

Computer says no

About bad error messages

Sascha Peukert
Software Engineer





About Neo4j

- Creators of *Neo4j Graph Plattform* and *Neo4j* - World's leading open-source graph database
- Company founded 2007 in **Sweden**
- Today 700+ employees in San Mateo, London, **Malmö** and remote



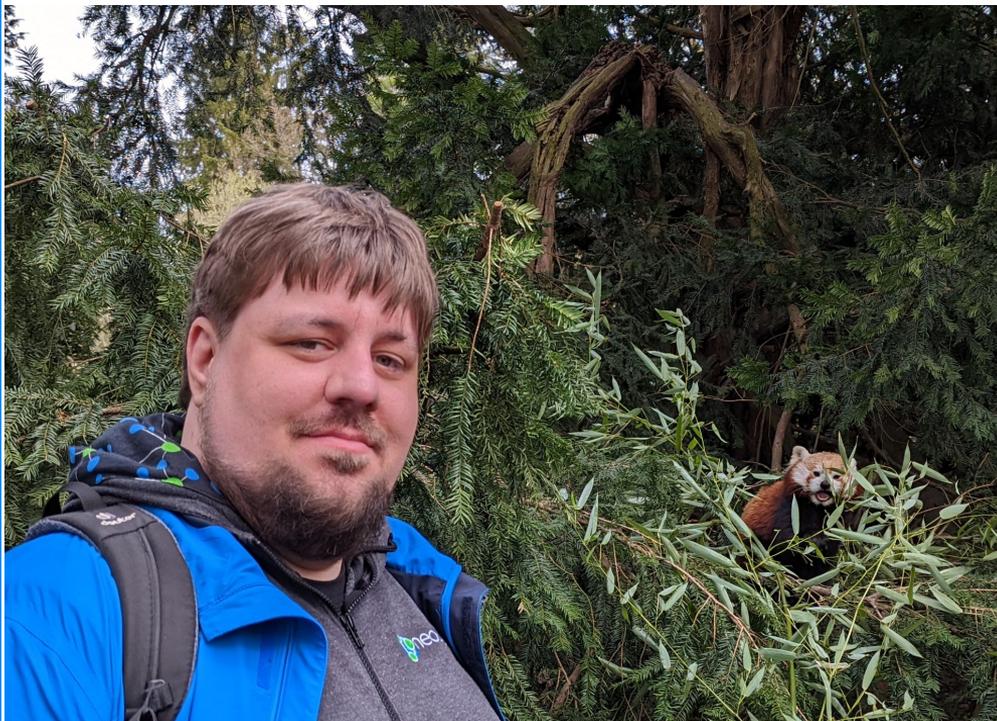
nodes 22

November 16-17

Join our free 24 hour virtual conference
by developers and data scientists

Register now: neo4j.com/nodes-2022





Sascha Peukert

Software Engineer @ Neo4j

- 2017 - 2021 Cypher
- Since 2021 Ops Manager

sascha.peukert@neo4j.com

Twitter: @SasPeuk



Your slide ran into a problem and needs to restart. We're just collecting some error info, and then we'll restart for you.

103% complete



For more information about this issue and possible fixes, visit
<http://windows.com/stopcode>

If you call a support person, give them this info:
Stop code: CRITICAL_SLIDE_DIED

Errors, Shmerrors. Why should I care?



"We found that:

- 1) participants read error messages and the difficulty of reading these messages is comparable to the difficulty of reading source code,*
- 2) **difficulty reading error messages significantly predicts participants' task performance,** and*
- 3) participants allocate a substantial portion of their total task to reading error messages (13%-25%)."*

T. Barik *et al.*, "Do Developers Read Compiler Error Messages?," *2017 IEEE/ACM 39th International Conference on Software Engineering (ICSE)*, 2017, pp. 575-585, doi: 10.1109/ICSE.2017.59.

It all started with a request



Adam Cowley Aug 25th, 2020 at 10:37 AM

Hey Team, one thing that I would really love is more context to errors in Cypher. A couple of times recently I've been writing long import statements and had to spend a long time debugging them because the errors don't contain enough information. This is the one I'm trying to tackle at the moment as part of a 57 line cypher statement with many create/set commands:

```
Neo.ClientError.Statement.TypeError  
Property values can only be of primitive types or arrays thereof
```

Even the offending label/property would be good enough.

