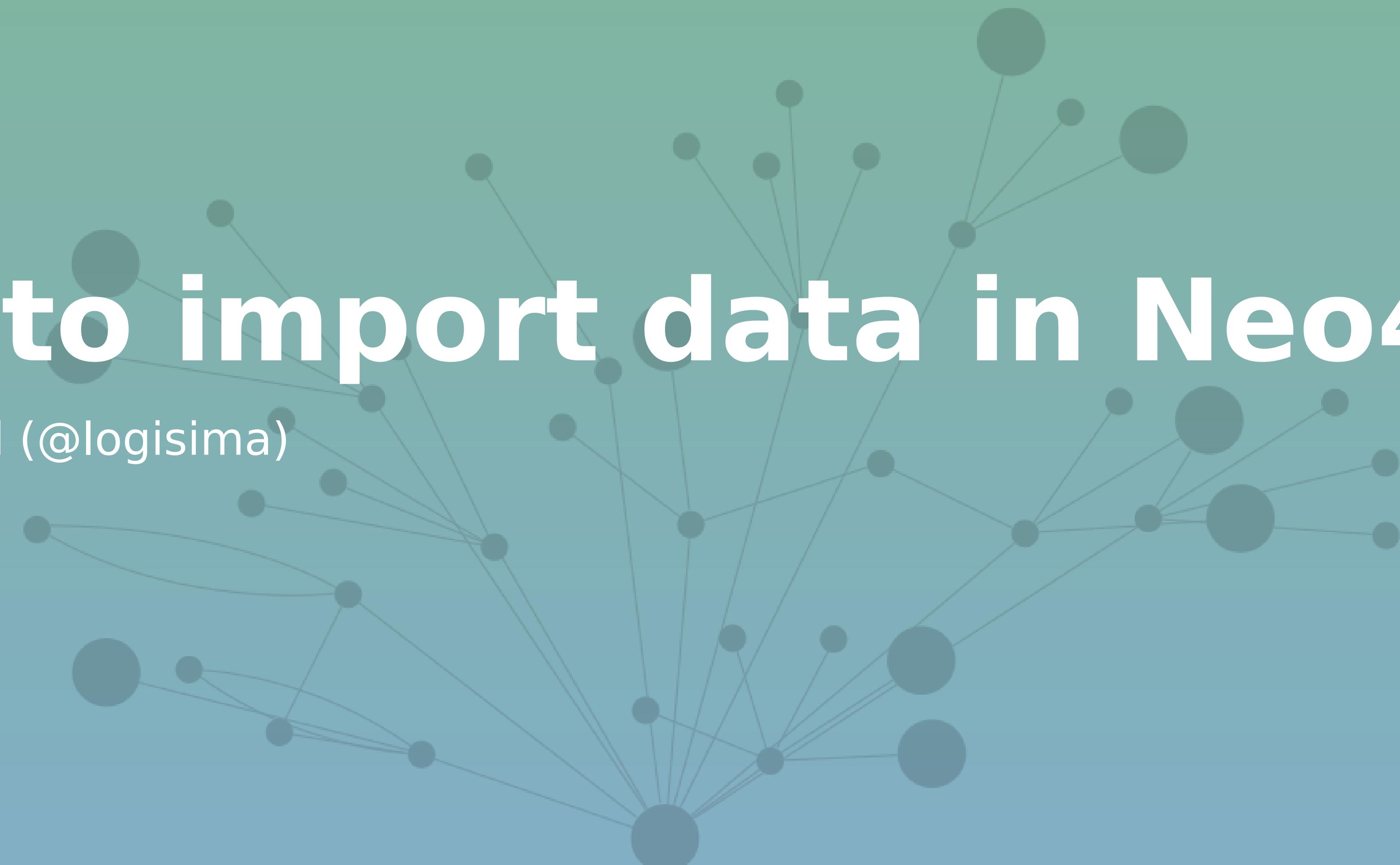


How to import data in Neo4j



Benoit Simard (@logisima)



