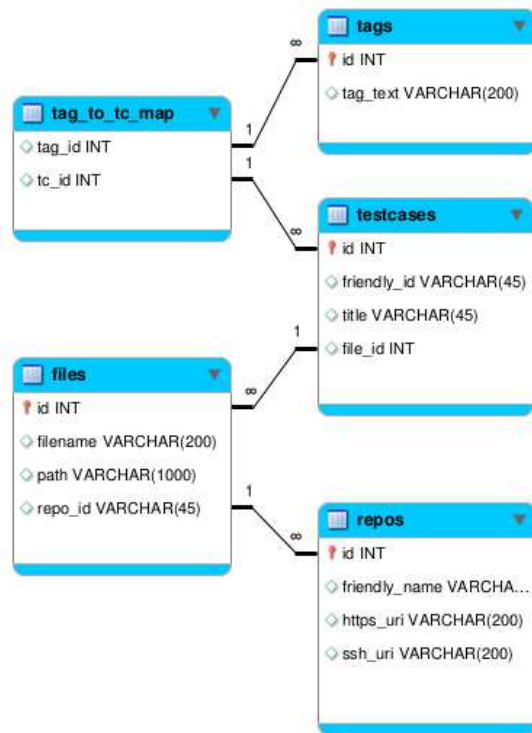


# Introduction to graph databases

Why you might find them useful...

- Tóth Attila
- HUSTEF, October 24-25, 2016, Budapest

So far ...  
... we used relational databases



## RDBMS

- Data in predefined schema
- Data in rows scattered to tables
- Relation ensured via (unique) keys
- Search is accelerated with indexes

Now we have ...

... graph databases as well

An exciting (new) technology to store and visualize data

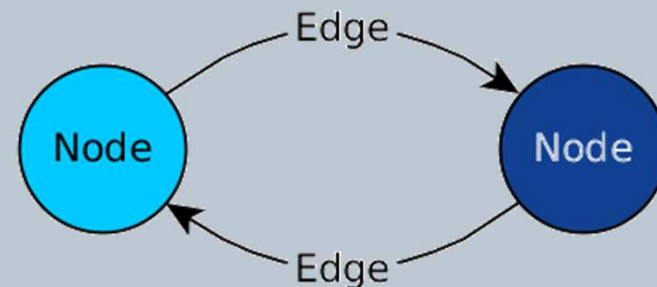
### Storage

The data is mapped to

- Nodes: objects of data
- Edges: relationship between nodes
- Properties: data assigned to nodes or edges

### Visualization

- Table view
- Graphical representation





And ...

... how does this benefit us?

### Data modelling

- Quick mapping of real life data to its database abstraction
- Easy spotting of errors or debugging data sets

### Query

- Finding related objects
  - Find nodes based on their relationship to others
  - On fixed or variable length path
- Infer relationships
  - if  $a \rightarrow b$  and  $b \rightarrow c$  then  $a \rightarrow c$

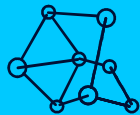
From where ...  
... it might sound familiar

