



DATASTAX

Beyond Read-Modify-Write

@ALTobey

Open Source Mechanic for Apache Cassandra

Datastax

Apache Cassandra



The Problem



The Problem

Users expect their infrastructure to Just Work.



The Problem

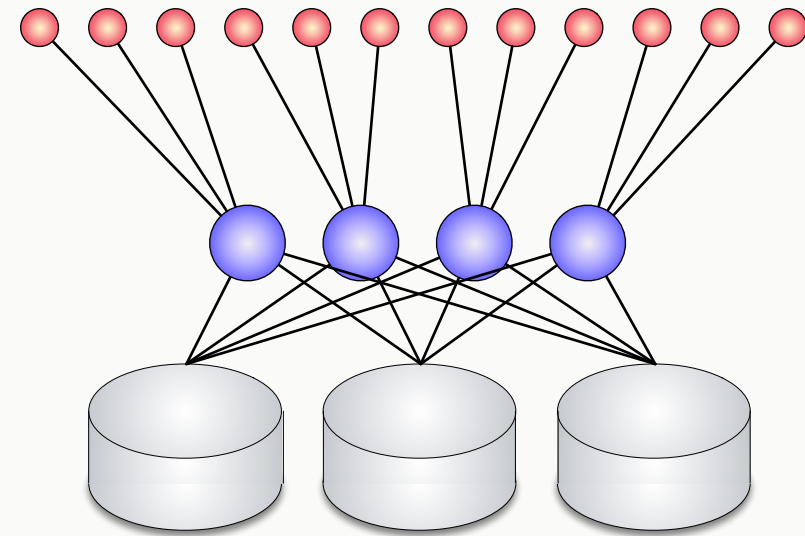