Who I am

Benoit Simard

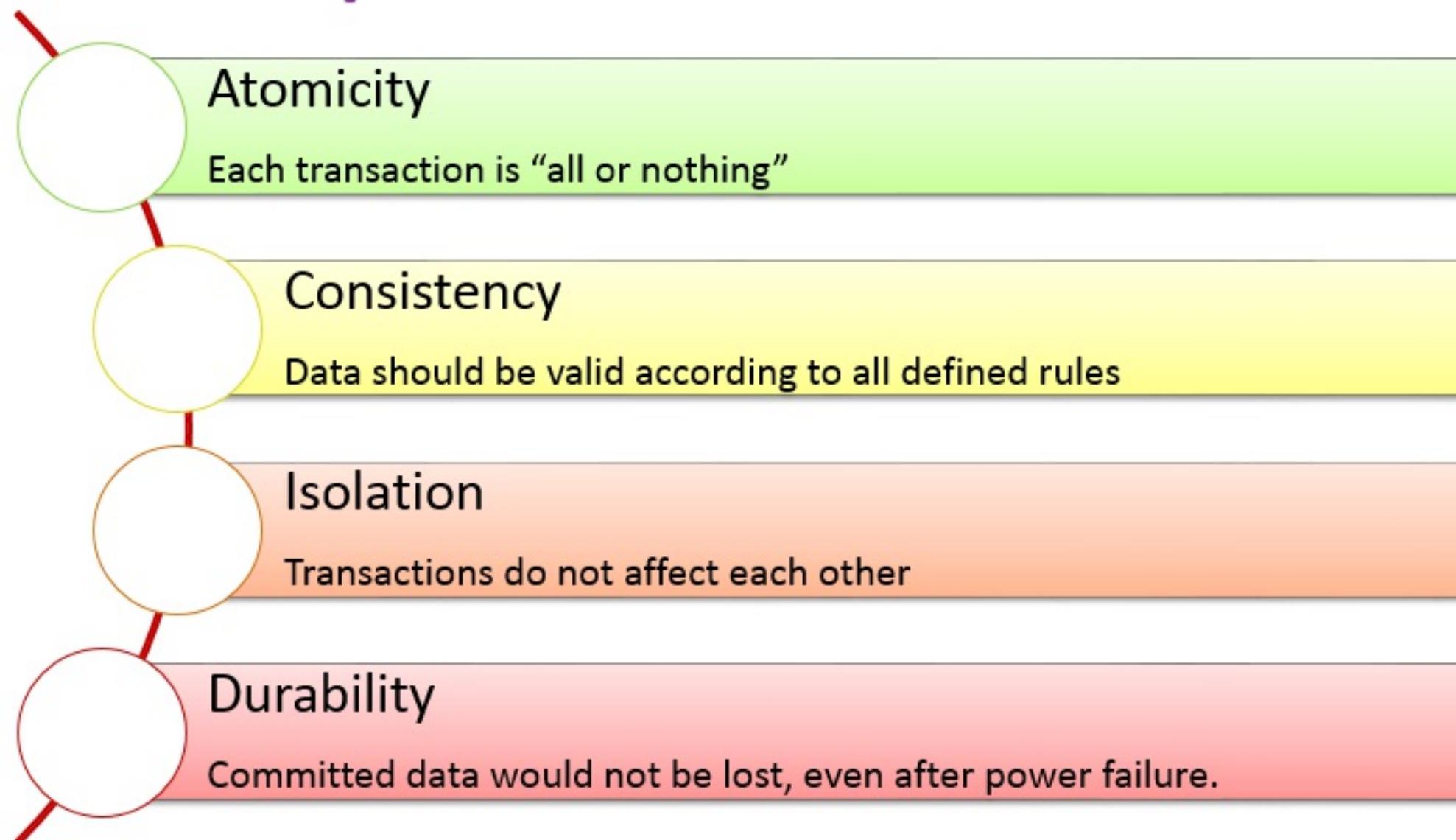


- Neo4j consultant for 3 years
- Mainly on the french territory
- Web developper, addict to graph & open data
- **Mail : benoit@neo4j.com**
- **Twitter : @logisima**