Graph representation has been in use for a long time

Social  
networks



Mind  
maps



The  
World  
Wide Web



Search  
engine  
ranking

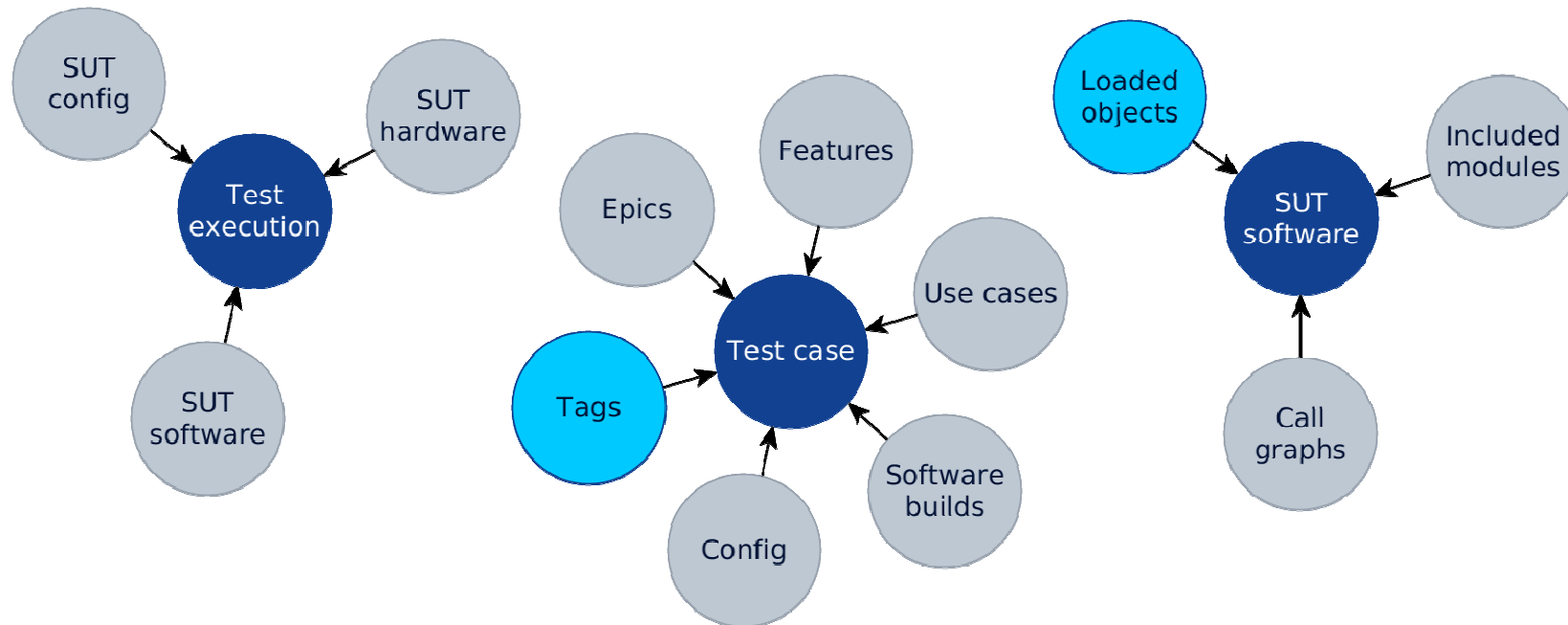


Investigative  
journalism



So let us make use of it ...  
... for testing

We also have a number of objects that are related to each other





## Example 1: Test cases and tags

### Background

#### **Problem**

- Feature related test cases are executed only in a certain release
- Subset is added to regression set
- "Rest" could be used targeted
- Automatic selection is needed

#### **Test scripts**

- Different test levels and SUT config
- Scattered to multiple repositories and files
- Use different test tools

#### **Solution**

- Tag test cases
- Create test sets dynamically based on chosen tags using graph DB

