

Neo4j Driver Documentation 1.0.0- SNAPSHOT

Table of Contents

- First Example 1
- Driver 2
 - Configuration 2
 - Construction 3
 - Dependencies 4
 - Lifecycle 4
 - SSL / TLS 6
 - URL Format 6
 - Version Mapping Table..... 6
 - Websockets 6
- Errors 6
- Getting Started 7
 - Download..... 7
 - Setup 8
 - Versioning 8
- Results 8
 - Concepts 9
 - Introspection..... 9
 - Iteration 9
 - Plan + Profile..... 9
 - Reading Records..... 9
 - Utility Methods 9
- Session 10
 - Monitoring..... 10
 - Run Statement 10
 - Session State 11
 - Syncing..... 12
- Transactions 12
 - State..... 12
 - Implicit Transcation 12
 - BEGIN, COMMIT, ROLLBACK 12
 - Nested Transactions 13
- Type Mapping 13

First Example

Example 1. Foo bar example

```
def test_can_run_simple_statement_from_bytes_string(self):
    session = GraphDatabase.driver("bolt://localhost").session()
    count = 0
    for record in session.run(b"RETURN 1 AS n"):
        assert record[0] == 1
        assert record["n"] == 1
        assert record.n == 1
        assert repr(record)
```

```
String text = "MATCH (n) RETURN n";

// when
Statement statement = new Statement( text, NO_PARAMETERS );
```

```
var neo4j = require('neo4j');

var statement = ['MERGE (alice:Person {name:{name_a},age:{age_a}})',
  'MERGE (bob:Person {name:{name_b},age:{age_b}})',
  'CREATE UNIQUE (alice)-[alice_knows_bob:KNOWS]->(bob)',
  'RETURN alice, bob, alice_knows_bob'
];

var params = {
  name_a: 'Alice',
  age_a: 33,
  name_b: 'Bob',
  age_b: 44
};

var driver = neo4j.driver("bolt://localhost");

var streamSession = driver.session();
var streamResult = streamSession.run(statement.join(' '), params);
streamResult.subscribe({
  onNext: function(record) {
    // On receipt of RECORD
    for(var i in record) {
      console.log(i);
      console.log(record[i]);
    }
  }
});
```

```

    }
    }, onCompleted: function() {
        var summary = streamResult.summarize();
        //Print number of nodes created
        console.log('');
        console.log(summary.updateStatistics.nodesCreated());
        streamSession.close();
    }, onError: function(error) {
        console.log(error);
    }
});

var promiseSession = driver.session();
var promiseResult = promiseSession.run(statement.join(' '), params);
promiseResult.then(function(records) {
    records.forEach(function(record) {
        for(var i in record) {
            console.log(i);
            console.log(record[i]);
        }
    });
    var summary = promiseResult.summarize();
    //Print number of nodes created
    console.log('');
    console.log(summary.updateStatistics.nodesCreated());
})
.catch(function(error) {
    console.log(error);
})
.then(function(){
    promiseSession.close();
});

```

Driver

Some may argue that the opposite of Uniform Drivers is Casual Drivers, or even Naked Drivers. This is of course preposterous. The opposite and complement of Uniform Drivers is Uniform Passengers, as should be obvious to anyone upon serious reflection. A valiant case has been made by the Dry Cleaner's guild that, as two negatives cancel out, the true opposite is Iform Drivers. Needless to say, this frivolous play on words was uniformly frowned upon by the rest of the community.

Configuration

1. an arrangement of parts or elements in a particular form, figure, or combination: *the unrepeatable*

configuration of the stars at the moment of your birth | the broad configuration of the economy remains capitalist.

- **Computing** the arrangement or set-up of the hardware and software that make up a computer system. *the PC comes with a removable hard disk drive as part of the standard configuration.*
- **Chemistry** the fixed three-dimensional relationship of the atoms in a molecule, defined by the bonds between them. Compare with CONFORMATION.

2. **Psychology** another term for GESTALT.

