

An operational intro to Neo4j



What are we going to do?



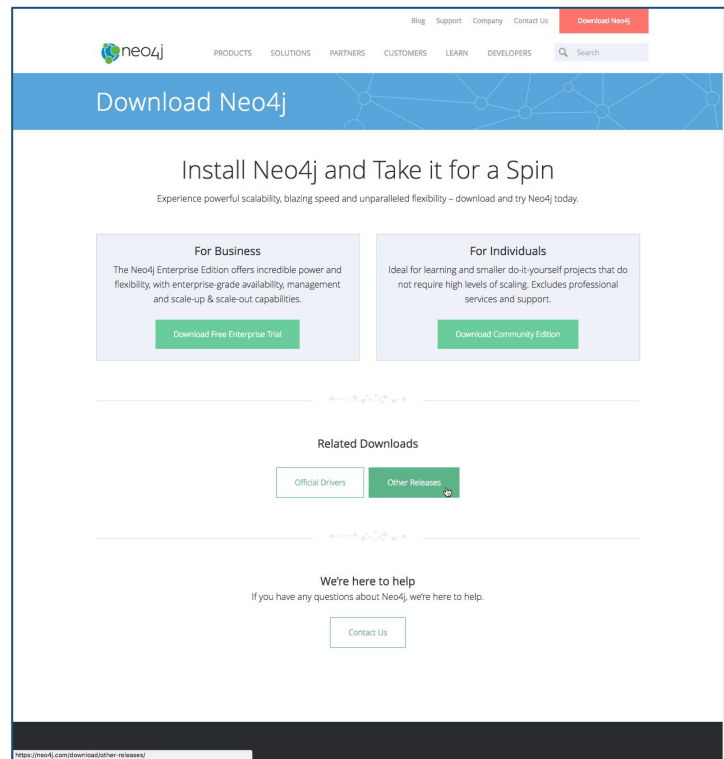
- Download and install Neo4j
- Overview of what Neo4j consists of
- Launch Neo4j and import a demo dataset
- Intro to Neo4j's command line tools



Obtain Neo4j Enterprise Edition



- Get Neo4j:
 - Copy Neo4j from your USB stick onto your machine, or
 - download it from <http://neo4j.com/download>
- Unpack Neo4j
 - Double-click, or
 - `tar -xf neo4j-enterprise-3.1.4-unix.tar.gz`

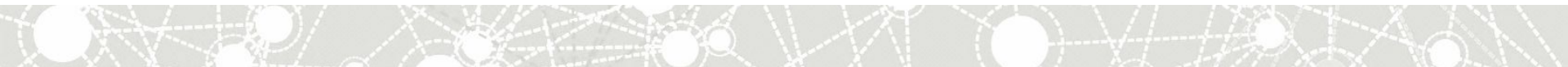


Obtaining Neo4j



There are several other ways to obtain Neo4j:

- Debian package
- Docker image
- Neo4j Desktop (for development only)



This is where command line tools live.

```
bin/  
├── cypher-shell  
├── neo4j  
└── neo4j-admin
```

Updating the default password



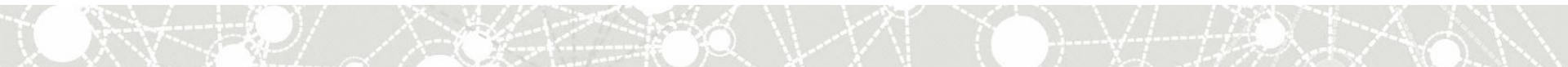
We need to update the default password on first use. Run the following command to do this:

```
./bin/neo4j-admin set-initial-password gcctraining
```

Script to:

- start
- console
- stop
- restart
- view the status of

the database.



Run the following command from your Neo4j home directory:

```
./bin/neo4j start
```

Starting Neo4j.

Started neo4j (pid 3844). By default, it is available at <http://localhost:7474/>

There may be a short delay until the server is ready.

See logs/neo4j.log for current status.