Some Reminders

ACID Database

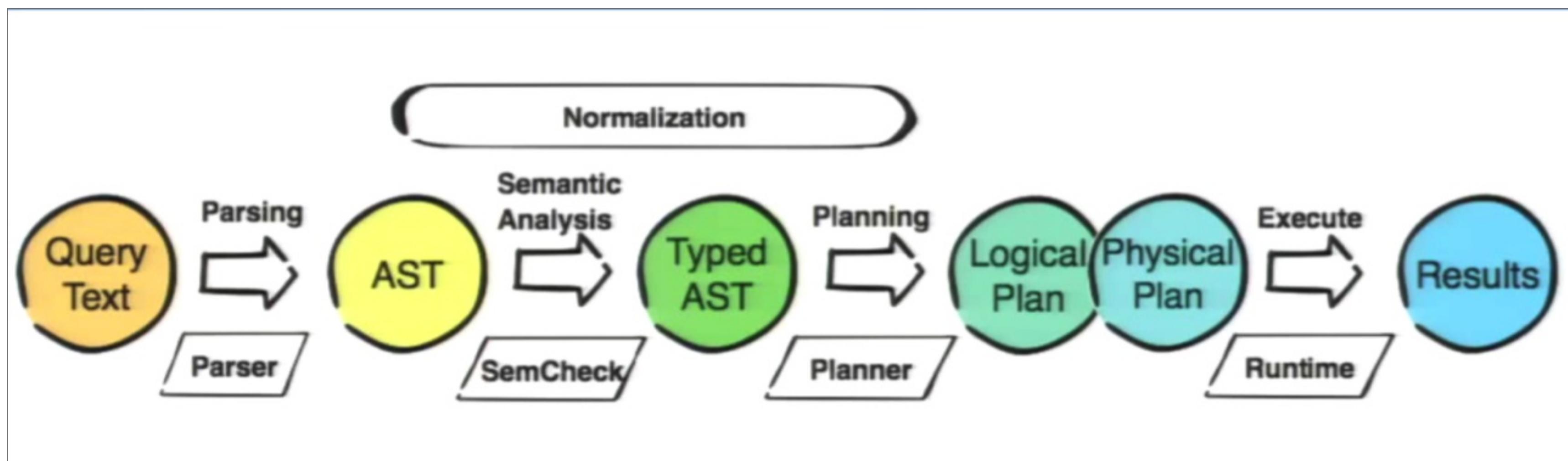
ACID Properties



- Transaction are **all or nothing**
 - *Retry mechanism*
- **Lock manager** : locks nodes and relationships during a transaction
 - *You can spend a lot of time to wait a lock or even worse to have a dead lock.*
- Everything is sequentially written into the **transaction log**
 - *You need a good hard drive*

Cypher / Query plan

- Cypher is a declarative language, like SQL
- Need to be parsed (AST)
- Interpreted by an optimizer to create its query plan
- Query plan is then executed



*All those operations take times, so you have to parameterized your queries
Neo4j will be able to reuse a query plan from its cache.*

Neo4j configuration hints

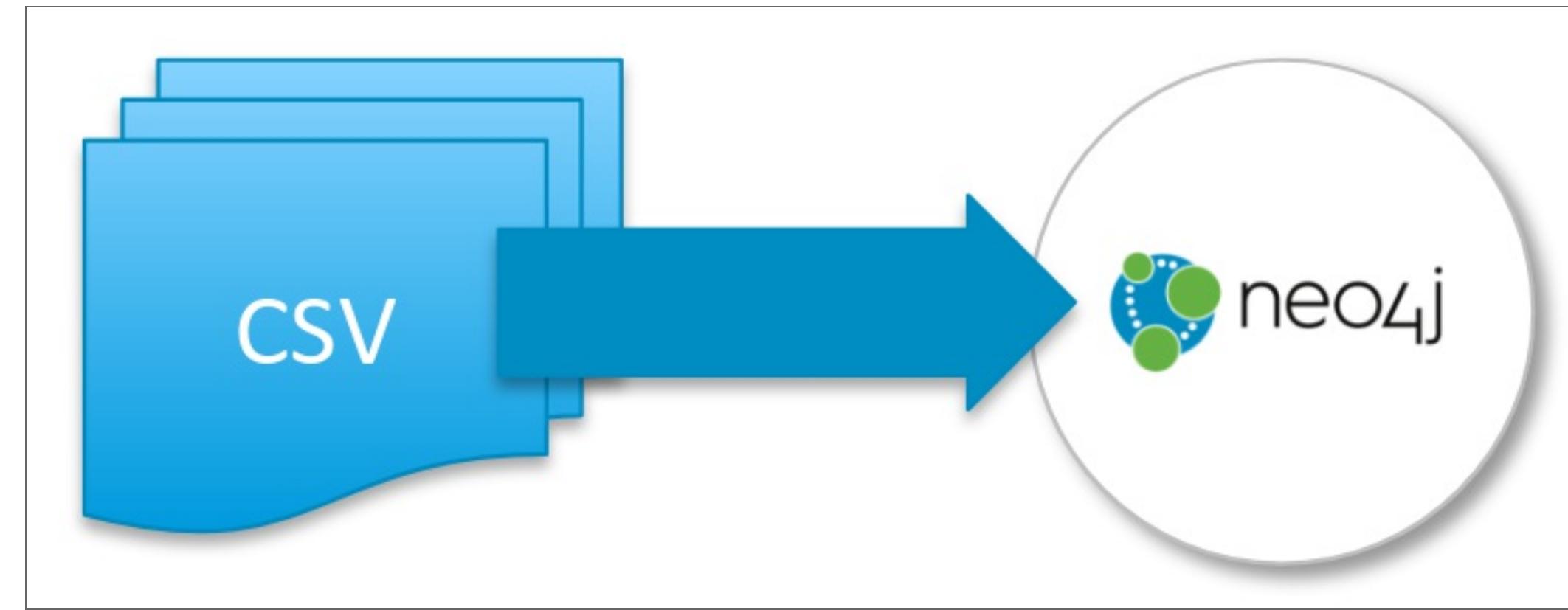
<https://neo4j.com/docs/operations-manual/current/reference/configuration-settings>

