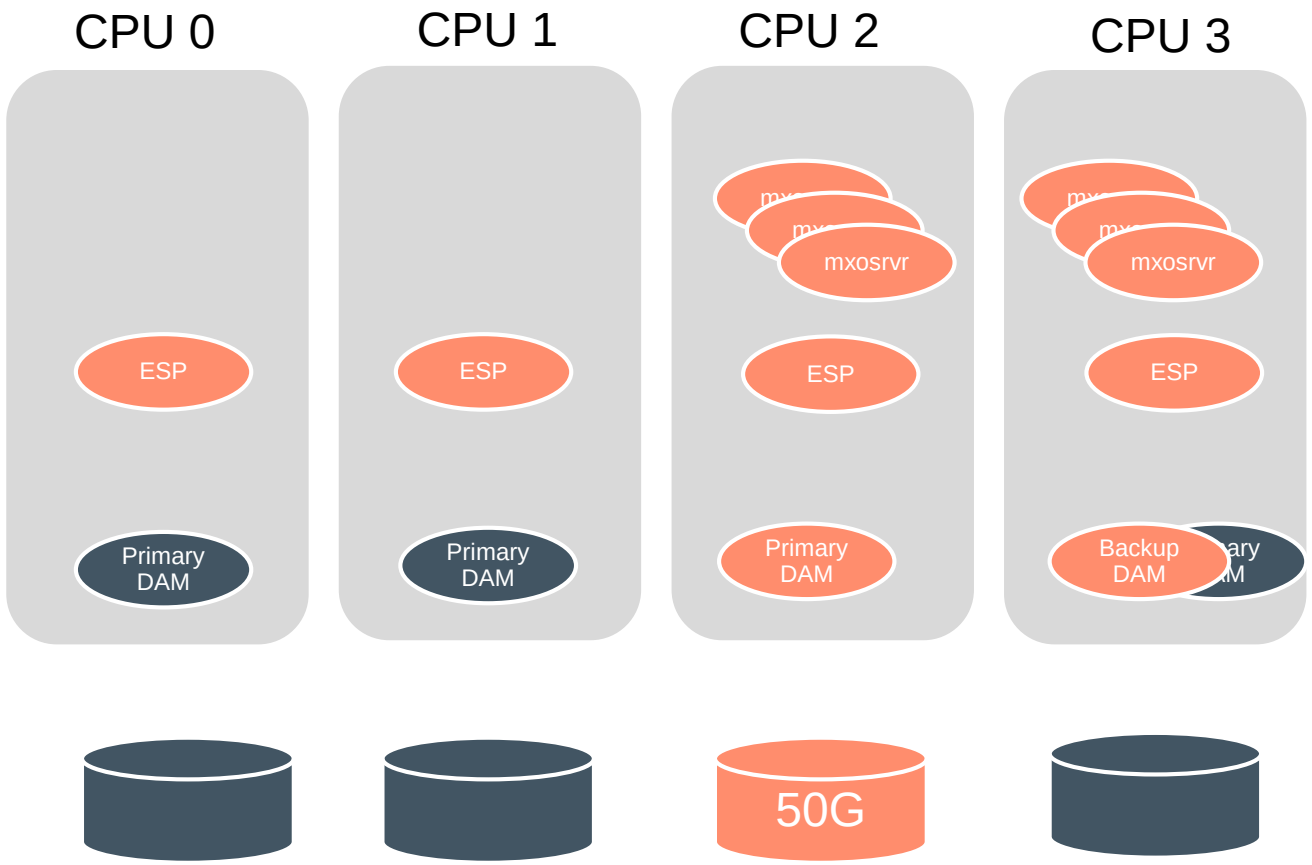
SQL/MX DBS Compute resources

- 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)

