Neo4J Tools

Neo4j Admin tool

Introduction

- For managing your Neo4j instance
- Command-line tool

Syntax and commands

Command	Description
check-consistency	Check the consistency of a database
report	Collect the most common information needed for remote assessments
store-info	Print information about a Neo4j database store
memrec	Print Neo4j heap and pagecache memory settings recommendations.
import	Import from a collection of CSV files
copy	Copy data from an existing database to a new database
set-initial-password	Sets the initial password of the initial admin user (neo4j)
dump	Dump a database into a single-file archive.
load	Load a database from an archive created with the dump command.
backup	Perform an online backup from a running Neo4j server.
restore	Restore a backed-up database.
unbind	Removes cluster state data from a stopped Neo4j server.

Environment variables

NEO4J_DEBUG	Set to anything to enable debug output.
NEO4J_HOME	Neo4j home directory.
NEO4J_CONF	Path to directory which contains neo4j.conf.
HEAP_SIZE	Set JVM maximum heap size during command execution. Takes a number and a unit, for example 512m.
JAVA_OPTS	Additional JVM arguments.

Consistency checker

Introduction

- To check consistency of a database or the backup

- Examples:

  - echo "START DATABASE neo4j" | cypher-shell -u neo4j -p secret -d system

  - neo4j-admin check-consistency --database=neo4j

  - neo4j-admin check-consistency --backup backup/neo4j-backup

Report tool

Introduction

- To simplify the support process by collecting the relevant information in a standard way

- To share this information with the Neo4j Support organization, you will have to send it manually.

Command

- neo4j-admin report

Option	Default	Description
--to	reports/	Specifies to target directory where the report should be written to.
--list		Will list available classifiers.
--verbose		Will instruct the tool to print more verbose output.
--force		Will disable the available disk space check.
--pid		Specify process id of running Neo4j instance. Only applicable when used together with classifiers indicated as *Online* in the table below.

Display store information

Introduction

- The store-info command outputs the version of a database store format

```
$neo4j-home> /bin/neo4j-admin store-info data/databases/graph.db
Store format version:        SF4.0.0
Store format introduced in:   4.0.0
```

- Run the store-info command to see if the store format of the backup is compatible with running Neo4j instance

```
$neo4j-home> bin/neo4j-admin store-info /tmp/3518/mygraph.db
Store format version:        vE.H.4 \ ①
Store format introduced in:  3.4.0 \ ②
Store format superseded in:  4.0.0 \ ③
```

neo4j-admin store-info /data/databases/neo4j

neo4j-admin store-info /tmp/3518/mygraph.db

Memory recommendations

Introduction

- Gives heuristic memory setting recommendations
- The heuristic is based on
  - Total memory of the system or
  - The amount of memory specified with the --memory argument
- Examples:
  - neo4j-admin memrec --memory=16g

```
$neo4j-home> bin/neo4j-admin memrec --memory=16g

...

...

...

# Based on the above, the following memory settings are recommended:
dbms.memory.heap.initial_size=5g
dbms.memory.heap.max_size=5g
dbms.memory.pagecache.size=7g
```

Import

Introduction

- For batch imports of large amounts of data into a Neo4j database from CSV files
- Can only be used to load data into a previously unused database

Syntax

- neo4j-admin import --nodes import/movies_header.csv,import/movies.csv \
- --nodes import/actors_header.csv,import/actors.csv \
- --relationships import/roles_header.csv,import/roles.csv

CSV File header format

- Specifies how the data fields should be interpreted
- Contains information for each field, with the format
  - <name>:<field_type>

Properties

```
:ID,name,joined:date,active:boolean,points:int
user01,Joe Soap,2017-05-05,true,10
user02,Jane Doe,2017-08-21,true,15
user03,Moe Know,2018-02-17,false,7
```

Node files

- Contains node data

- Can have
  - ID field
  - LABEL field as well as
  - Properties

```
tt0133093,"The Matrix",1999,Movie
tt0234215,"The Matrix Reloaded",2003,Movie;Sequel
tt0242653,"The Matrix Revolutions",2003,Movie;Sequel
```

Relationship files

- Three mandatory fields:

- TYPE
  - The relationship type to use for this relationship.

- START_ID
  - The ID of the start node for this relationship.

- END_ID
  - The ID of the end node for this relationship.

Relationship files

- The name of character that the actor is playing in this movie is stored as a role property on the relationship.

```
keanu,"Neo",tt0133093,ACTED_IN
keanu,"Neo",tt0234215,ACTED_IN
keanu,"Neo",tt0242653,ACTED_IN
laurence,"Morpheus",tt0133093,ACTED_IN
laurence,"Morpheus",tt0234215,ACTED_IN
laurence,"Morpheus",tt0242653,ACTED_IN
carrieanne,"Trinity",tt0133093,ACTED_IN
carrieanne,"Trinity",tt0234215,ACTED_IN
carrieanne,"Trinity",tt0242653,ACTED_IN
```

Dump and load databases

Introduction

- A Neo4j database can be dumped and loaded using the following commands:
  - neo4j-admin dump --database=<database> --to=<destination-path>
  - neo4j-admin load --from=<archive-path> --database=<database> [--force]

Limitations

- Database must be stopped
- Database must be created subsequently using CREATE DATABASE

Usage

- neo4j-admin dump --database=neo4j --to=/backups/neo4j/2016-10-02.dump

- ls /backups/neo4j

- neo4j-admin load --from=/backups/neo4j/2016-10-02.dump --database=neo4j --force

- If using the load command to seed a Causal Cluster, must first perform **neo4j-admin unbind** on each of the cluster instances

Unbind a Core Server

Introduction

- To remove cluster state of Causal Cluster member

- Examples of usage
  - To turn a Causal Cluster server into a standalone server
  - When seeding a Causal Cluster with existing store files
  - When recovering a Causal Cluster

- Command
  - neo4j-admin unbind

Copy a database

Introduction

- Used to copy data from an existing database to a new database
- The existing database must be stopped before copying from it
- The destination database must not yet exist
- Example:
  - STOP DATABASE neo4j
  - neo4j-admin copy --from-database=neo4j --to-database=copy-neo4j
  - CREATE DATABASE copy-neo4j
  - START DATABASE neo4j

Cypher Shell

Introduction

- To run queries and perform administrative tasks against a Neo4j instance
- Example:
  - $neo4j-home> bin/cypher-shell -u neo4j -p <password>

Examples

- neo4j> :help

- neo4j> MATCH (n) RETURN n;

Using cat (UNIX)

- $neo4j-home> cat examples.cypher | bin/cypher-shell -u neo4j -p <password> --format plain

Transactions

- Supports explicit transactions
- Transaction states are controlled using the keywords
  - :begin
  - :commit
  - :rollback
- neo4j> :begin
- neo4j> CREATE (:Person {name : 'Edward Mygma', alias : 'The Riddler' });
- neo4j> :commit