Construction

His accent serenades me and presses at the door
Broken promise to be around for my mind war
And my accomplishment's interrogating me
Today's become tomorrow before I wanted it to be
And desperate discussions
The start of the destruction
Our sign, in my mind
I'll be fine
I'll be fine

'Cause I'm under construction everyone
So you'll have to mind the mess
I'm under some construction

I always had to try harder, I never really could keep up
Sitting in the corner with my illness and bad luck
But in this humble place I'm feeling like red wine
And I hope to get better with some time
I'll be fine
With some time
I'll be fine

'Cause I'm under construction everyone
So you'll have to mind the mess
I'm under some construction

Construction
Modification
Motivation
Of construction

And the rituals that soothe and disgust me will be gone
With some time
I'll be fine
With some time
I'll be fine
I'll be fine

I'm under construction everyone
So you'll have to mind the mess
I'm under some construction
I'm under construction everyone
So you'll have to mind the mess
I'm under some construction
Under construction
Construction
Feel better with some construction

— No Doubt, Construction Deconstruction @ Everything In Time

Dependencies

Please import them. Also, include them.

Lifecycle

```
usage: mvn [options] [<goal(s)>] [<phase(s)>]
```

Options:

-am, --also-make

If project list is specified, also build projects required by the list

-amd, --also-make-dependents

If project list is specified, also build projects that depend on projects on the list

<code>-B,--batch-mode</code>	Run in non-interactive (batch) mode
<code>-b,--builder <arg></code>	The id of the build strategy to use.
<code>-C,--strict-checksums</code>	Fail the build if checksums don't match
<code>-c,--lax-checksums</code>	Warn if checksums don't match
<code>-cpu,--check-plugin-updates</code>	Ineffective, only kept for backward compatibility
<code>-D,--define <arg></code>	Define a system property
<code>-e,--errors</code>	Produce execution error messages
<code>-emp,--encrypt-master-password <arg></code>	Encrypt master security password
<code>-ep,--encrypt-password <arg></code>	Encrypt server password
<code>-f,--file <arg></code>	Force the use of an alternate POM file (or directory with pom.xml).
<code>-fae,--fail-at-end</code>	Only fail the build afterwards; allow all non-impacted builds to continue
<code>-ff,--fail-fast</code>	Stop at first failure in reactorized builds
<code>-fn,--fail-never</code>	NEVER fail the build, regardless of project result
<code>-gs,--global-settings <arg></code>	Alternate path for the global settings file
<code>-gt,--global-toolchains <arg></code>	Alternate path for the global toolchains file
<code>-h,--help</code>	Display help information
<code>-l,--log-file <arg></code>	Log file to where all build output will go.
<code>-llr,--legacy-local-repository</code>	Use Maven 2 Legacy Local Repository behaviour, ie no use of <code>_remote.repositories</code> . Can also be activated by using <code>-Dmaven.legacyLocalRepo=true</code>
<code>-N,--non-recursive</code>	Do not recurse into sub-projects
<code>-npr,--no-plugin-registry</code>	Ineffective, only kept for backward compatibility
<code>-npu,--no-plugin-updates</code>	Ineffective, only kept for backward compatibility
<code>-nsu,--no-snapshot-updates</code>	Suppress SNAPSHOT updates
<code>-o,--offline</code>	Work offline
<code>-P,--activate-profiles <arg></code>	Comma-delimited list of profiles to activate
<code>-pl,--projects <arg></code>	Comma-delimited list of specified reactor projects to build instead of all projects. A project can be specified by <code>[groupId]:artifactId</code> or by its relative path.

<code>-q,--quiet</code>	Quiet output - only show errors
<code>-rf,--resume-from <arg></code>	Resume reactor from specified project
<code>-s,--settings <arg></code>	Alternate path for the user settings file
<code>-T,--threads <arg></code>	Thread count, for instance 2.0C where C is core multiplied
<code>-t,--toolchains <arg></code>	Alternate path for the user toolchains file
<code>-U,--update-snapshots</code>	Forces a check for missing releases and updated snapshots on remote repositories
<code>-up,--update-plugins</code>	Ineffective, only kept for backward compatibility
<code>-V,--show-version</code>	Display version information WITHOUT stopping build
<code>-v,--version</code>	Display version information
<code>-X,--debug</code>	Produce execution debug output

SSL / TLS

Is it secret, is it safe?