- **cypher.min_replan_interval** : The minimum time between possible cypher query replanning events (10s).
- **cypher.statistics_divergence_threshold** : The threshold when a plan is considered stale (0.75).
- **dbms.query_cache_size** : The number of Cypher query execution plans that are cached (1000).

Overview of import methods

Method 1 : Cypher LOAD CSV

<http://neo4j.com/docs/developer-manual/current/cypher/clauses/load-csv/>



Methodode 1 : Resume

Most

- Plain Cypher
- Transactionnal
- Really easy to put in place
- Batch your transaction for you
- Fast for up to 10 Million of entities
- Based on CSV files : no flux with the IS, easy to generate

Least

- You can **NOT** do a lot of extract / transformation (just what cypher can do)
- Slow for an initial import with a lot of data.

Method 2 : Plain Cypher (like in SQL)

- Again, use **Parameterized queries**
 - **Batch** your transactions
 - Use the **WITH ... UNWIND pattern** : less network traffic and can be usefull to refactor some queries in one (so reused of the query plan).
-
-

Method 2 : Resume

Most

- Plain Cypher
- Transactionnal
- A lot of freedom

Least

- You have to write a lot of code
- Slow for an initial import with a lot of data.

Method 3 : Neo4j import tool

<https://neo4j.com/docs/operations-manual/current/tools/import/>

You can import really fast a huge amount of data with this tool. It bypass some Neo4j internal mechanisms (like transaction) to be super fast.

Most

- Really really fast
- Perfect to initiate a database
- Easy to use (just one command line)

Least

- Can only initialize a database (offline and empty database)
- No transaction
- Strict format for the CSV files

Method 4 : Batch Inserter

<https://neo4j.com/docs/operations-manual/current/tools/import/>

You can import/update a lot of data into a Neo4j database.

Most

- Fast
- Can initiate and update a database

Least

- Mono-threading
- Need to write some java code
- No transaction
- Database must be offline



To resume

Method	For init/ update	Transactional	Size	Rapidity	Easy of use
LOAD CSV	BOTH	TRUE	< 10M	* * *	* * * * *
Cypher queries	BOTH	TRUE	No limit	* * * *	* * *
Import tool	INIT	FALSE	No limit	* * * * *	* * * *
Batch Inserter	BOTH	FALSE	10M - 50M	* * *	* *

Talend

Neo4j components

Project

[**http://sim51.github.io/neo4j-talend-component/**](http://sim51.github.io/neo4j-talend-component/)