## Example 1: Test cases and tags

### Database schema

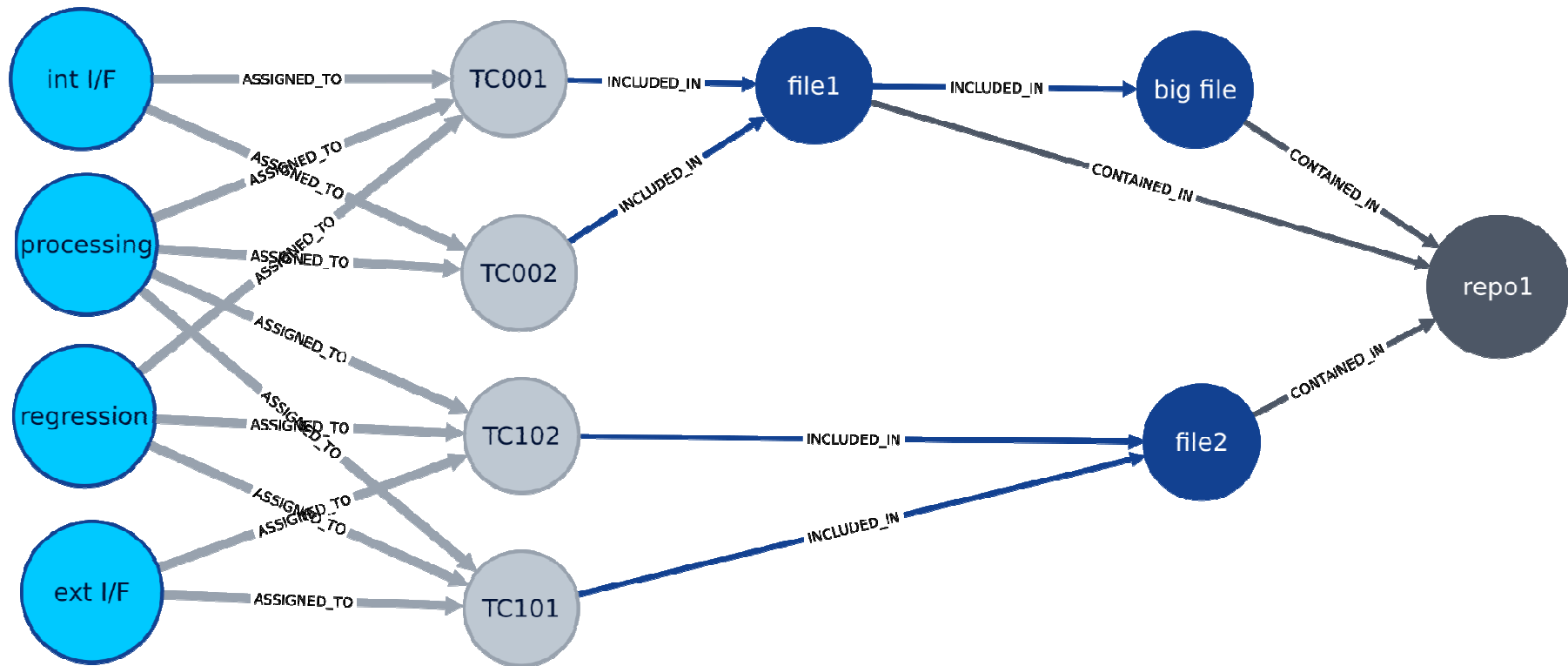
- Nodes:
  - Tag
  - Test case
  - File
  - Repository
- Relationships:





## Example 1: Test cases and tags

Simplified example



## Example 2: Security test for dependency loading

### Background

#### Static vs. dynamic linking

- Static: everything is compiled into the binary
- Dynamic: binary loads its dependencies dynamically during runtime

#### Assumptions

- Dynamic loaded files can further load other libraries
- The load chain is arbitrarily long

## Example 2: Security test for dependency loading

### Problem statement

Inspired by: <https://github.com/quarkslab/binmap>

#### **Problem**

- If there is a vulnerability in one of the OS provided libraries,
- How do we know to what extent our application is affected?

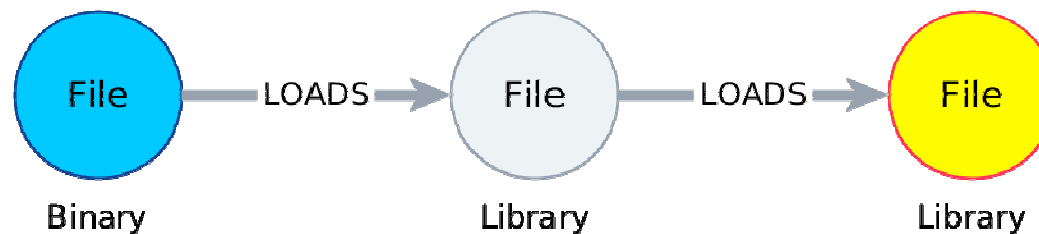
#### **Solution**

- Use graph DB and search the answer for
  - Who is loading the vulnerable library and function?
  - Is our application loading it via other libraries?