The Problem

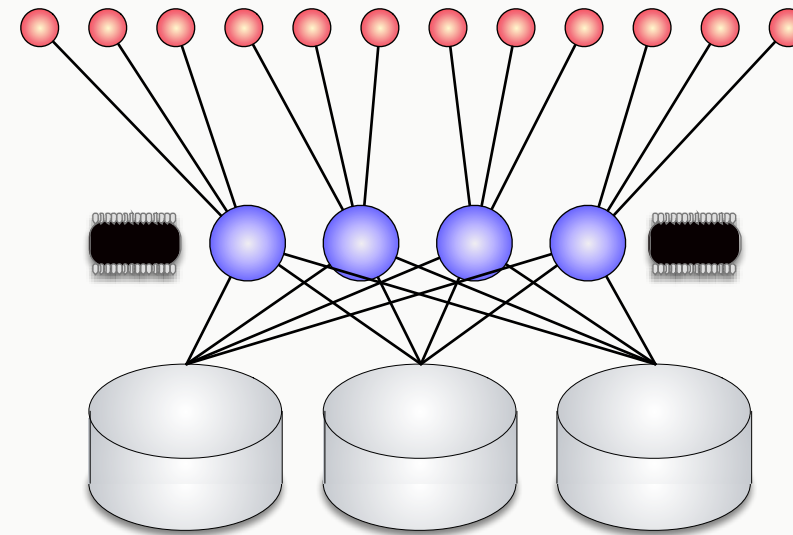


Evolution

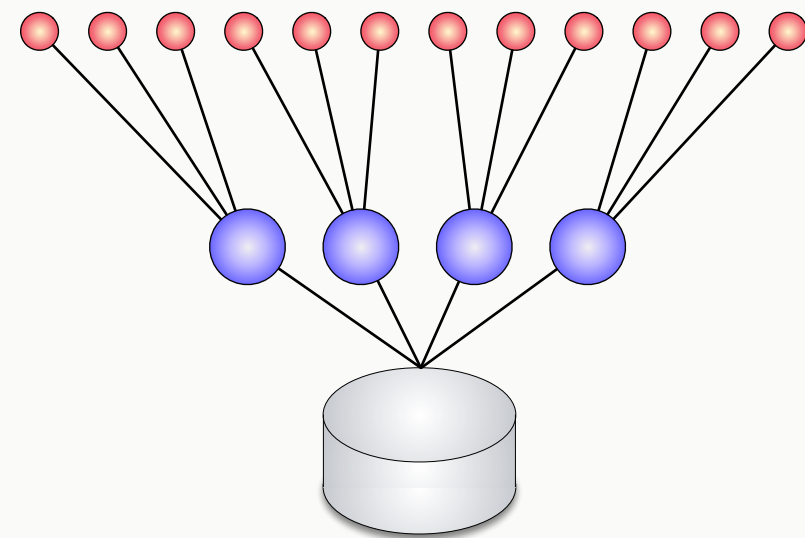
3-tier + read scaled DB