Run the following command from your Neo4j home directory:

```
./bin/neo4j start
```

Active database: graph.db

Directories in use:

```
home:      /var/lib/neo4j
config:    /var/lib/neo4j/conf
logs:      /var/lib/neo4j/logs
plugins:    /var/lib/neo4j/plugins
import:     /var/lib/neo4j/import
data:      /var/lib/neo4j/data
certificates: /var/lib/neo4j/certificates
run:       /var/lib/neo4j/run
```

Starting Neo4j.

```
2018-08-28 14:24:28.174+0000 INFO  ===== Neo4j 3.4.6 =====
2018-08-28 14:24:28.211+0000 INFO  Starting...
2018-08-28 14:24:30.298+0000 INFO  Initiating metrics...
2018-08-28 14:24:31.066+0000 INFO  Sending metrics to CSV file at
/var/lib/neo4j/metrics
2018-08-28 14:24:31.284+0000 INFO  Bolt enabled on 127.0.0.1:7687.
2018-08-28 14:24:33.070+0000 INFO  Started.
2018-08-28 14:24:33.278+0000 INFO  Mounted REST API at: /db/manage
2018-08-28 14:24:33.329+0000 INFO  Server thread metrics has been
registered successfully
2018-08-28 14:24:33.965+0000 INFO  Remote interface available at
http://localhost:7474/
```

Watch the startup process



Have a look at the log file to see what's going on while the database is starting up.

```
tail -f logs/neo4j.log
```



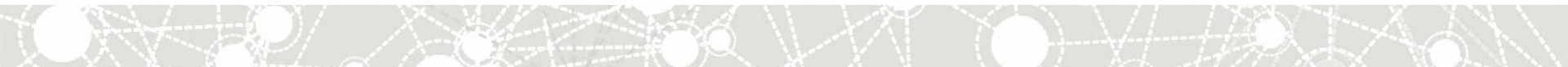
Check if the database is up and running



We can also call the status or console commands to view the status of the database:

```
./bin/neo4j console  
Neo4j is already running (pid 86209).
```

```
./bin/neo4j status  
Neo4j is already running (pid 86209).
```



