

Neo4j implementation updates

Bugfixes and decisions

Mats Rydberg

mats@neotechnology.com



Neo4j and Cypher

- Neo4j has the first ever Cypher implementation
- Neo4j viewed as reference implementation, ie the Truth™
- Neo4j implementation still contains bugs, and Cypher is still not fully specified
 - Changes will be introduced and semantical decisions made
 - 'Bugs' / 'corner cases'
 - Sometimes our mental model is wrong, sometimes the implementation is wrong

Neo4j -- recent decisions

- LIMIT and updates
- UNWIND and non-lists
- Order of UNION fields

LIMIT and updates

```
MATCH (s:Start)
CREATE (n)-[:TO]->(:End)
RETURN n.position
LIMIT 10
```

- Each clause completes before next starts
- Subclauses CIP suggests LIMIT allowed directly on reading clauses

```
MATCH (s:Start)
WITH *
LIMIT 10
CREATE (n)-[:TO]->(:End)
RETURN n.position
```

Never-ending query:

```
MATCH ()
CREATE ()
```

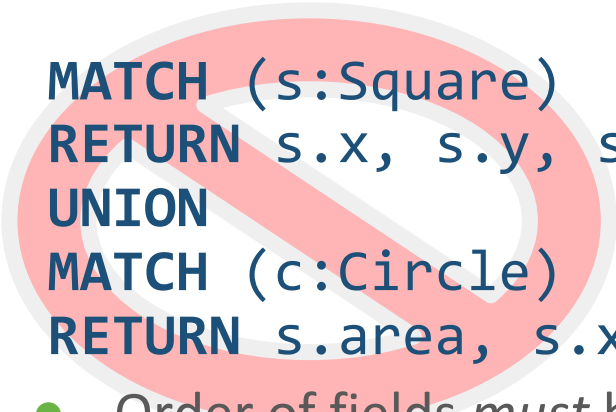
UNWIND and non-lists

```
UNWIND [null, 'string', []] AS i
UNWIND i AS j
RETURN j
```

- Any non-list value is treated as singleton list, including null
- Empty list halts execution

```
j
-----
null
'string'
```

Order of UNION fields



```
MATCH (s:Square)
RETURN s.x, s.y, s.area
UNION
MATCH (c:Circle)
RETURN s.area, s.x, s.y
```

```
MATCH (s:Square)
RETURN s.x, s.y, s.area
UNION
MATCH (c:Circle)
RETURN s.x, s.y, s.area
```

- Order of fields *must* be the same for *all* sub-parts of UNION
- Order of fields are relevant for return records
 - This may be up for wider debate

Questions?