## Example 2: Security test for dependency loading

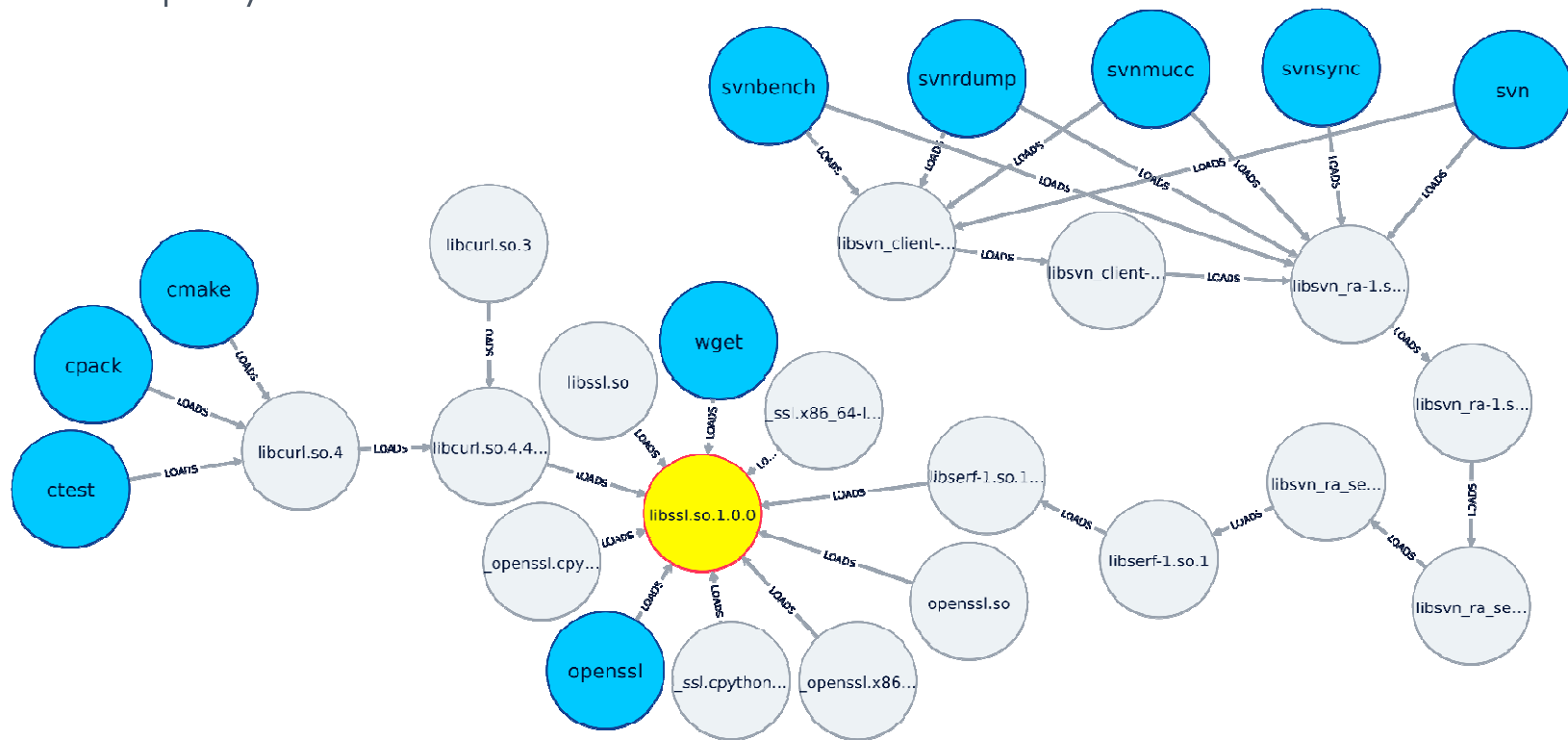
### Database model

- Nodes:
  - File: binary and library files
  - Properties: file path
- Edges:
  - Relationship: which file loads which other file
  - Properties: functions loaded