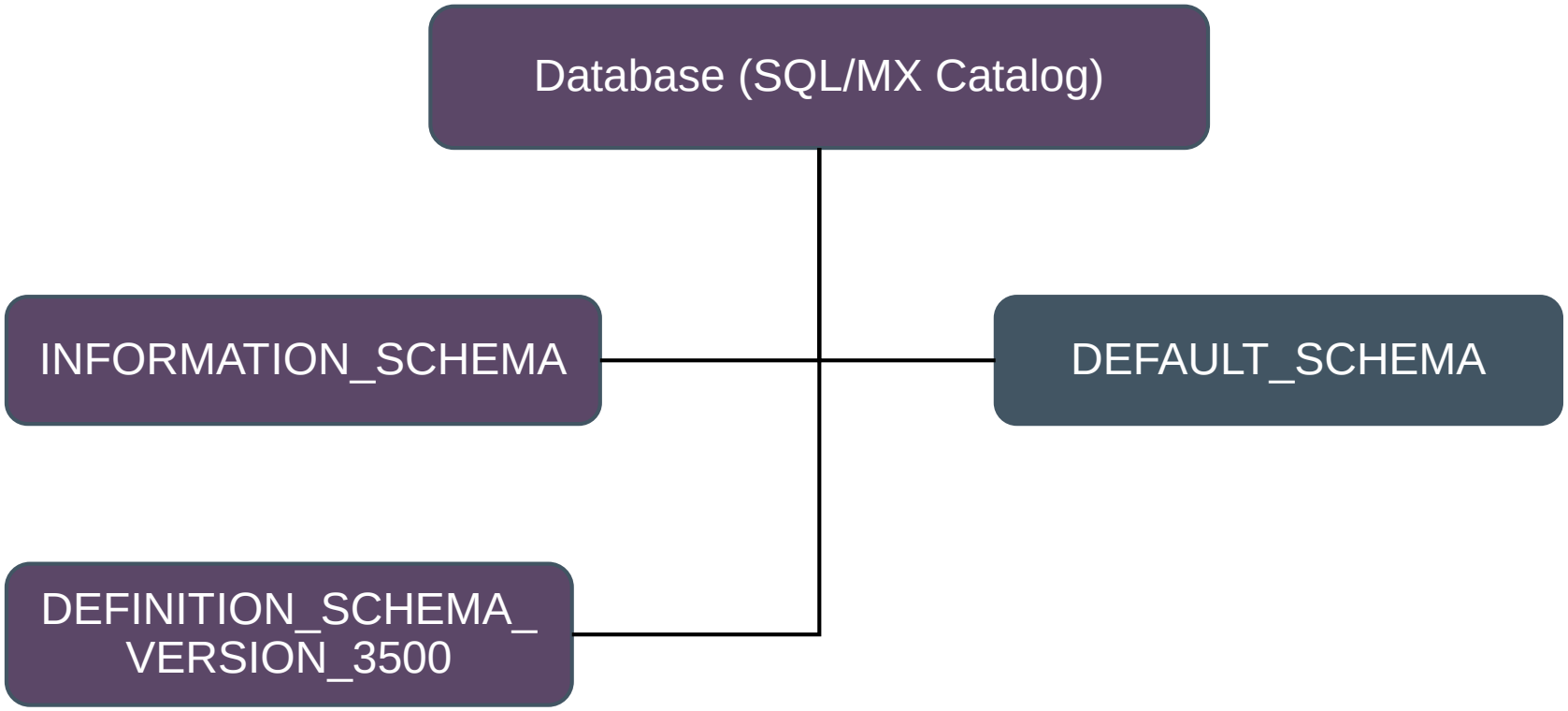


Process overview at-a-glance (2)



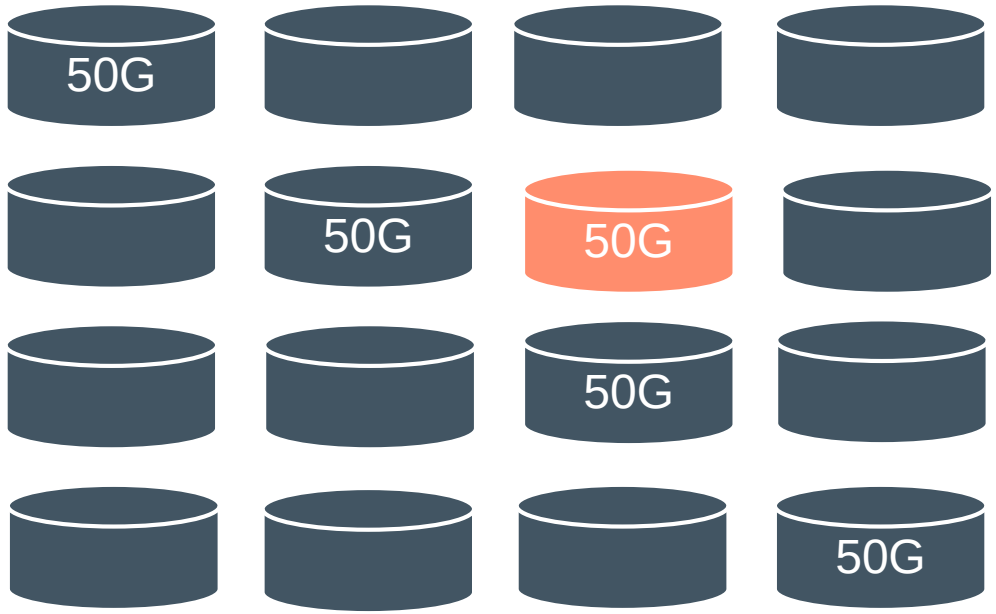
Using the database

Initial schemas in SQL/MX DBS



User uses database (1)

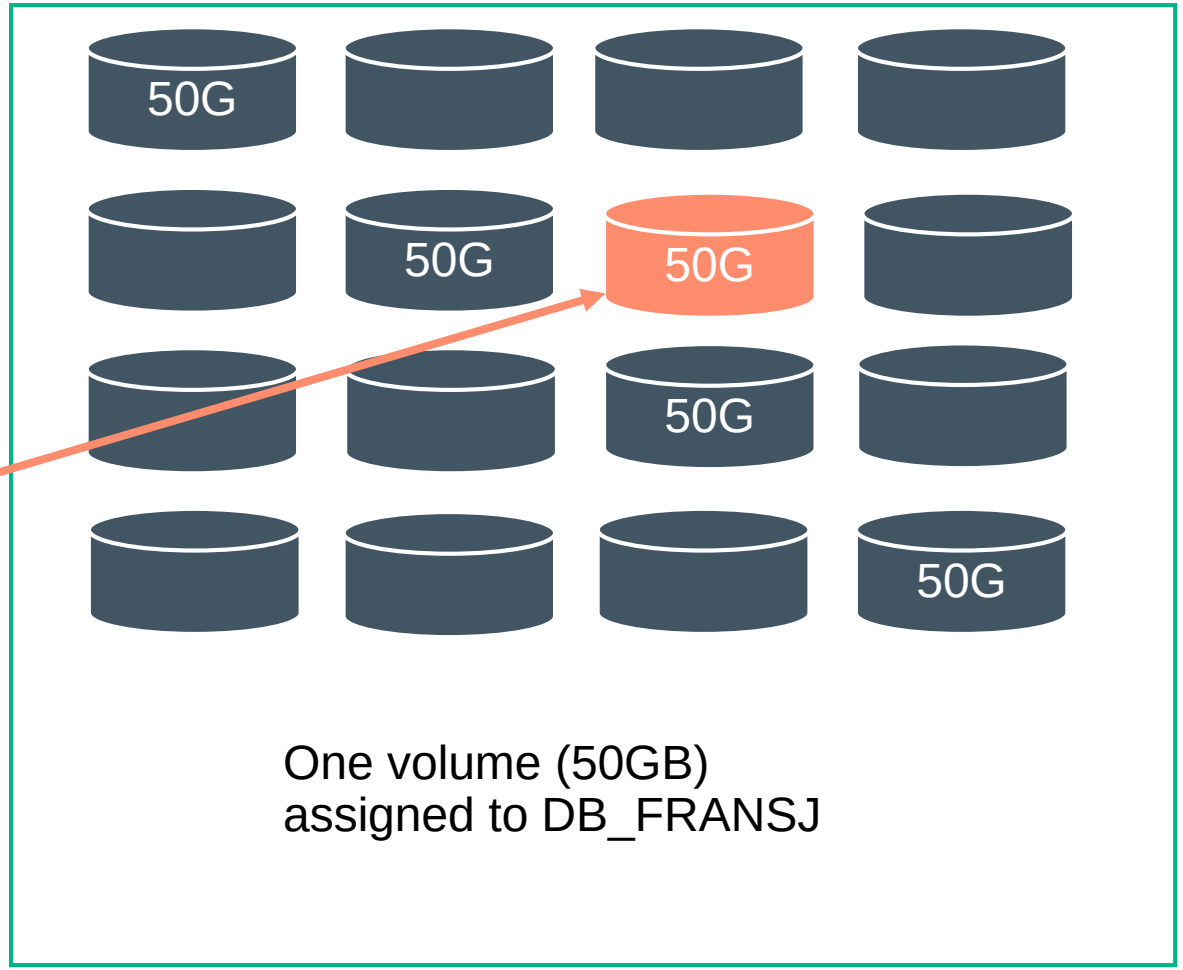
```
~> rmxci -h nsk-nsx09:2100 -dsn DB_FRANSJ -u
frans@hpe.com -p welcome
Welcome to the NonStop(TM) SQL/MX Remote
Conversational Interface
(c) Copyright 2015-2016 Hewlett Packard Enterprise
Development Company, LP


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.
```