The other one that stands out is merging on null values (eg using LOAD CSV) - `Cannot merge node using null property value for id` - if it could tell me what label/variable it was on then it would be a huge help



Enhancement: Pointing out the problem



Trying to set unsupported values (e.g. a map value) for properties will result in an error message that will now point out the erroneous value.

Before:

```
Property values can only be of primitive types or arrays thereof
```

After:

```
Property values can only be of primitive types or arrays thereof.  
Encountered: Map{A -> String("B")}.
```

The very next day



Sascha Peukert Aug 26th, 2020 at 4:38 PM

Hello good people in the fields! The Cypher implementation team decided to start a new initiative. We want to collect all the unhelpful and bad Cypher error messages that you think we can improve upon. If you have any of those, please leave them (and a bit of context if possible) as a comment on this card: [https://neo4j.com/developer/cypher-error-messages/](#)





Enhancement: Giving more context

Database content:

```
(:Person {name: "Tim"})
```

Query:

```
MATCH (n) UNWIND properties(n) as data RETURN data[0]
```

Hello valerio



```
valerio@neo4j> SHOW USER PRIVILEGES;
```

access	action	resource	graph	segment	role	user
"GRANTED"	"access"	"database"	"DEFAULT"	"database"	"role1"	"valerio"
"GRANTED"	"access"	"database"	"DEFAULT"	"database"	"PUBLIC"	"valerio"

Lets create a new user...



```
valerio@neo4j> SHOW USER PRIVILEGES;
```

access	action	resource	graph	segment	role	user
"GRANTED"	"access"	"database"	"DEFAULT"	"database"	"role1"	"valerio"
"GRANTED"	"access"	"database"	"DEFAULT"	"database"	"PUBLIC"	"valerio"

```
valerio@neo4j> CREATE USER john SET PASSWORD 'abc';
```

Permission denied!



```
valerio@neo4j> SHOW USER PRIVILEGES;
```

access	action	resource	graph	segment	role	user
"GRANTED"	"access"	"database"	"DEFAULT"	"database"	"role1"	"valerio"
"GRANTED"	"access"	"database"	"DEFAULT"	"database"	"PUBLIC"	"valerio"

```
valerio@neo4j> CREATE USER john SET PASSWORD 'abc';  
Permission denied.
```

What actually needs to happen



```
valerio@neo4j> SHOW USER PRIVILEGES;
```

access	action	resource	graph	segment	role	user
"GRANTED"	"access"	"database"	"DEFAULT"	"database"	"role1"	"valerio"
"GRANTED"	"access"	"database"	"DEFAULT"	"database"	"PUBLIC"	"valerio"

```
valerio@neo4j> CREATE USER john SET PASSWORD 'abc';  
Permission denied.
```

```
neo4j@neo4j> GRANT CREATE USER ON DBMS TO role1;
```

```
valerio@neo4j> CREATE USER john SET PASSWORD 'abc';
```

Enhancement: Pointing out what to do now

Improved error message when an operation is denied by the system. The error message now provides information about the privilege that got denied or is missing.

Before:

```
Permission denied.
```

After:

```
Permission has not been granted for CREATE USER. Try executing SHOW USER PRIVILEGES to determine the missing or denied privileges. In case of missing privileges, they need to be granted (See GRANT). In case of denied privileges, they need to be revoked (See REVOKE) and granted.
```



Yes, but actually no.

This works:

```
USE system
```

```
CALL dbms.security.listUsers() yield username, roles, flags
```

```
RETURN username as user, roles, flags
```

This failed:

```
USE system
```

```
CALL dbms.security.listUsers() yield username, roles, flags
```

```
WITH username as u, roles as r, flags as f
```

```
RETURN u, r, f
```



Yes, but actually no.

This failed:

```
USE system
```

```
CALL dbms.security.listUsers() yield username, roles, flags
```

```
WITH username as u, roles as r, flags as f
```

```
RETURN u, r, f
```

The error:

```
Not a recognised system command or procedure. This Cypher command can only be executed in a user database: CALL dbms.security.listUsers()
```

Wait, what?



Yes, but actually no.

This failed:

```
USE system  
  
CALL dbms.security.listUsers() yield username, roles, flags  
  
WITH username as u, roles as r, flags as f  
  
RETURN u, r, f
```

The error:

Not a recognised system command or procedure. This Cypher command can only be executed in a user database: CALL dbms.security.listUsers()

Okay, so I'll just run it on a user database!