URL Format

```
$url = trim($url, '!"#$%&\'()*+,-./@:;<=>[\\]^_`{|}~');
```

Version Mapping Table

Let's map all the version tables.

Websockets

Put a websocket in it, will ya

Errors

Some things can go wrong. If that happens, null can be returned. Also, boolean can be returned false. There can also be FAILURE AKA ACK_FAILURE.

Example

```
MATCH (null)-[:merge]->(true)
with null.delete as foreach, `true`.false as null
return 2 + foreach, coalesce(null, 3.1415)
limit 10;
```

With this query, you can get error, even when you have 100 heap.

Getting Started

Ready, set, get, start!

Download

To download the driver do

Example 2. Foo bar example

```
pip install neo4j-driver
```

```
<dependencies>
  <dependency>
    <groupId>org.neo4j.driver</groupId>
    <artifactId>neo4j-java-driver</artifactId>
    <version>1.0-SNAPSHOT</version>
  </dependency>
</dependencies>

<repositories>
  <repository>
    <id>neo4j-snapshot-repository</id>
    <name>Neo4j Maven 2 snapshot repository</name>
    <url>http://m2.neo4j.org/content/repositories/snapshots</url>
    <snapshots>
      <enabled>true</enabled>
    </snapshots>
    <releases>
      <enabled>false</enabled>
    </releases>
  </repository>
</repositories>
```

```
var neo4j = require('lib/neo4j');
```

Setup

Yes, that's a good idea.

Versioning

Please see [the other section about this](#).

Results

Hopefully, if you do it right, you will get results.

Concepts

'Concept' comes from the latin *concipio*, meaning 'I grasp'. Thus the same latin root gives us *to conceive*, meaning that act of the intellect whereby the soul is formally identical with a material substance, 'grasping it' through or under a 'concept', and the biological act of generating a new human being in the womb. It is rather poetic that the embryo of a child of the mind and of a child of the body share an etymological ancestry.

Record View

View the records.

Record Cursor

Curse the records.

Introspection

Too deep for me, I'm afraid.

Iteration

Rinse, repeat.

Plan + Profile

You can profile your queries and get the execution plan returned.

Reading Records

Usually I listen to them, but ok.

Utility Methods

```

class Ball extends Throwable {}

class P {

    P target;

    P (P target) {
        this.target = target;
    }

    void aim (Ball ball) {

        try {
            throw Ball;
        }
        catch (Ball b) {
            target.aim (b);
        }
    }

    public static void main (String[] args) {
        P parent = new P (null);
        P child = new P (parent);
        parent.target = child;
        parent.aim (new Ball ());
    }
}

```

Session

Therapy, we all need it.

Monitoring

You can monitor the network traffic. Connect your screen to your modem with a RGB cable.

Run Statement

Example 3. Foo bar example

```
def test_can_run_simple_statement_with_params(self):
    session = GraphDatabase.driver("bolt://localhost").session()
    count = 0
    for record in session.run("RETURN {x} AS n", {"x": {"abc": ["d", "e",
"f"]}}):
        assert record[0] == {"abc": ["d", "e", "f"]}
        assert record["n"] == {"abc": ["d", "e", "f"]}
        assert repr(record)
        assert len(record) == 1
        count += 1
    session.close()
    assert count == 1
```

```
// when
Statement statement = new Statement( text, null );

// then
assertThat( statement.text(), equalTo( text ) );
assertThat( statement.parameters(), equalTo( NO_PARAMETERS ) );
```

```
    }
});
var summary = promiseResult.summarize();
//Print number of nodes created
console.log('');
console.log(summary.updateStatistics.nodesCreated());
})
.catch(function(error) {
    console.log(error);
})
.then(function(){
    promiseSession.close();
});
```

Session State

One transaction, one result. We don't do pay-it-forward.

You can do parallel sessions, tho.

Syncing

Pipelining is related to syncing.