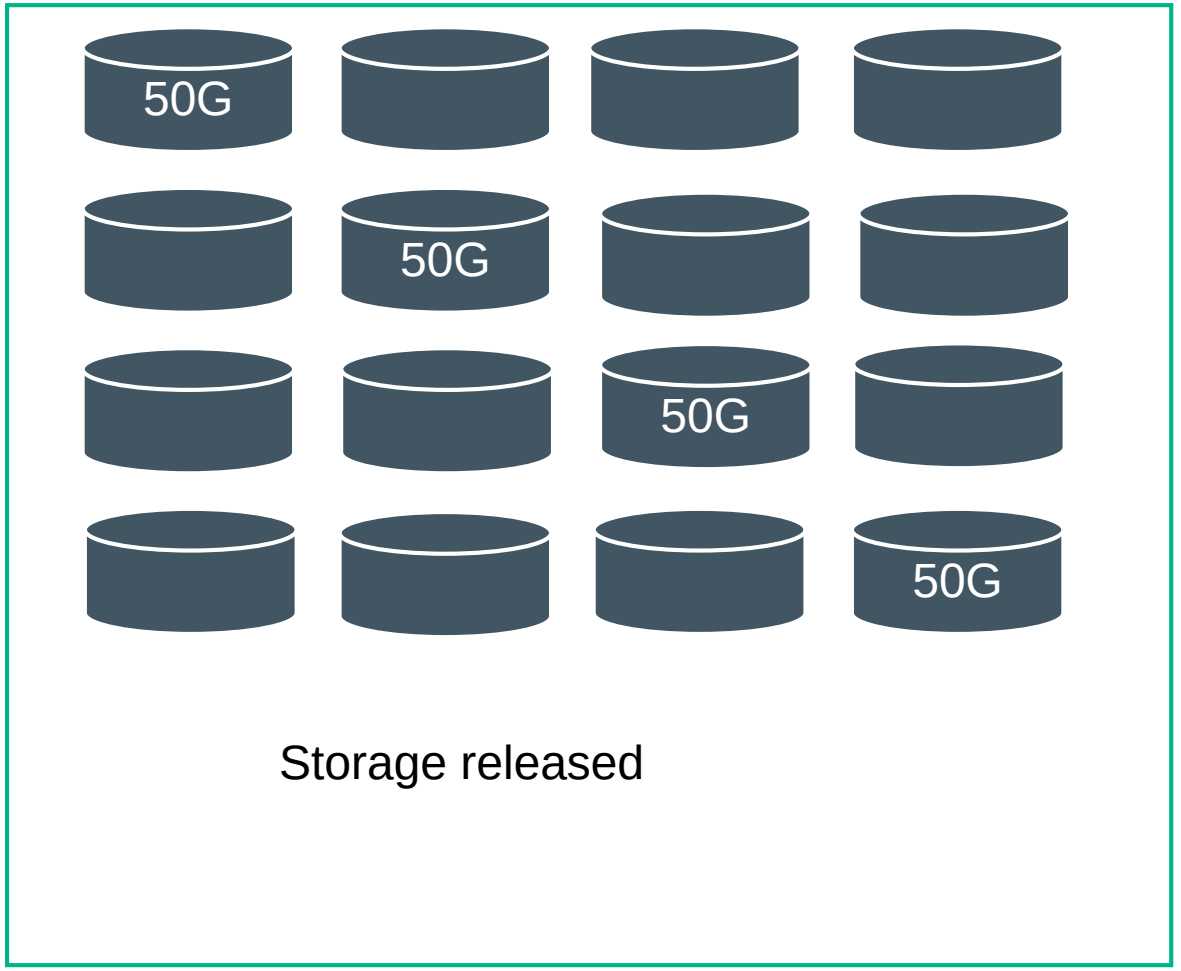


Hewlett Packard Enterprise

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

Behind the scenes

DBS administration and installation

Frans Jongma, NonStop Advanced Technology Center

Agenda

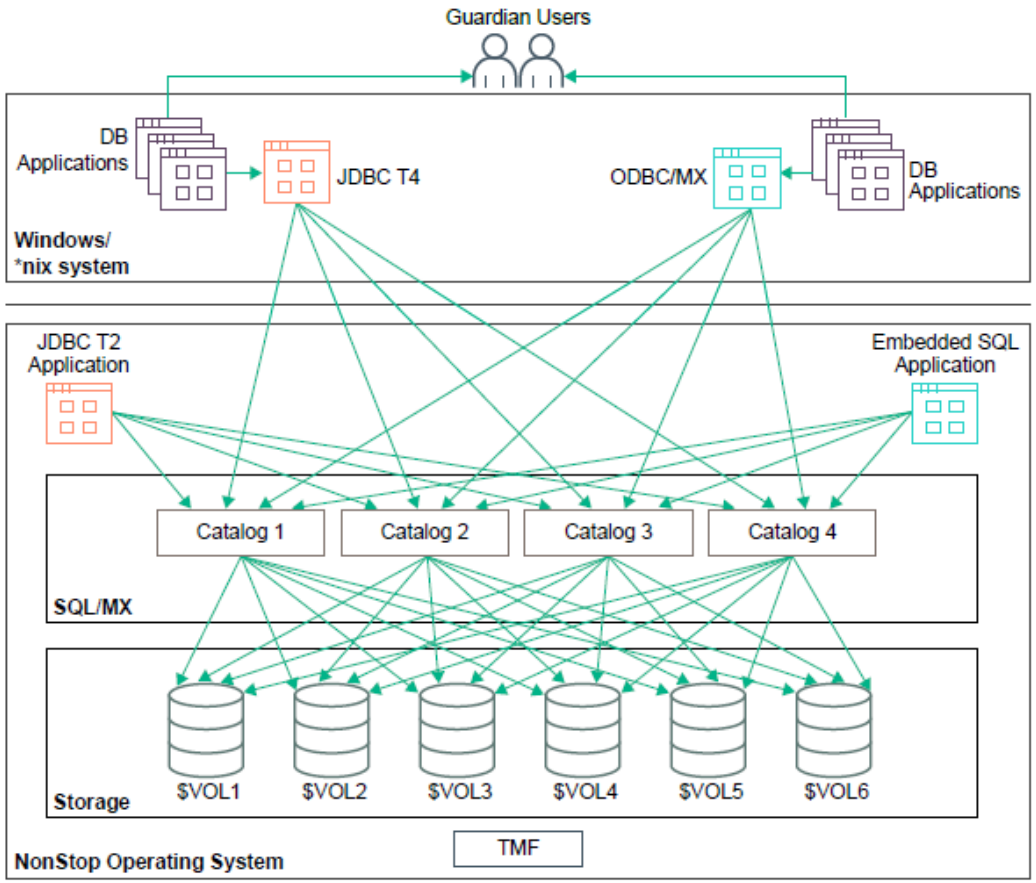
- How is DBS different from SQL/MX as you know it?
- Administration of database services
- DBS resource administration
  - Defined by InstallDBS
- DBS database administration
  - Maintained by DBS management API mxdbs
- DBS Installation
- Post-installation suggestions

Differences-at-a-glace

Between traditional SQL/MX and DBS

What makes MXDBS different?