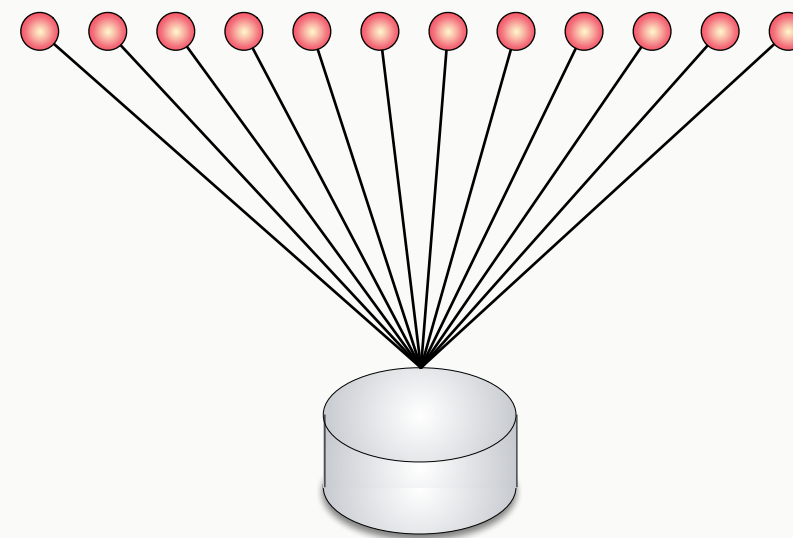
+ cache



Classic 3-tier



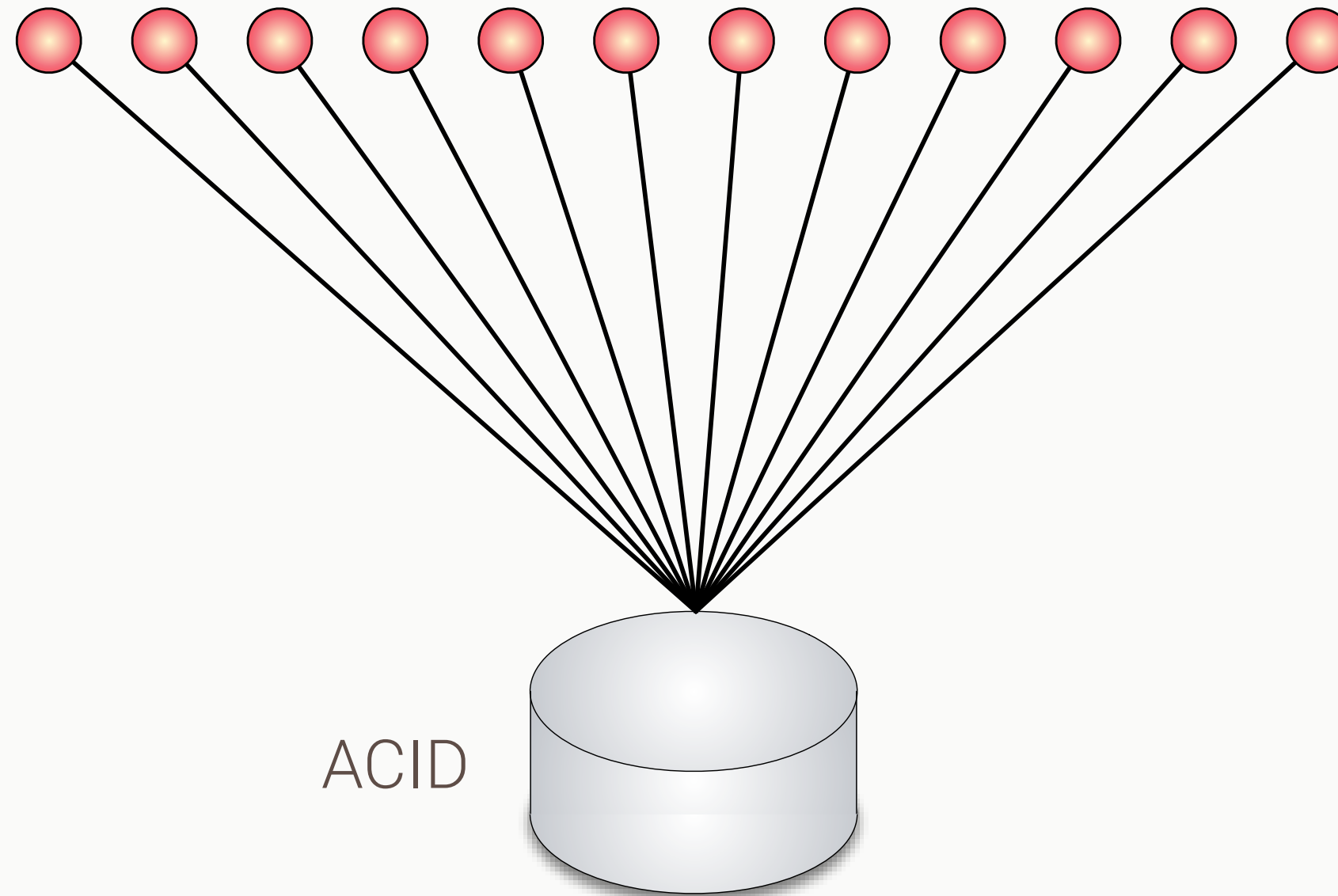
Client/Server