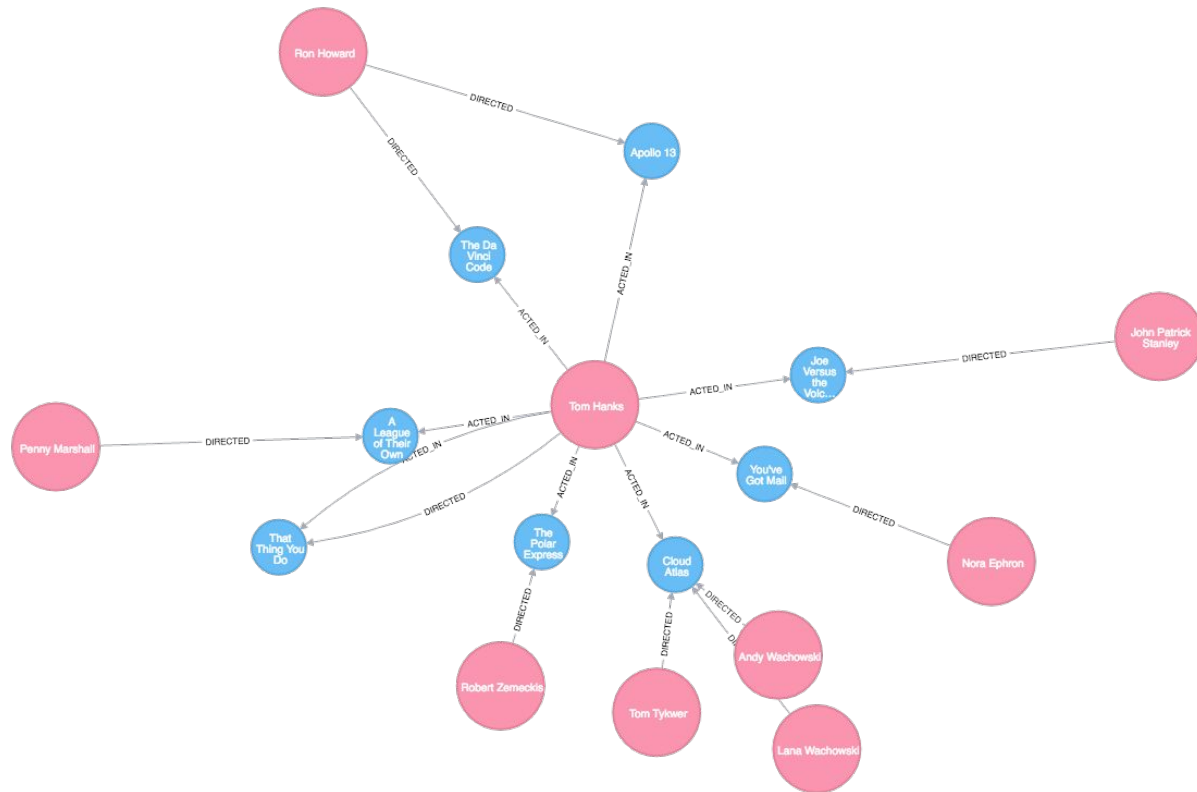
:play movies



Navigate to <http://localhost:7474/> and type the following command into the query bar:

```
:play movies
```

:play movies



Neo4j Browser 101



- Run queries with CMD (CTRL) + Enter
- Insert new-line with SHIFT + Enter
- Expand the query bar with ESC
- CMD (CTRL) + Up Arrow to scroll through past queries
- Use `:clear` to clear past results
- Double-click a node and see what happens!
- Change node colors
- Change which node property is displayed

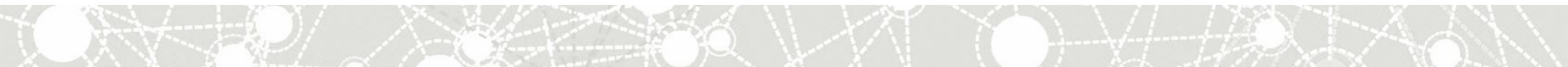


:play sysinfo



Type the following command into the query bar to get an overview of the state of the server:

`:sysinfo`



:play sysinfo

Store Sizes ?

Array Store	8.00 KiB
Logical Log	2.68 KiB
Node Store	8.00 KiB
Property Store	7.97 KiB
Relationship Store	0 B
String Store Size	8.00 KiB
Total Store Size	133.33 KiB

ID Allocation ?

Node ID	1
Property ID	1
Relationship ID	0
Relationship Type ID	0

Page Cache ?

Faults	33	Flushes	0
Evictions	0	Eviction Exceptions	0
File Mappings	44	File Unmappings	28
Bytes Read	213263	Bytes Written	0

Transactions ?

Last Tx Id	20
Current	0
Peak	3
Opened	184
Committed	155

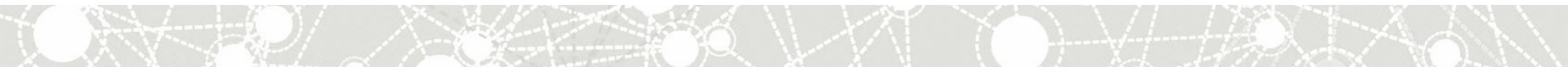
High Availability ?

Not enabled.

Cluster ?

No cluster.

Note: some metrics only available in Neo4j Enterprise. AUTO-REFRESH ☐ OFF ↺



bin/cypher-shell



A command line shell where you can execute Cypher against an instance of Neo4j. Interacts with the database using the Bolt Protocol.

Replaces `neo4j-shell`.



Run the following command from your Neo4j home directory:

```
./bin/cypher-shell -u neo4j -p <password>
```

Connected to Neo4j at bolt://localhost:7687 as user neo4j.

Type :help for a list of available commands.

```
neo4j>
```

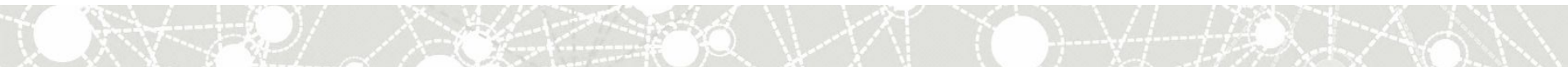


Trying out bin/cypher-shell



Type the following command into cypher-shell:

```
CALL db.labels();
```



Viewing all the procedures



You can see a list of all the available procedures by running the following procedure:

```
CALL dbms.procedures();
```

Try out some of the other procedures to get the hang of what they do.