## Example 2: Security test for dependency loading

Random query for libssl on Ubuntu



Some points to consider ...  
... based on the examples

### **Performance**

- Measure and compare with traditional RDBMS and different graph DB solutions
- Graph DBs slow down on increasing number of relations / node
- RDBMS slow down on increasing number of nodes

### **Compatibility**

- Graph DBs use proprietary (non-standard) query languages
- There is no standard like SQL for RDBMS
- Swap from one graph DB to the other is not so easy



## Conclusion ...

... even simpler put

Say **NO** to Graph DB when

- It is faster with traditional RDBMS
- You need complex recursive graph queries to achieve what you want

Say **YES** to Graph DB when

- It is faster with Graph DB
  - No recursion is needed
  - Path traversal faster
- Relations vary in length
- Willingness to learn the specifics of the chosen graph DB solution

If you want to try ...  
... graph database implementations

- Lot of tools (both proprietary and open source) to choose from
- Note: feature sets are overlapping, but different
- Some examples with open source edition:

**ArangoDB**

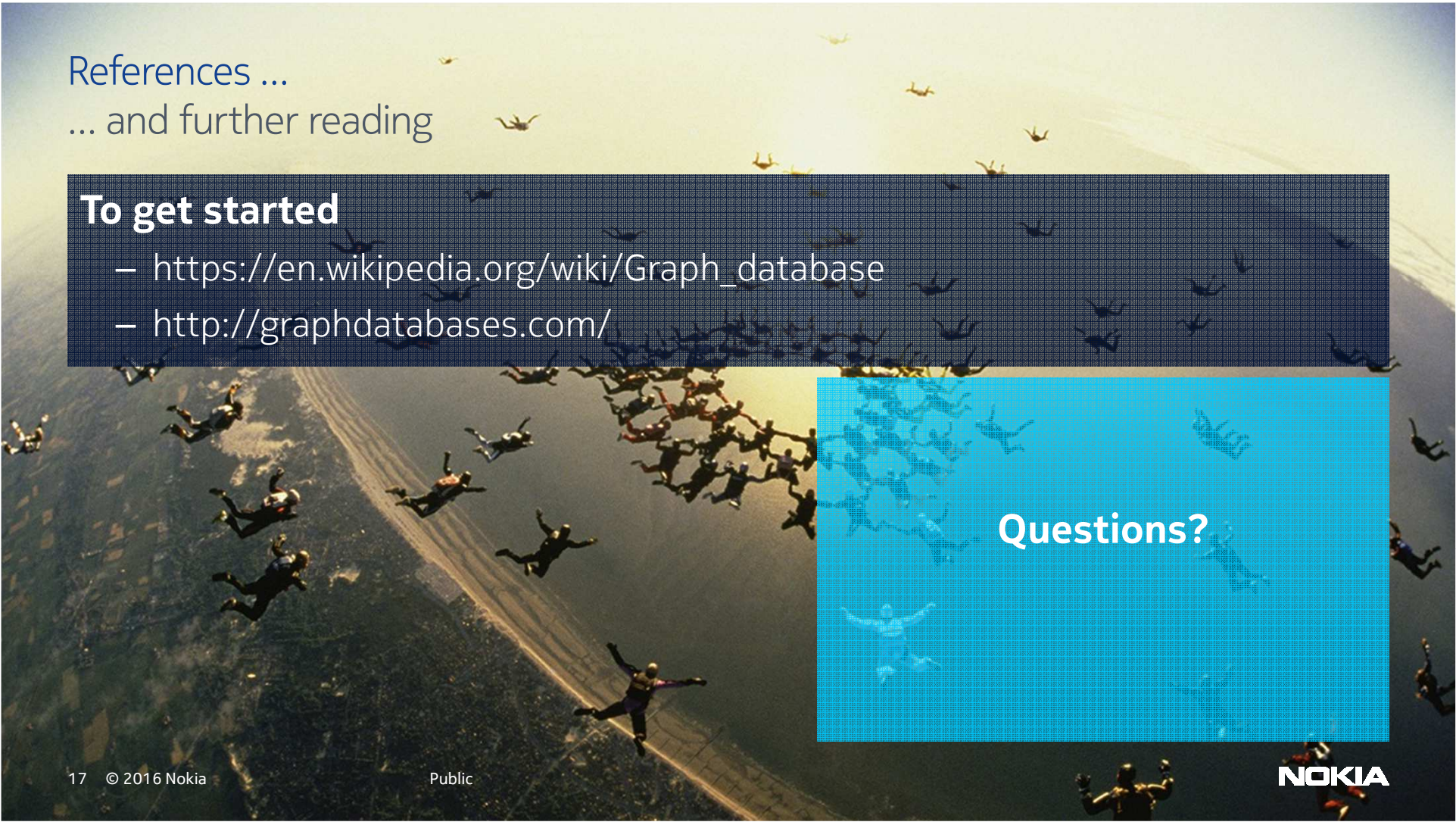
<https://www.arangodb.com>

**Neo4j**

<https://neo4j.com>

**OrientDB**

<http://orientdb.com>



References ...  
... and further reading

## To get started

- [https://en.wikipedia.org/wiki/Graph\\_database](https://en.wikipedia.org/wiki/Graph_database)
- <http://graphdatabases.com/>

Questions?

**NOKIA**

# Copyright and confidentiality

The contents of this document are proprietary and confidential property of Nokia. This document is provided subject to confidentiality obligations of the applicable agreement(s).

This document is intended for use of Nokia's customers and collaborators only for the purpose for which this document is submitted by Nokia. No part of this document may be reproduced or made available to the public or to any third party in any form or means without the prior written permission of Nokia. This document is to be used by properly trained professional personnel. Any use of the contents in this document is limited strictly to the use(s) specifically created in the applicable agreement(s) under which the document is submitted. The user of this document may voluntarily provide suggestions, comments or other feedback to Nokia in respect of the contents of this document ("Feedback").