Will be part of the next version of Talend studio

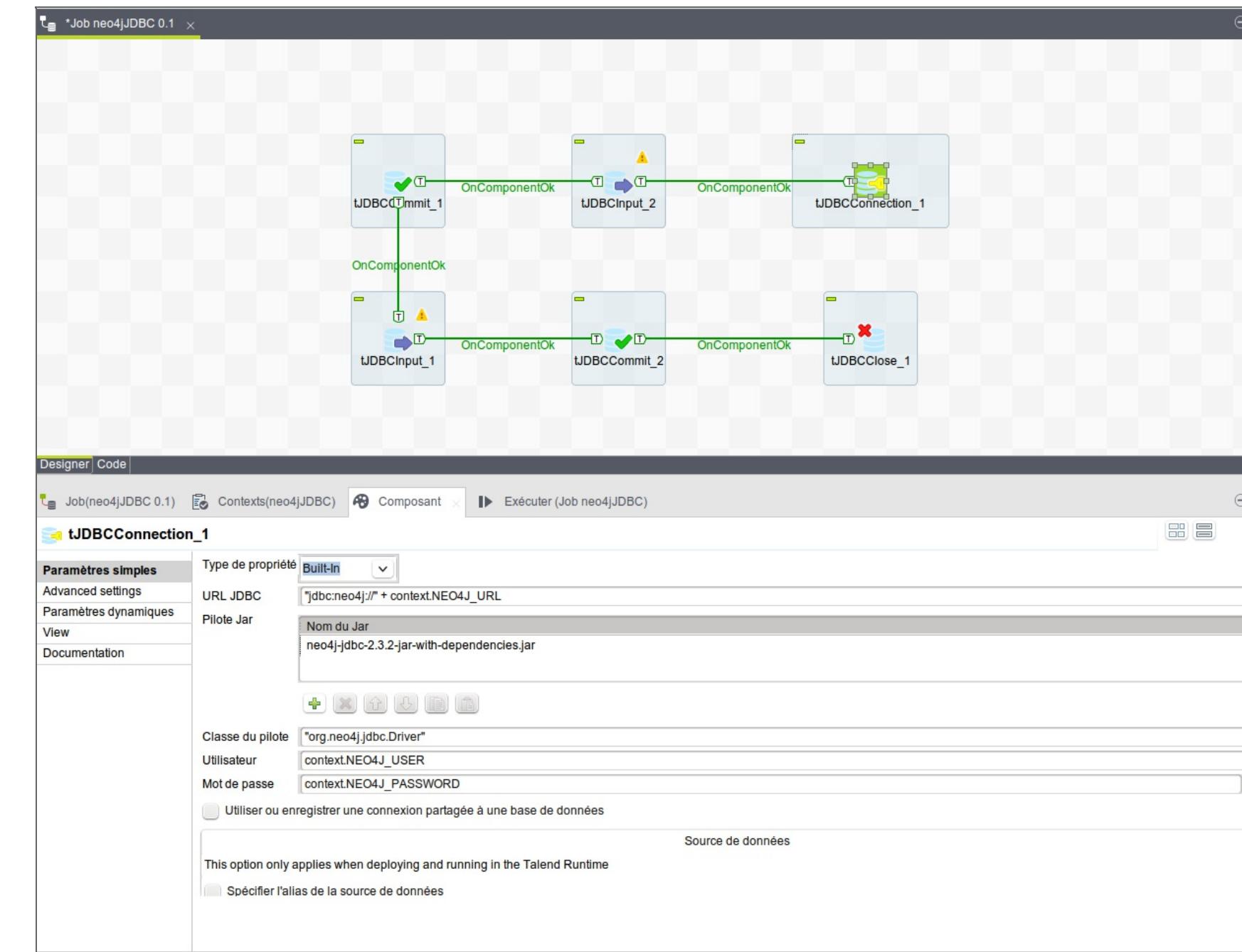
Focus on

- Neo4j import tool
- Batch Inserter mode

JDBC Component

Talend has some JDBC components and Neo4j has an official JDBC Driver : <http://neo4j-contrib.github.io/neo4j-jdbc/>

You can use the component tRow to insert data into Neo4j, but there is one lack : you can't specify the batch size ⇒ autocommit or one big transaction.



Questions to ask

Questions

- Where the data come from ?
- Is there more than one data sources ?
- Do you need to enhance some data with an other datasource ?
- Do you have some security restrictions in your IS (firewall, DMZ, ...) ?
- What is the amount of data you want to import at once ?
- How long should take an import ?
- Do you only need to initiate the database ?
- All the writes to Neo4j come from the ETL ?
- How many process do you want to do ?