Client/Server



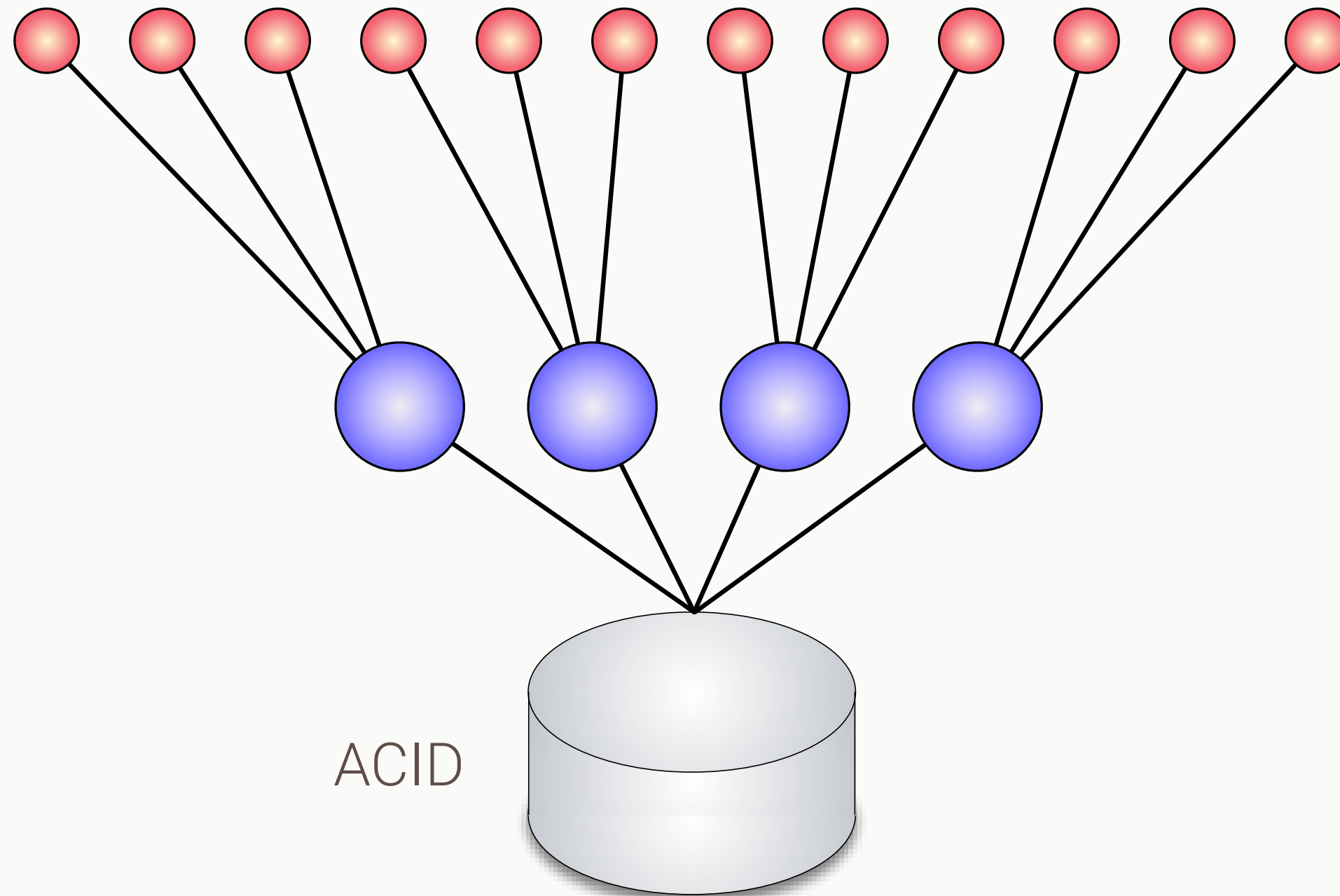
Client/Server



3-tier



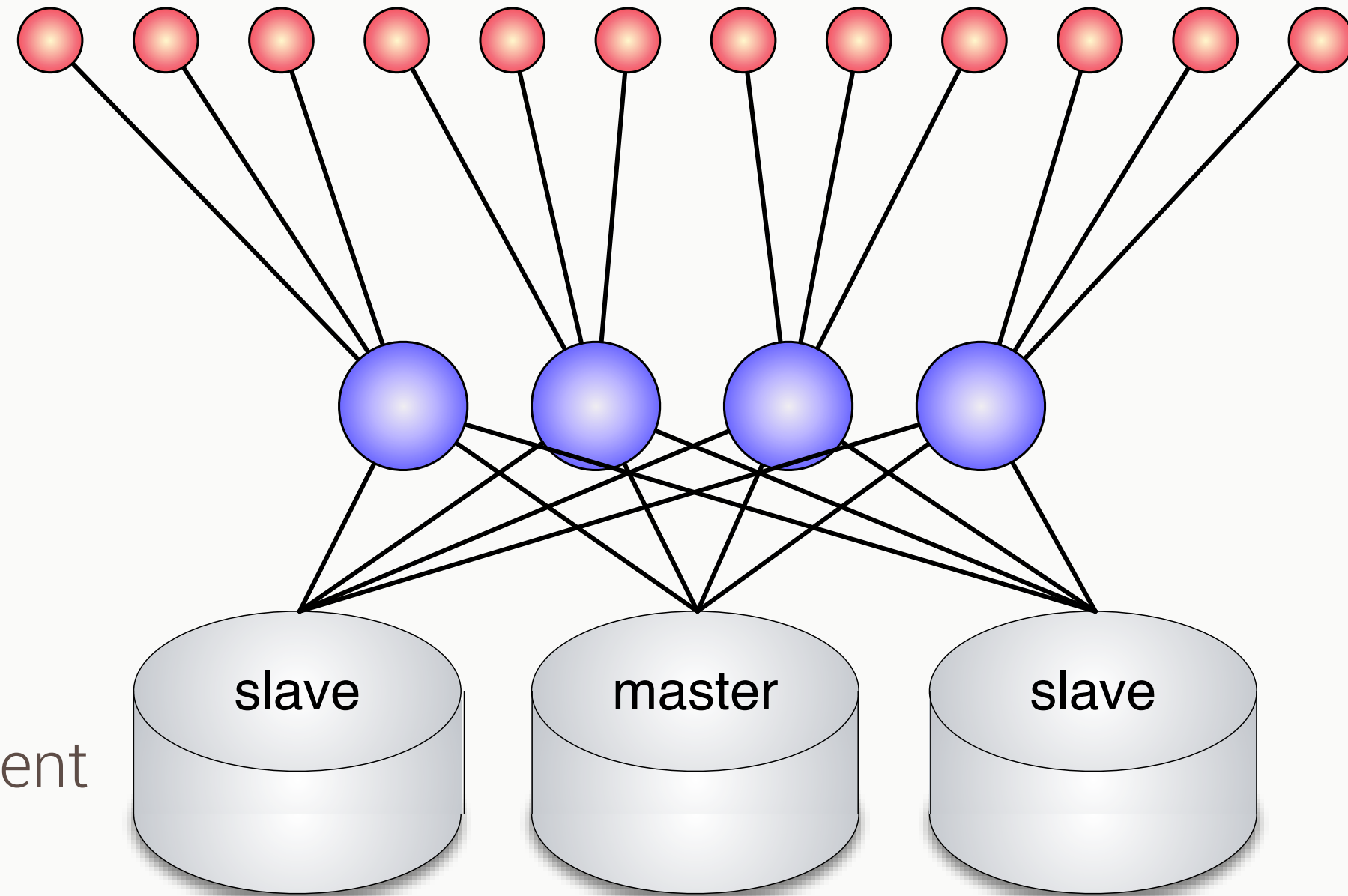
3-tier