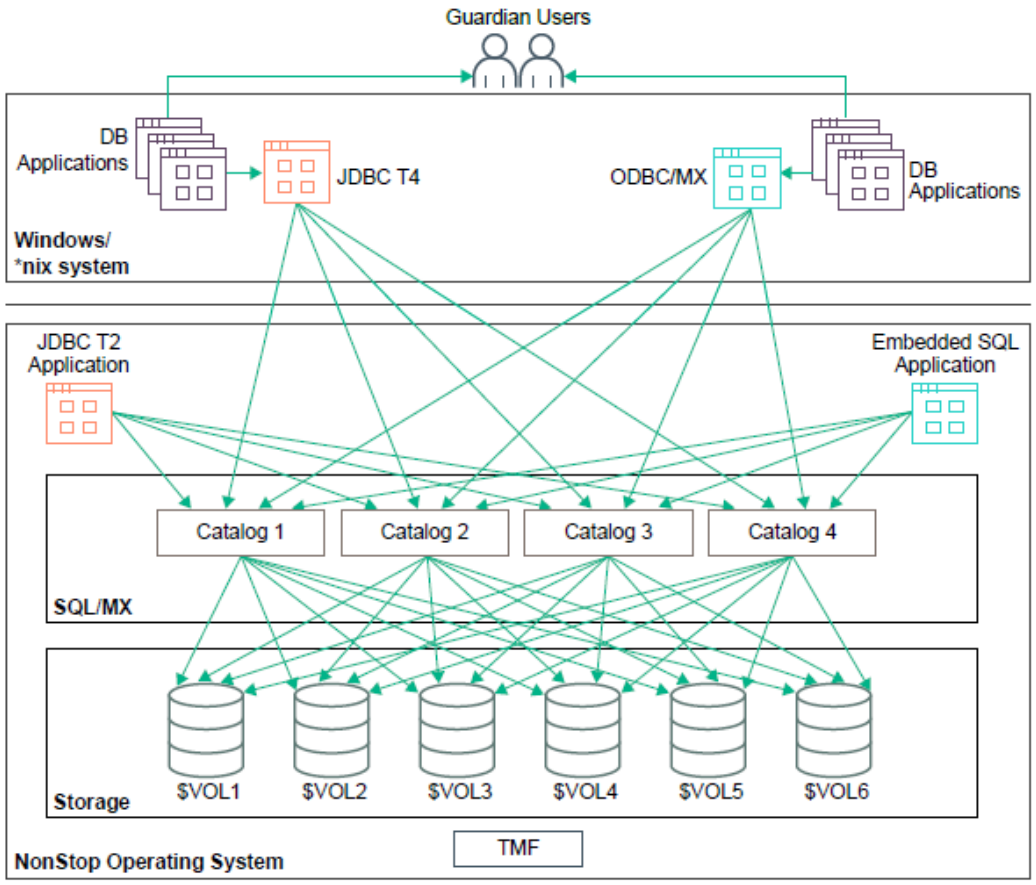
Overview of traditional SQL/MX access

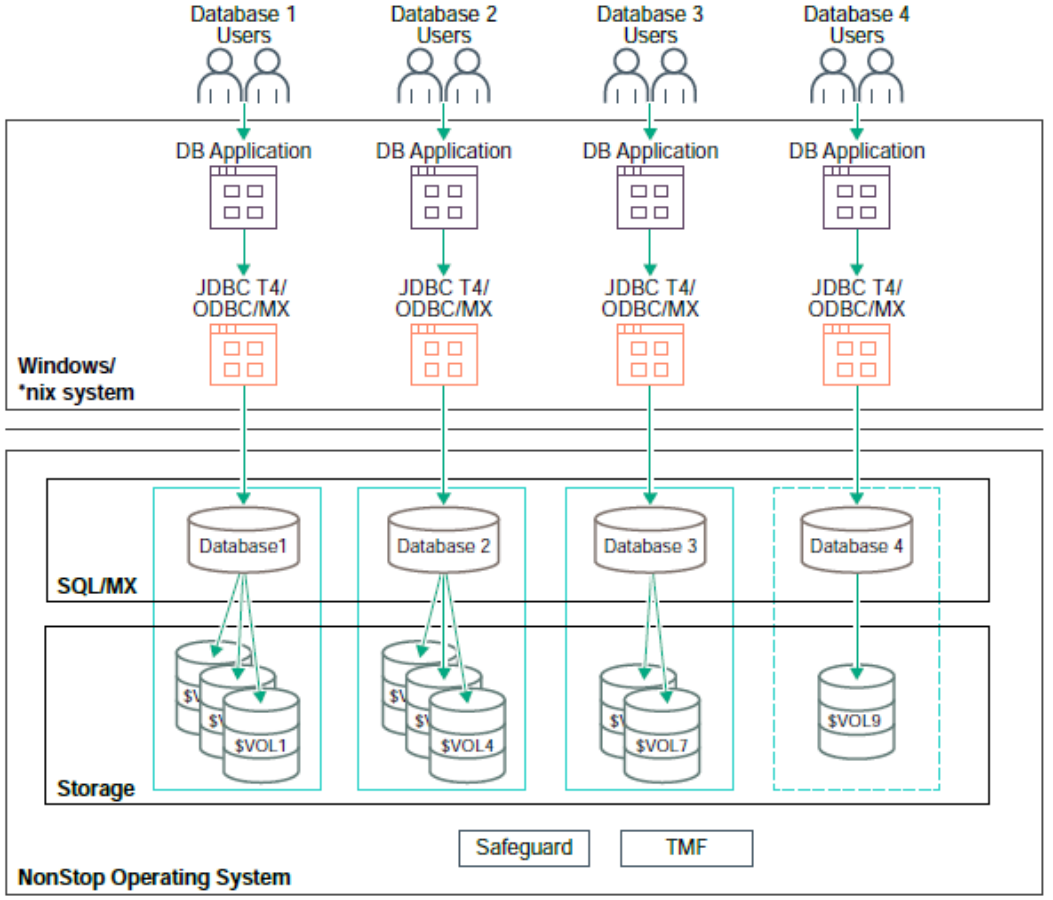


Source: SQL/MX Database Services Manual.

What makes MXDBS different?

Comparing to SQL/MX with DBS





Hewlett Packard Enterprise

intel

Administration of Database Services

- SQL/MX DBS is all about sharing resources
- And makes sure not to share too much
  - Isolate environments that need to be isolated
  - Have more protection enforced than usual on a NonStop system
  - Automation rather than manual DBA actions
- DBS administration includes
  - Database users – coordinated with Safeguard
  - DBS-specific user information
  - User privileges
    - Including privilege groups
  - Storage that can be used
  - CPUs that can be used
  - Databases defined in DBS

User and database isolation

Feature	SQL/MX	SQL/MX DBS
User creation	Manual via TACL/Safecom	Automatic
User access to shell or TACL	Yes	No
Read system metadata	Yes	No, only access via views in the INFORMATION_SCHEMA
Read other user metadata	Yes	No
Access other user's data	If granted	If granted
Access other database data	n/a (use grant/revoke)	If database shared and if granted
Reference data in other catalog in DDL	Yes	No
Access to storage	All volumes unless Safeguard restricted	Only to assigned volumes

Administration of resources and databases

SQL/MX DBS Resources

Defined when installDBS is executed

- Defines the TENANT group and TENANT.ADMIN user
- The OSS home directory for DBS data
- Safeguard file-sharing groups
- Safeguard user-groups for database users
- Which TMF audited volumes are used exclusively
  - These volumes will be Safeguard-owned by the TENANT.ADMIN user
  - No other users can access these volumes
  - Once assigned to a database, only the DB users can access these volumes
- MXCS Services and ports
  - Persistent process DBS_MGMT_MXOAS ($ZAS01, port 2000)
  - Persistent process DBS_ACCESS_MXOAS ($ZAS02, port 2100)
- TENANT.ADMIN password

SQL/MX DBS Resources in system metadata

- Catalog NONSTOP_SQLMX_*node*
  - version 3500 or higher