Such Feedback may be used in Nokia products and related specifications or other documentation. Accordingly, if the user of this document gives Nokia Feedback on the contents of this document, Nokia may freely use, disclose, reproduce, license, distribute and otherwise commercialize the feedback in any Nokia product, technology, service, specification or other documentation.

Nokia operates a policy of ongoing development. Nokia reserves the right to make changes and improvements to any of the products and/or services described in this document or withdraw this document at any time without prior notice.

The contents of this document are provided "as is". Except as required by applicable law, no warranties of any kind, either express or implied, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose,

are made in relation to the accuracy, reliability or contents of this document. NOKIA SHALL NOT BE RESPONSIBLE IN ANY EVENT FOR ERRORS IN THIS DOCUMENT or for any loss of data or income or any special, incidental, consequential, indirect or direct damages howsoever caused, that might arise from the use of this document or any contents of this document.

This document and the product(s) it describes are protected by copyright according to the applicable laws.

Nokia is a registered trademark of Nokia Corporation. Other product and company names mentioned herein may be trademarks or trade names of their respective owners.

## Revision history and metadata

Please delete this slide if document is uncontrolled

Document ID: D547165604 Document Location: IMS Sharenet Organization: MN CC Cloud SDM Registers R&D									
Version	Description of charges	Date	Author	Owner	Status	Reviewed by	Reviewed date	Approver	Approval date
0.1	First version	2016-09-14	Tóth Attila	Tóth Attila	Verify				
1.0	Review comments applied and approved	2016-09-15	Tóth Attila	Tóth Attila	Approved		2016-09-15	Bujdosó Tibor	2016-09-15