3-tier master/slave



3-tier master/slave

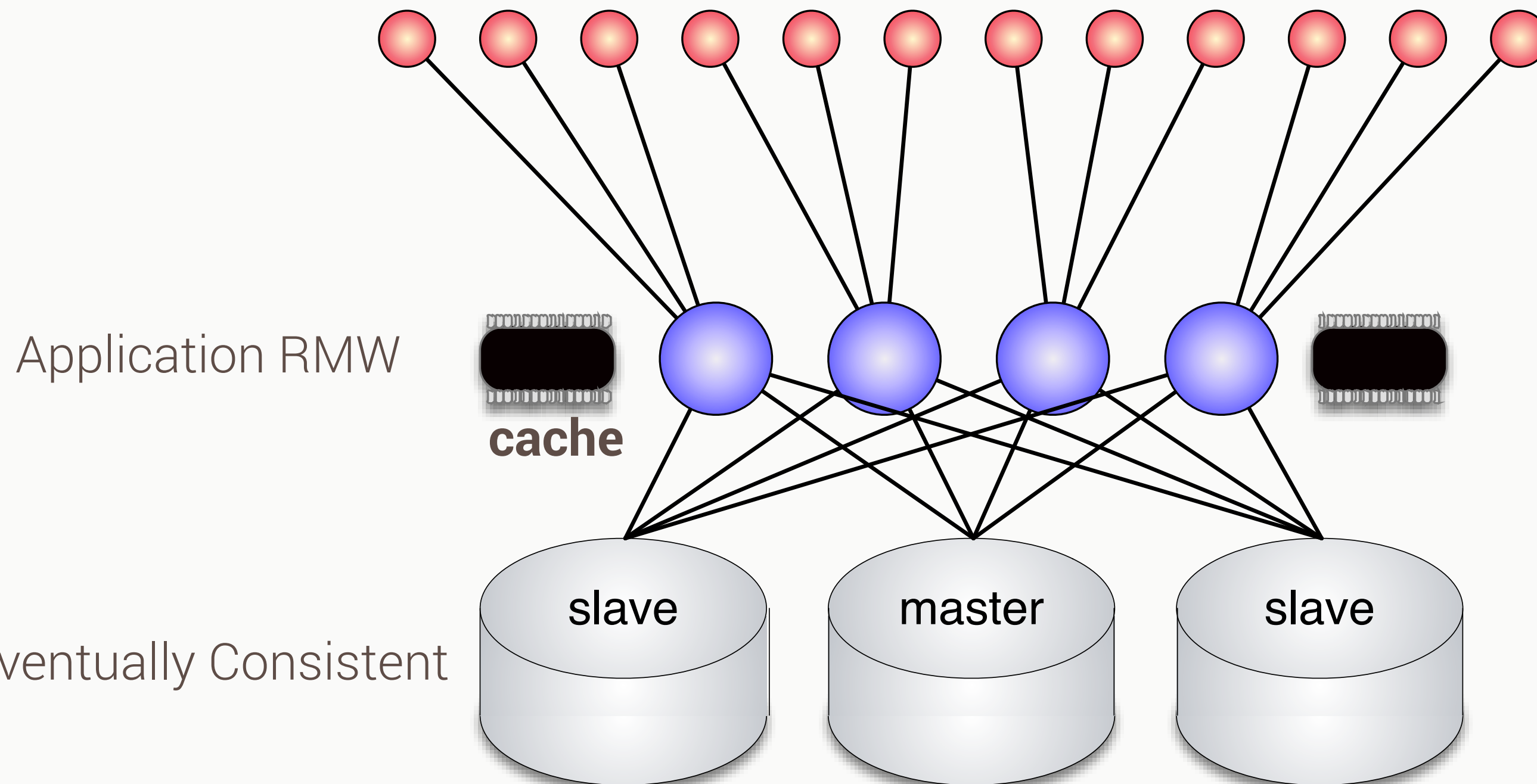


Eventually Consistent

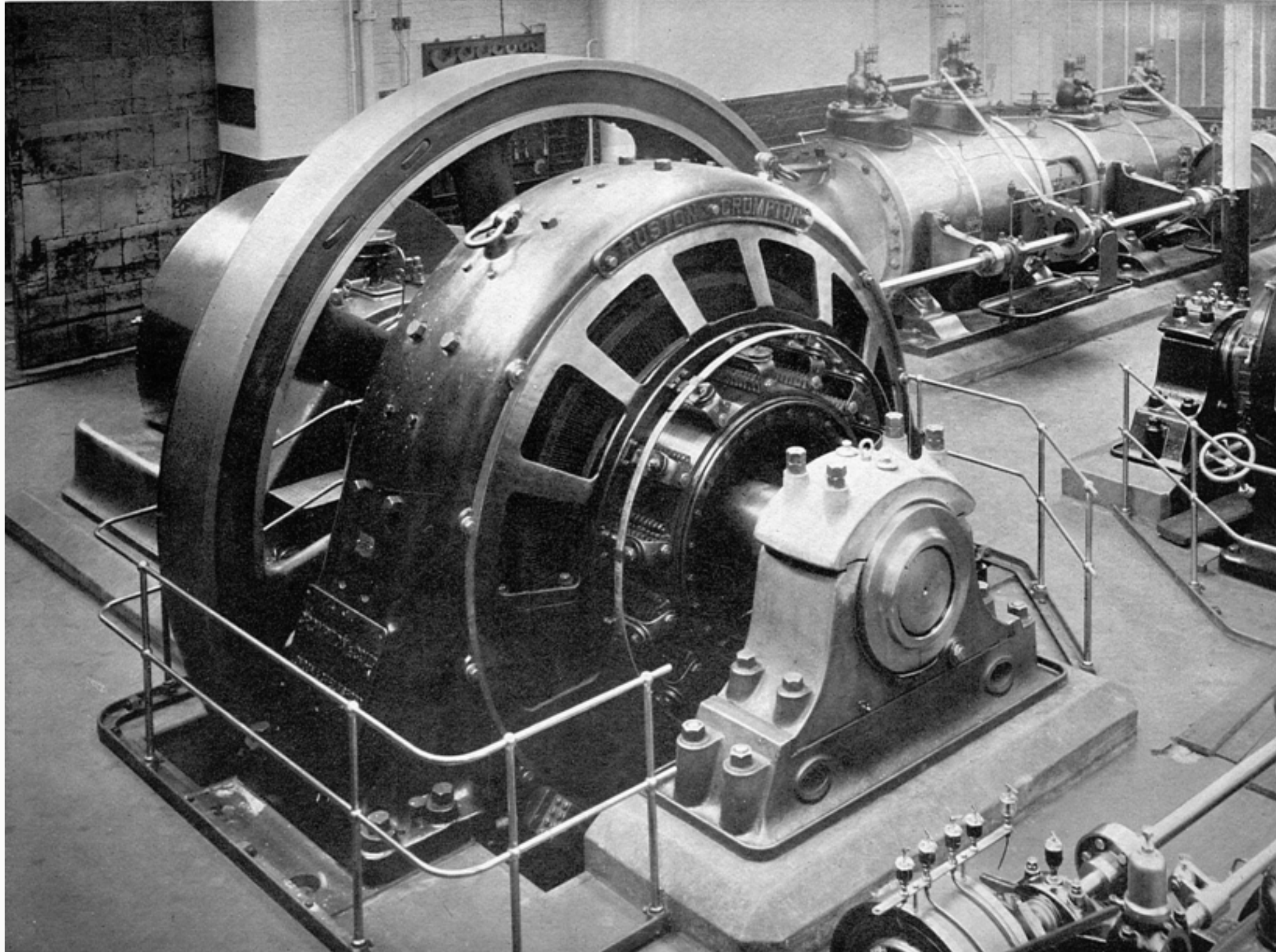