- Schema SYSTEM_DBS_SCHEMA
- Tables
  - DBS_CPUs — CPUs in the system that can be used by DBS
  - DBS_GLOBALS — Global information (Guardian group and user ID for tenant admin)
  - DBS_PLATFORM_USERS — List of users created by installDBS
  - DBS_SERVICES — The two services defined for DBS
  - DBS_SFG_GROUPS — Safeguard groups
  - DBS_VOLUMES — Volumes allocated for DBS use

SQL/MX DBS Administration

Maintained by the functions of mxdbs

- Functions select resources from the DBS resource pool and assign them to specific databases
- Function: **db-create**
  - Storage (volumes) , CPU assignments, assigns external names to internal user IDs
  - Use Safeguard to assign volume protection to database users
  - Creates data source definitions and store them in metadata
- Function: **db-delete**
  - Releases volumes back to the DBS storage pool
  - Removes data source definitions
  - Return Safeguard ownership to tenant administrator
- Function: **db-add-user**, **db-remove-user**, **db-user-change-access**
  - Maintain membership of users to privilege-groups

SQL/MX DBS administration in system metadata

- Catalog NONSTOP_SQLMX_*node*
- Schema SYSTEM_DBS_SCHEMA and SYSTEM_SECURITY_SCHEMA
- Tables
  - ALL_DATABASES
  - DATABASE_CPUS Which CPUs are used by this DB (based on volumes)
  - DATABASE_DS Data source name and Database UID
  - DATABASE_PRIVILEGE_GROUPS
  - DATABASE VOLUMES Volumes assigned to the database

  - Tables in SYSTEM_SECURITY_SCHEMA
    - DATABASE_USERS Users defined on the system at time of InstallDBS and DBS reserved user IDs
    - DATABASE_USERS_EXT External user names referring to NonStop user IDs

SQL/MX DBS information in each database

System metadata made available using views

- Catalog — Every SQL/MX DBS database catalog
- Schema — INFORMATION_SCHEMA
- Views
  - DB_CPUS — Which CPUs are used by this DB (based on volumes)
  - DB_DS — Data source name and Database UID
  - DB_PRIVILEGE_GROUPS
  - DB_SCHEMAS — Schemas defined in this database
  - DB_STORAGE — Volumes assigned to the database
  - DB_USERS — Database users