Deploying procedures



We can also deploy our own custom procedures into the plugins directory.

The Neo4j community are maintaining a library of useful procedures in the apoc project:

<https://github.com/neo4j-contrib/neo4j-apoc-procedures/releases>



Deploying procedures

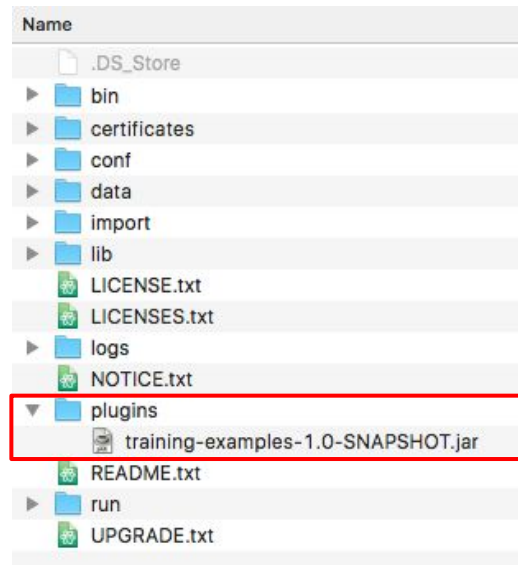


Copy the plugins folders
from **USB Stick**
to the **default.graphdb**
folder

(or `$NEO4J_HOME`)

Then restart Neo4j:

```
./bin/neo4j restart
```



Viewing all the procedures



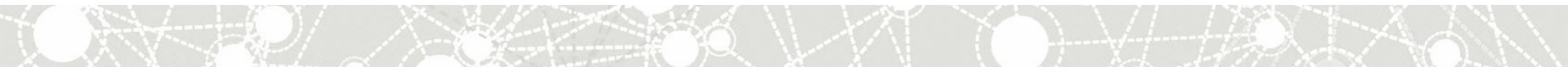
Now run the procedure again:

```
CALL dbms.procedures();
```

\$ call dbms.procedures()			    				
Rows	dbms.security.changePassword	dbms.security.changePassword(password :: STRING?, requirePasswordChange = false :: BOOLEAN?) :: VOID	Change the current user's password.				
	dbms.security.changeUserPassword	dbms.security.changeUserPassword(username :: STRING?, newPassword :: STRING?, requirePasswordChange = true :: BOOLEAN?) :: VOID	Change the given user's password.				
Text	dbms.security.clearAuthCache	dbms.security.clearAuthCache() :: VOID	Clears authentication and authorization cache.				
	dbms.security.createRole	dbms.security.createRole(roleName :: STRING?) :: VOID	Create a new role.				
Code	dbms.security.createUser	dbms.security.createUser(username :: STRING?, password :: STRING?, requirePasswordChange = true :: BOOLEAN?) :: VOID	Create a new user.				
	dbms.security.deleteRole	dbms.security.deleteRole(roleName :: STRING?) :: VOID	Delete the specified role. Any role assignments will be removed.				
	dbms.security.deleteUser	dbms.security.deleteUser(username :: STRING?) :: VOID	Delete the specified user.				
	dbms.security.listRoles	dbms.security.listRoles() :: (role :: STRING?, users :: LIST? OF STRING?)	List all available roles.				
	dbms.security.listRolesForUser	dbms.security.listRolesForUser(username :: STRING?) :: (value :: STRING?)	List all roles assigned to the specified user.				
	dbms.security.listUsers	dbms.security.listUsers() :: (username :: STRING?, roles :: LIST? OF STRING?, flags :: LIST? OF STRING?)	List all local users.				
	dbms.security.listUsersForRole	dbms.security.listUsersForRole(roleName :: STRING?) :: (value :: STRING?)	List all users currently assigned the specified role.				
	dbms.security.removeRoleFromUser	dbms.security.removeRoleFromUser(roleName :: STRING?, username :: STRING?) :: VOID	Unassign a role from the user.				
	dbms.security.showCurrentUser	dbms.security.showCurrentUser() :: (username :: STRING?, roles :: LIST? OF STRING?, flags :: LIST? OF STRING?)	Show the current user.				
	dbms.security.suspendUser	dbms.security.suspendUser(username :: STRING?) :: VOID	Suspend the specified user.				
	dbms.setTXMetaData	dbms.setTXMetaData(data :: MAP?) :: VOID	(empty)				
	training.moviesOnly	training.moviesOnly(name :: STRING?) :: (title :: STRING?)	Find movies by an actor				

Returned 35 records in 92 ms.