This is an administration command and it should be executed against the system database:
dbms.security.listUsers

WTF?

Enhancement: Pointing out the problem, explaining the rules

The bad error messages hid a bug!

Before:

```
Not a recognised system command or procedure. This Cypher command can only be executed in a user database: CALL dbms.security.listUsers()
```

&

```
This is an administration command and it should be executed against the system database: dbms.security.listUsers
```

After:

The following unsupported clauses were used: WITH.

The system database supports a restricted set of Cypher clauses. The supported clause structure for procedure calls is: CALL, YIELD, RETURN. YIELD and RETURN clauses are optional. The order of the clauses is fix and each can only occur once.

Good error messages



Disclaimer:

There is no absolute truth as to what makes a good error message. The following are just some ideas/guidelines. Use as little or as much as you like.

Good error messages (1)

Error message style guide

- Upper/Lowercase beginning

unable to open file

Unable to open file



Freely based on: <https://www.swyx.io/write-errors-that-don-t-make-me-think-24hg>

Good error messages (1)

Error message style guide

- Upper/Lowercase beginning
- **Punctuation at the end**



unable to open file

unable to open file.

Freely based on: <https://www.swyx.io/write-errors-that-don-t-make-me-think-24hg>

Good error messages (1)

Error message style guide

- Upper/Lowercase beginning
- Punctuation at the end
- **Actionability**

no such directory

Loading service config
failed

Freely based on: <https://www.swyx.io/write-errors-that-don-t-make-me-think-24hg>

Good error messages (1)

Error message style guide

- Upper/Lowercase beginning
- Punctuation at the end
- Actionability
- **(Positive) Wording**



unable to open file

failed to open file

can't open file

impossible to open file

Freely based on: <https://www.swyx.io/write-errors-that-don-t-make-me-think-24hg>

Good error messages (1)

Error message style guide

- Upper/Lowercase beginning
- Punctuation at the end
- Actionability
- (Positive) Wording
- **Reflect logical actions, not merely function/class names**



LoadServiceConfig

Load service config

Freely based on: <https://www.swyx.io/write-errors-that-don-t-make-me-think-24hg>

Good error messages (2)

Error message design principles

- Write for users

Remember your audience

Remember the information
asymmetry between
developer/maintainer and
user



Freely based on: <https://www.swyx.io/write-errors-that-don-t-make-me-think-24hg>

Good error messages (2)

Error message design principles

- Write for users
- **Whyyy? And what to do now?**

Answer why something has happened

Provide context
(*Expected x, got y*)

Explain or hint how to fix it (maybe link to docs?)

Freely based on: <https://www.swyx.io/write-errors-that-don-t-make-me-think-24hg>

Good error messages (2)

Error message design principles

- Write for users
- Whyyy? And what to do now?
- **Searchability of error**



People will google this error, what do you want them to find?

Go1ang vs GoLand

Error codes can be helpful

Freely based on: <https://www.swyx.io/write-errors-that-don-t-make-me-think-24hg>

Good error messages (2)

Error message design principles

- Write for users
- Whyyy? And what to do now?
- Searchability of error
- **Log errors**

Do. It.

Offer formatting to make it
easier to distinguish from
normal log messages

Freely based on: <https://www.swyx.io/write-errors-that-don-t-make-me-think-24hg>

Good error messages (2)

Error message design principles

- Write for users
- Whyyy? And what to do now?
- Searchability of error
- Log errors
- **Errors are not docs**



No wall of text!

If it has to be a longer
text, structure it into
sections

Link to documentation but
not be the documentation

Freely based on: <https://www.swyx.io/write-errors-that-don-t-make-me-think-24hg>

Summary

- Don't let error messages be a side product of your engineering
 - Error messages can hide bugs
 - Good error messages -> higher productivity!
 - They are user/developer experience!
 - Those are the people that pay for your product!
- Suggestion: Collect bad error messages from your users/field team & take some time to try to improve them once in a while (Day of cleanup?)
- Create error message style guides and design principles for your team/organization/project
- A good error message "does not make me think"

<https://www.swyx.io/write-errors-that-don-t-make-me-think-24hg>

Thank you!

Contact me via

Email: sascha.peukert@neo4j.com

Twitter: [@SasPeuk](https://twitter.com/SasPeuk)