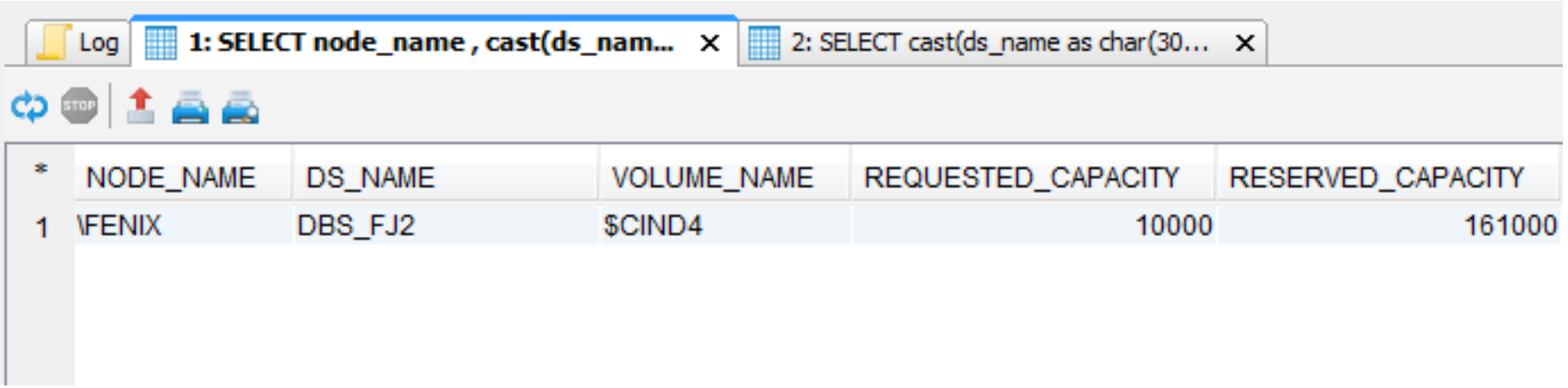
Some useful system metadata queries

Examples to be executed by DBAs

Example metadata queries

View DBS databases and their volumes

```
SET SCHEMA SYSTEM_DBS_SCHEMA; -- Note: need to set catalog to your node
SELECT node_name
, cast(ds_name as char(30)) ds_name
, volume_name
, requested_capacity
, reserved_capacity FROM DATABASE_DS
NATURAL JOIN                DATABASE_VOLUMES
ORDER BY 1,2,3;
```

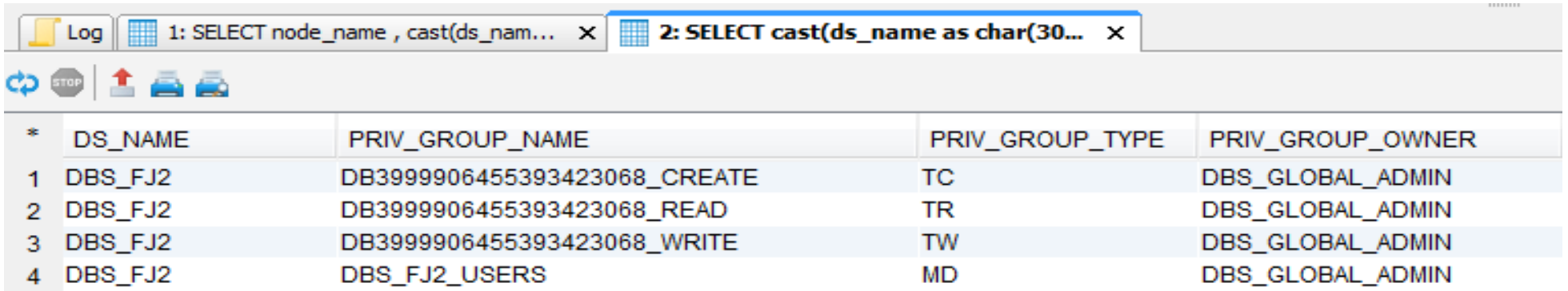


*	NODE_NAME	DS_NAME	VOLUME_NAME	REQUESTED_CAPACITY	RESERVED_CAPACITY
1	\FENIX	DBS_FJ2	$CIND4	10000	161000

Example metadata queries

View databases and their privilege groups

```
SET SCHEMA SYSTEM_DBS_SCHEMA; -- Note: need to set catalog to your node
SELECT cast(ds_name as char(30)) ds_name
, CAST (priv_group_name AS CHAR(50)) priv_group_name
, priv_group_type
, CAST(EXT.EXTERNAL_USER_NAME AS CHAR(30)) priv_group_owner
FROM          DATABASE_DS
NATURAL JOIN DATABASE_PRIVILEGE_GROUPS
NATURAL JOIN SYSTEM_SECURITY_SCHEMA.PRIVILEGE_GROUPS PG
JOIN SYSTEM_SECURITY_SCHEMA.DATABASE_USERS_EXT EXT ON PG.PRIV_GROUP_OWNER = EXT.USERID
ORDER BY 1,2;
```