This is where configuration files live.



This is where configuration files live.

conf/

- |— jmx.access
- |— jmx.password
- └— neo4j.conf



The name of the database to mount

```
#dbms.active_database=graph.db
```

Whether requests to Neo4j are authenticated.

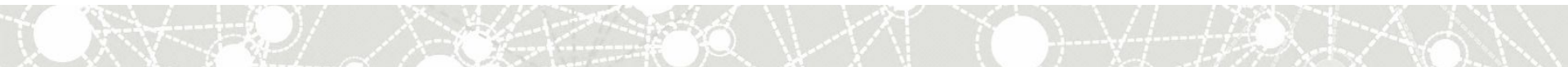
To disable authentication, uncomment this line

```
#dbms.security.auth_enabled=false
```

```
#####
```

Core/Edge Configuration

```
#####
```



Change HTTP port



HTTP Connector

`dbms.connector.http.type=HTTP`

`dbms.connector.http.enabled=true`

#dbms.connector.http.listen_address=:7474



Change HTTP port

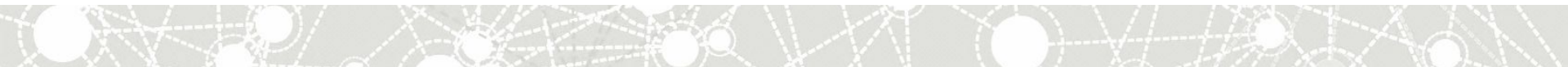


HTTP Connector

```
dbms.connector.http.type=HTTP
```

```
dbms.connector.http.enabled=true
```

```
dbms.connector.http.listen_address=:8484
```



Restart Neo4j



Run the following command:

```
./bin/neo4j restart
```

Navigate to <http://localhost:8484> in your browser



data



This is where the data lives.

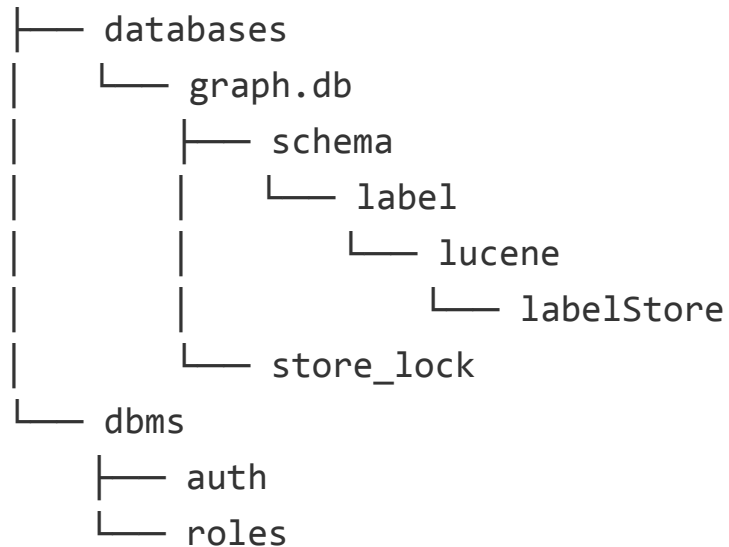


This is where the data lives.

```
data/
├── databases
│   ├── graph.db
│   │   ├── schema
│   │   │   ├── label
│   │   │   │   ├── lucene
│   │   │   │   └── labelStore
│   │   └── store_lock
│   └── dbms
│       ├── auth
│       └── roles
```

This is where the data lives.