Transactions

State

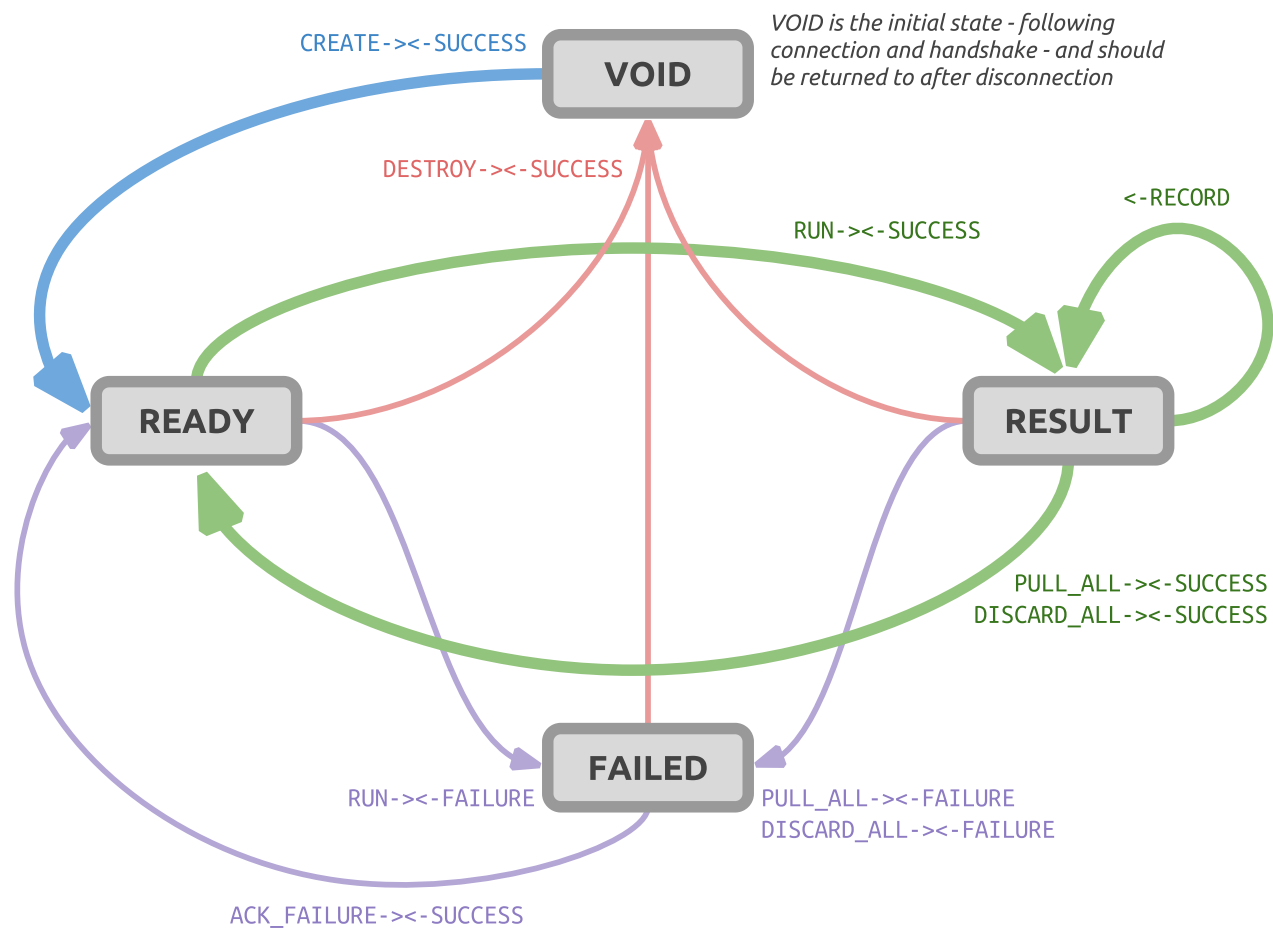


Figure 1. The state of the machine

Implicit Transcation

If you transact you will be implicated.

BEGIN, COMMIT, ROLLBACK

It's what they shout at Daft Punk concerts.

Nested Transactions

They are nice, but we don't do them yet. We probably should tell you how to deal with that.

Type Mapping

Example 4. Type mapping between the Neo4j public type system and them other languages

What are types?

Mmmhhh..

Just minify and lint it and you should be fine.