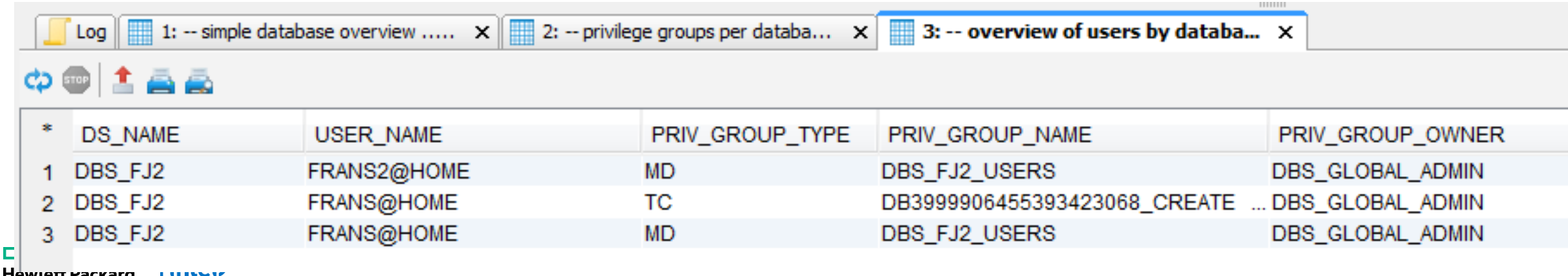


*	DS_NAME	PRIV_GROUP_NAME	PRIV_GROUP_TYPE	PRIV_GROUP_OWNER
1	DBS_FJ2	DB39999064553934230 68_CREATE	TC	DBS_GLOBAL_ADMIN
2	DBS_FJ2	DB39999064553934230 68_READ	TR	DBS_GLOBAL_ADMIN
3	DBS_FJ2	DB39999064553934230 68_WRITE	TW	DBS_GLOBAL_ADMIN
4	DBS_FJ2	DBS_FJ2_USERS	MD	DBS_GLOBAL_ADMIN

Example metadata queries

View databases and their users

```
SET SCHEMA SYSTEM_DBS_SCHEMA; -- Note: need to set catalog to your node
SELECT CAST(DS_NAME AS CHAR(30)) DS_NAME
, CAST(EXT2.EXTERNAL_USER_NAME AS CHAR(40)) USER_NAME
, DB_PG.PRIV_GROUP_TYPE
, CAST (PG.PRIV_GROUP_NAME AS CHAR(50)) PRIV_GROUP_NAME
, CAST(EXT.EXTERNAL_USER_NAME AS CHAR(40)) PRIV_GROUP_OWNER FROM DATABASE_DS DB
JOIN DATABASE_PRIVILEGE_GROUPS DB_PG                        ON DB.DB_UID = DB_PG.DB_UID
JOIN SYSTEM_SECURITY_SCHEMA.PRIVILEGE_GROUPS PG             ON DB_PG.PRIV_GROUP_UID = PG.PRIV_GROUP_UID
JOIN SYSTEM_SECURITY_SCHEMA.DATABASE_USERS_EXT EXT          ON PG.PRIV_GROUP_OWNER = EXT.USERID
JOIN SYSTEM_SECURITY_SCHEMA.PRIVILEGE_GROUP_MEMBERSHIP PGM  ON PGM.PRIV_GROUP_UID = PG.PRIV_GROUP_UID
JOIN SYSTEM_SECURITY_SCHEMA.DATABASE_USERS_EXT EXT2         ON PGM.USERID = EXT2.USERID
ORDER BY 1,2;
```



*	DS_NAME	USER_NAME	PRIV_GROUP_TYPE	PRIV_GROUP_NAME	PRIV_GROUP_OWNER
1	DBS_FJ2	FRANS2@HOME	MD	DBS_FJ2_USERS	DBS_GLOBAL_ADMIN
2	DBS_FJ2	FRANS@HOME	TC	DB39999064553934230 68_CREATE ...	DBS_GLOBAL_ADMIN
3	DBS_FJ2	FRANS@HOME	MD	DBS_FJ2_USERS	DBS_GLOBAL_ADMIN

DBS Installation

Prepare to install

– System metadata must be version 3500 or higher

– Define the general configuration parameters
  – Store these in a file called GeneralConfix.txt

– Define the volumes to be assigned to DBS
  – Store these in a file called VolumeConfig.txt

– Define the TENANT.ADMIN password

– Note that you do not have to specify TMF configuration details
  – Audit trail and Auxiliary Audit trails are part of TMF and is not specific to DBS
  – Consider assigning an AUX volume per processor

– See the examples in the SQLMX Database Services Manual

General Configuration

- Use a sample GeneralConfig.txt as the basis
- The configuration uses shell variables to set the environment
  - Example: export DATABASE_OSS_HOME="/mxdbsdata"
- Note the escaped $ in \$ZAS01
- Do not define too many file-sharing groups to reduce installDBS execution time

```
~> cat /usr/tandem/sqlmx/dbs/GeneralConfig.txt
export DATABASE_ADMIN_GROUP=100
export DATABASE_ADMIN_USER=TENANT.ADMIN
export DATABASE_OSS_HOME="/mxdbsdata"
export DATABASE_FSGROUP_PREFIX=DBS_FS_GROUP
export DATABASE_USER_GROUP_PREFIX=DBS
export DATABASE_MGMT_PORT=2000
export DATABASE_MGMT_PORT_RANGE=60
export DATABASE_MGMT_GENERIC_PROCESS=DBS_MGMT_MXOAS
export DATABASE_MGMT_PROCESS_NAME=\$ZAS01
export DATABASE_MGMT_DS_NAME=DBS_MGMT_DS
export DATABASE_ACCESS_PORT=2100
export DATABASE_ACCESS_GENERIC_PROCESS=DBS_ACCESS_MXOAS
export DATABASE_ACCESS_PROCESS_NAME=\$ZAS02
export DATABASE_FILE_SHARING_GROUPS=1001-1020
export DATABASE_USER_GROUPS=101-120
export DATABASE_USER_INITIAL_PW=
export DATABASE_ACCESS_PORT_RANGE=60000
export DATABASE_MGMT_HOST_NAME=FENIX
export DATABASE_ACCESS_HOST_NAME=FENIX
export DATABASE_CPUS="0 1"
```

Volume Configuration

- See example VolumeConfig.txt
- Easy to create from tmfcom
  - Info datavols as basis
  - Keep the volumes you want to assign to DBS

```
~> cat /usr/tandem/sqlmx/dbs/VolumeConfig.txt
  $CIND3    Mat    Online    Started
  $CIND4    Mat    Online    Started
  $CIND5    Mat    Online    Started
  $CIND6    Mat    Online    Started
```

Installation

- First run the validation
- Need SUPER.SUPER to validate and install
- `# cd /usr/tandem/sqlmx/dbs`
- `# ../bin/installDBS -validateonly -vols VolumeConfig.txt -config GeneralConfig.txt`
- When no errors are reported run the installation
- `# ../bin/installDBS -vols VolumeConfig.txt -config GeneralConfig.txt -adminPW xyzzy`
- Create a database to verify
- `# ../bin/mxdbs db-create dbs_test 10 test_user Welcome-1234`