data/



`dbms.directories.data=data`

This is where the data lives.

```
data/
├── databases
│   └── graph.db
│       ├── schema
│       │   └── label
│       │       └── lucene
│       │           └── labelStore
│       └── store_lock
└── dbms
    ├── auth
    └── roles
```

`dbms.active_database=graph.db`

data/databases/graph.db



neostore
neostore.counts.db.a
neostore.counts.db.b
neostore.id
neostore.labeltokenstore.db
neostore.labeltokenstore.db.id
neostore.labeltokenstore.db.names
neostore.labeltokenstore.db.names.id
neostore.nodestore.db
neostore.nodestore.db.id
neostore.nodestore.db.labels
neostore.nodestore.db.labels.id
neostore.propertystore.db
neostore.propertystore.db.arrays
neostore.propertystore.db.arrays.id
neostore.propertystore.db.id
neostore.propertystore.db.index
neostore.propertystore.db.index.id

neostore.propertystore.db.index.keys
neostore.propertystore.db.index.keys.id
neostore.propertystore.db.strings
neostore.propertystore.db.strings.id
neostore.relationshipgroupstore.db
neostore.relationshipgroupstore.db.id
neostore.relationshipstore.db
neostore.relationshipstore.db.id
neostore.relationshiptypestore.db
neostore.relationshiptypestore.db.id
neostore.relationshiptypestore.db.names
neostore.relationshiptypestore.db.names.id
neostore.schemastore.db
neostore.schemastore.db.id
neostore.transaction.db.0
schema
store_lock

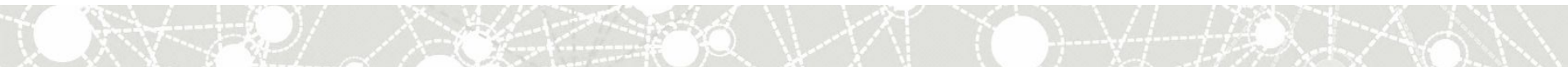


Transaction log



neostore
neostore.counts.db.a
neostore.counts.db.b
neostore.id
neostore.labeltokenstore.db
neostore.labeltokenstore.db.id
neostore.labeltokenstore.db.names
neostore.labeltokenstore.db.names.id
neostore.nodestore.db
neostore.nodestore.db.id
neostore.nodestore.db.labels
neostore.nodestore.db.labels.id
neostore.propertystore.db
neostore.propertystore.db.arrays
neostore.propertystore.db.arrays.id
neostore.propertystore.db.id
neostore.propertystore.db.index
neostore.propertystore.db.index.id

neostore.propertystore.db.index.keys
neostore.propertystore.db.index.keys.id
neostore.propertystore.db.strings
neostore.propertystore.db.strings.id
neostore.relationshipgroupstore.db
neostore.relationshipgroupstore.db.id
neostore.relationshipstore.db
neostore.relationshipstore.db.id
neostore.relationshiptypestore.db
neostore.relationshiptypestore.db.id
neostore.relationshiptypestore.db.names
neostore.relationshiptypestore.db.names.id
neostore.schemastore.db
neostore.schemastore.db.id
neostore.transaction.db.0
schema
store_lock



Store files



neostore

neostore.counts.db.a

neostore.counts.db.b

neostore.id

neostore.labeltokenstore.db

neostore.labeltokenstore.db.id

neostore.labeltokenstore.db.names

neostore.labeltokenstore.db.names.id

neostore.nodestore.db

neostore.nodestore.db.id

neostore.nodestore.db.labels

neostore.nodestore.db.labels.id

neostore.propertystore.db

neostore.propertystore.db.arrays

neostore.propertystore.db.arrays.id

neostore.propertystore.db.id

neostore.propertystore.db.index

neostore.propertystore.db.index.id

neostore.propertystore.db.index.keys

neostore.propertystore.db.index.keys.id

neostore.propertystore.db.strings

neostore.propertystore.db.strings.id

neostore.relationshipgroupstore.db

neostore.relationshipgroupstore.db.id

neostore.relationshipstore.db

neostore.relationshipstore.db.id

neostore.relationshiptypestore.db

neostore.relationshiptypestore.db.id

neostore.relationshiptypestore.db.names

neostore.relationshiptypestore.db.names.id

neostore.schemastore.db

neostore.schemastore.db.id

neostore.transaction.db.0

schema

store_lock

Metadata file



neostore

neostore.counts.db.a
neostore.counts.db.b
neostore.id
neostore.labeltokenstore.db
neostore.labeltokenstore.db.id
neostore.labeltokenstore.db.names
neostore.labeltokenstore.db.names.id
neostore.nodestore.db
neostore.nodestore.db.id
neostore.nodestore.db.labels
neostore.nodestore.db.labels.id
neostore.propertystore.db
neostore.propertystore.db.arrays
neostore.propertystore.db.arrays.id
neostore.propertystore.db.id
neostore.propertystore.db.index
neostore.propertystore.db.index.id

neostore.propertystore.db.index.keys
neostore.propertystore.db.index.keys.id
neostore.propertystore.db.strings
neostore.propertystore.db.strings.id
neostore.relationshipgroupstore.db
neostore.relationshipgroupstore.db.id
neostore.relationshipstore.db
neostore.relationshipstore.db.id
neostore.relationshiptypestore.db
neostore.relationshiptypestore.db.id
neostore.relationshiptypestore.db.names
neostore.relationshiptypestore.db.names.id
neostore.schemastore.db
neostore.schemastore.db.id
neostore.transaction.db.0
schema
store_lock

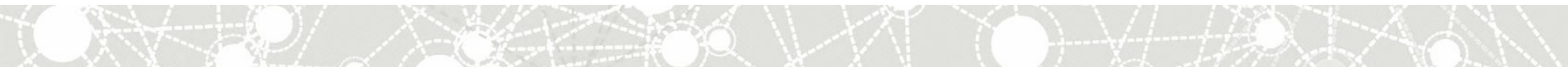
Schema



neostore
neostore.counts.db.a
neostore.counts.db.b
neostore.id
neostore.labeltokenstore.db
neostore.labeltokenstore.db.id
neostore.labeltokenstore.db.names
neostore.labeltokenstore.db.names.id
neostore.nodestore.db
neostore.nodestore.db.id
neostore.nodestore.db.labels
neostore.nodestore.db.labels.id
neostore.propertystore.db
neostore.propertystore.db.arrays
neostore.propertystore.db.arrays.id
neostore.propertystore.db.id
neostore.propertystore.db.index
neostore.propertystore.db.index.id

neostore.propertystore.db.index.keys
neostore.propertystore.db.index.keys.id
neostore.propertystore.db.strings
neostore.propertystore.db.strings.id
neostore.relationshipgroupstore.db
neostore.relationshipgroupstore.db.id
neostore.relationshipstore.db
neostore.relationshipstore.db.id
neostore.relationshiptypestore.db
neostore.relationshiptypestore.db.id
neostore.relationshiptypestore.db.names
neostore.relationshiptypestore.db.names.id
neostore.schemastore.db
neostore.schemastore.db.id
neostore.transaction.db.0