- Remember to use a different password than listed here

Post install

- Create a small test database to verify all is working
  - As super.super
  - Or sudo as shown in the example
- Create a download area and symbolic links to client software
  - Makes is easier to download and install software for clients
  - Example: mxcreatelinks script
- Setup ssh for easy access to mxdbs from client servers or workstations

```
/usr/tandem/sqlmx> sudo bin/mxdbs db-create
dbs_test 10 test_user Welcome-1234
Hewlett Packard Enterprise NonStop(TM) SQL/MX
DBS Client 3.5.1
(c) Copyright 2016, 2017 Hewlett Packard
Enterprise Development LP.

db-create command started.

MXCS Service Host    : FENIX
MXCS Service Port    : 2100
Datasource Name      : DBS_TEST
Initial Schema Name  : DEFAULT_SCHEMA
OSS Directory        : DB1002

---  mxdbs operation complete.

/usr/tandem/sqlmx> sudo bin/mxdbs db-delete
dbs_test
```

Post installation

Setup ssh for tenant.admin

Setup ssh for mxdbs (client side)

- SSH can be used to allow running remote commands without entering passwords
- Requires public keys to be exchanged between client / host
- Here is an example

Running on Cygwin:

```
~> ssh-keygen -t rsa -b 2048 -C "frans@nsx09"
Generating public/private rsa key pair.
Enter file in which to save the key (/cygdrive/c/frans/.ssh/id_rsa):
Created directory '/cygdrive/c/frans/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /cygdrive/c/frans/.ssh/id_rsa.
Your public key has been saved in /cygdrive/c/frans/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:3lZaB9ct9nX0LpEEGbD80OrtpbmUOCx6uuToYdqb3IE frans@nsx09
```

And create the public key as follows

```
~> ssh-keygen -E md5 -lf /cygdrive/c/frans/.ssh/id_rsa.pub
2048 MD5:8e:73:f3:47:8c:c3:d5:46:eb:68:3d:29:11:63:83:c1 frans@nsx09
(RSA)

~> cat .ssh/config
Host nsk-nsx09
     User frans
Host mx-nsx09  (I use this to access the system for DBS)
     Hostname nsk-nsx09
     User tenant.admin
```

Setup ssh for mxdbs (server side)

- Need to add myself to the ssh configuration of the target system
- With this key, I can access the system as user frans and as user TENANT.ADMIN without having to enter a password
- Ideal for automated processes
- Here is an example

Running on NonStop (NSX09)

```
~> sudo gtacl -p sshcom \$zss0
SSHCOM T0801L02_20JAN2017_ACC - 2017-04-14 07:25:14.773
OPEN $zss0
% mode daemon
mode daemon
OK, switched to daemon mode
% alter user frans , publickey fjongma4 fingerprint
8e:73:f3:47:8c:c3:d5:46:eb:68:3d:29:11:63:83:c1
alter user frans , publickey fjongma4 fingerprint
8e:73:f3:47:8c:c3:d5:46:eb:68:3d:29:11:63:83:c1
OK, user frans altered
% alter user tenant.admin  , publickey fjongma4 fingerprint
8e:73:f3:47:8c:c3:d5:46:eb:68:3d:29:11:63:83:c1
alter user tenant.admin  , publickey fjongma4 fingerprint
8e:73:f3:47:8c:c3:d5:46:eb:68:3d:29:11:63:83:c1
OK, user tenant.admin altered
info user frans, detail

USER                 KEYS SYSTEM-USER       LAST-MODIFIED LAST-LOGON
STATUS
frans                   1 frans             14Apr17,07:17 14Apr17,04:08
THAWED
```

Post installation

mxcreatelinks

Purpose of mxcreatelinks

Creates OSS symbolic links to installation files in Guardian

```
/usr/tandem/sqlmx/downloads> ls -g
total 6
lrwxrwxrwx 1 SUPER 26 Dec 17 17:59 mxdmWin32ex.zip       -> /G/system/zmxodbc/mxdm32ex
lrwxrwxrwx 1 SUPER 26 Dec 17 17:59 mxdmWin64ex.zip       -> /G/system/zmxodbc/mxdm64ex
lrwxrwxrwx 1 SUPER 26 Dec 17 17:59 odbcHPUX64.tar        -> /G/system/zmxodbc/odbc64hi
lrwxrwxrwx 1 SUPER 26 Dec 17 17:59 odbcLinux32.tar       -> /G/system/zmxodbc/lodbctar
lrwxrwxrwx 1 SUPER 25 Dec 17 17:59 odbcLinux64.tar       -> /G/system/zmxodbc/lodbc64
lrwxrwxrwx 1 SUPER 25 Dec 17 17:59 odbcWin32.exe         -> /G/system/zmxodbc/tdmodbc
lrwxrwxrwx 1 SUPER 25 Dec 17 17:59 odbcWin32_unicode.exe -> /G/system/zmxodbc/odbcw32
lrwxrwxrwx 1 SUPER 26 Dec 17 17:59 odbcWin64.exe         -> /G/system/zmxodbc/nsodbc64
lrwxrwxrwx 1 SUPER 25 Dec 17 17:59 odbcWin64_unicode.exe -> /G/system/zmxodbc/odbcw64
lrwxrwxrwx 1 SUPER 26 Dec 17 17:59 rmxci.zip             -> /G/system/zmxodbc/t0774zip
lrwxrwxrwx 1 SUPER 26 Dec 17 17:59 t4DriverSoftware.tar  -> /G/system/zmxodbc/t1249tar
lrwxrwxrwx 1 SUPER 24 Dec 17 17:59 vqpWin32.exe          -> /G/system/zmxtools/mxvqp
```

Other thoughts

– Consider a separate EMS collector for the DBS data sources
  – Prior to running InstallDBS, change the script to add specific EMS collectors for DBS MXCS.
  – Release 3.6 allows MXOAS as process pairs.

Summary

- Reviewed the differences between DBS and common SQL/MX
- DBS uses new metadata to keep track of its users
  - External user IDs map to Guardian user IDs
  - The use of privilege groups
- How to configure resources for DBS installation
- Some tips

Hewlett Packard
Enterprise

Thank you

Frans.Jongma@hpe.com