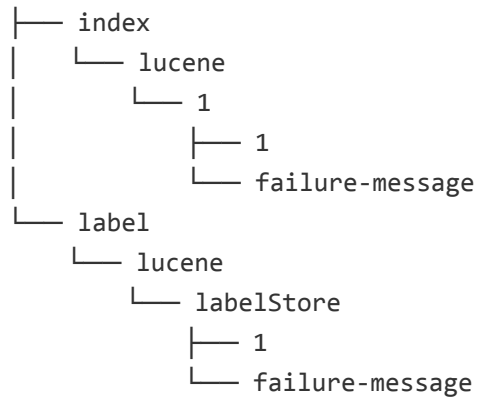
schema
store_lock



Schema



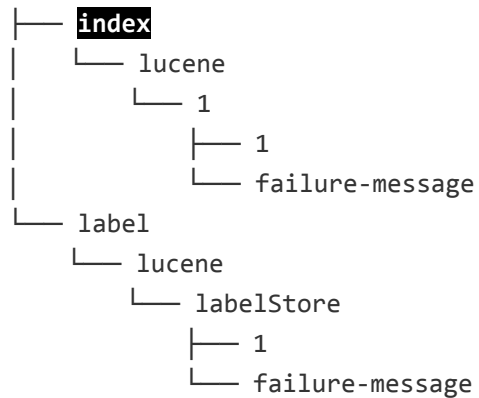
data/databases/graph.db/schema/



Indexes and Unique Constraints



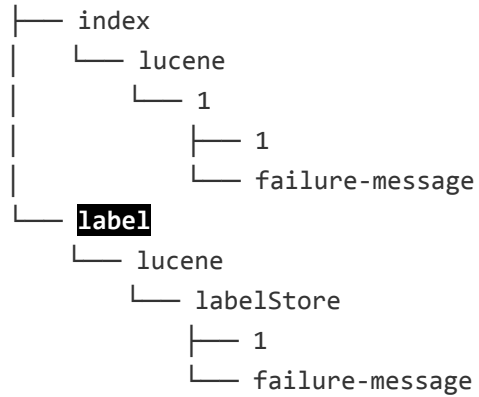
data/databases/graph.db/schema/



Label scan store



data/databases/graph.db/schema/



logs



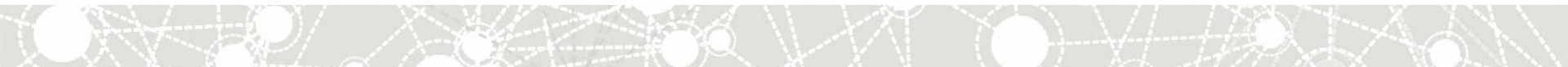
This is where log files live.



This is where log files live.

`neo4j.log` The standard log, where general information about Neo4j is written.

`debug.log` Information useful when debugging problems with Neo4j.



```
2016-09-14 09:04:14.185+0000 INFO Starting...
2016-09-14 09:04:14.793+0000 INFO Bolt enabled on localhost:7687.
2016-09-14 09:04:14.802+0000 INFO Initiating metrics...
2016-09-14 09:04:16.386+0000 INFO Started.
2016-09-14 09:04:16.508+0000 INFO Mounted REST API at: /db/manage
2016-09-14 09:04:17.013+0000 INFO Remote interface available at
http://localhost:7474/
```

logs/debug.log



2016-09-14 10:15:16.155+0000 INFO [o.n.k.i.t.l.c.CheckPointerImpl] Check Pointing triggered by scheduler for time threshold [19]: Starting check pointing...

2016-09-14 10:15:16.157+0000 INFO [o.n.k.i.t.l.c.CheckPointerImpl] Check Pointing triggered by scheduler for time threshold [19]: Starting store flush...

2016-09-14 10:15:16.158+0000 INFO [o.n.k.i.s.c.CountsTracker] About to rotate counts store at transaction 19 to [/tmp/neo4j-enterprise-3.1.0/data/databases/graph.db/neostore.counts.db.b], from [/tmp/neo4j-enterprise-3.1.0/data/databases/graph.db/neostore.counts.db.a].

2016-09-14 10:15:16.159+0000 INFO [o.n.k.i.s.c.CountsTracker] Successfully rotated counts store at transaction 19 to [/tmp/neo4j-enterprise-3.1.0/data/databases/graph.db/neostore.counts.db.b], from [/tmp/neo4j-enterprise-3.1.0/data/databases/graph.db/neostore.counts.db.a].

2016-09-14 10:15:16.161+0000 INFO [o.n.k.i.t.l.c.CheckPointerImpl] Check Pointing triggered by scheduler for time threshold [19]: Store flush completed

2016-09-14 10:15:16.161+0000 INFO [o.n.k.i.t.l.c.CheckPointerImpl] Check Pointing triggered by scheduler for time threshold [19]: Starting appending check point entry into the tx log...

2016-09-14 10:15:16.162+0000 INFO [o.n.k.i.t.l.c.CheckPointerImpl] Check Pointing triggered by scheduler for time threshold [19]: Appending check point entry into the tx log completed

2016-09-14 10:15:16.162+0000 INFO [o.n.k.i.t.l.c.CheckPointerImpl] Check Pointing triggered by scheduler for time threshold [19]: Check pointing completed



End of Module

An operational intro to Neo4j

Questions ?