3-tier master/slave + cache



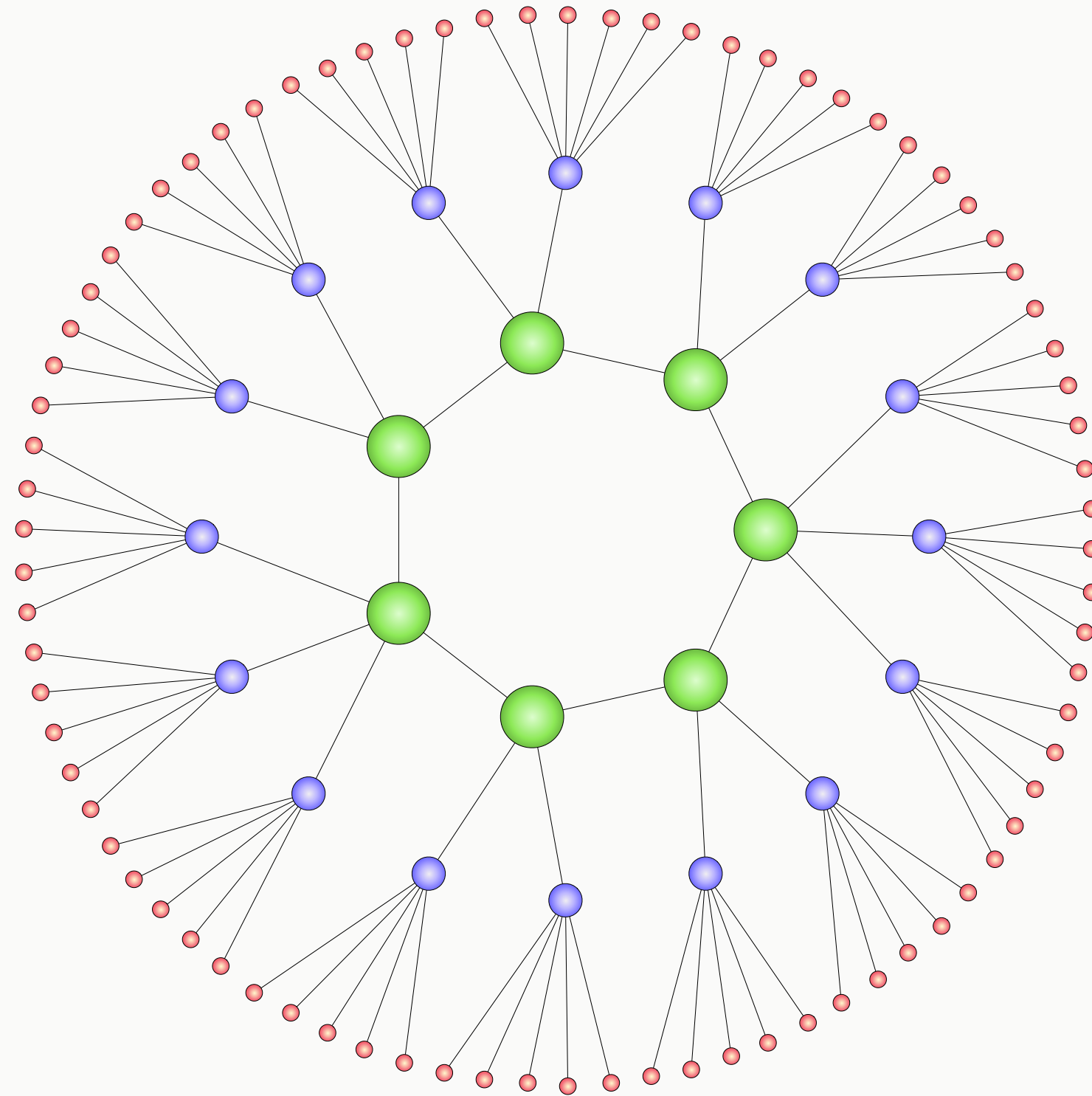
3-tier master/slave + caching



Dynamo



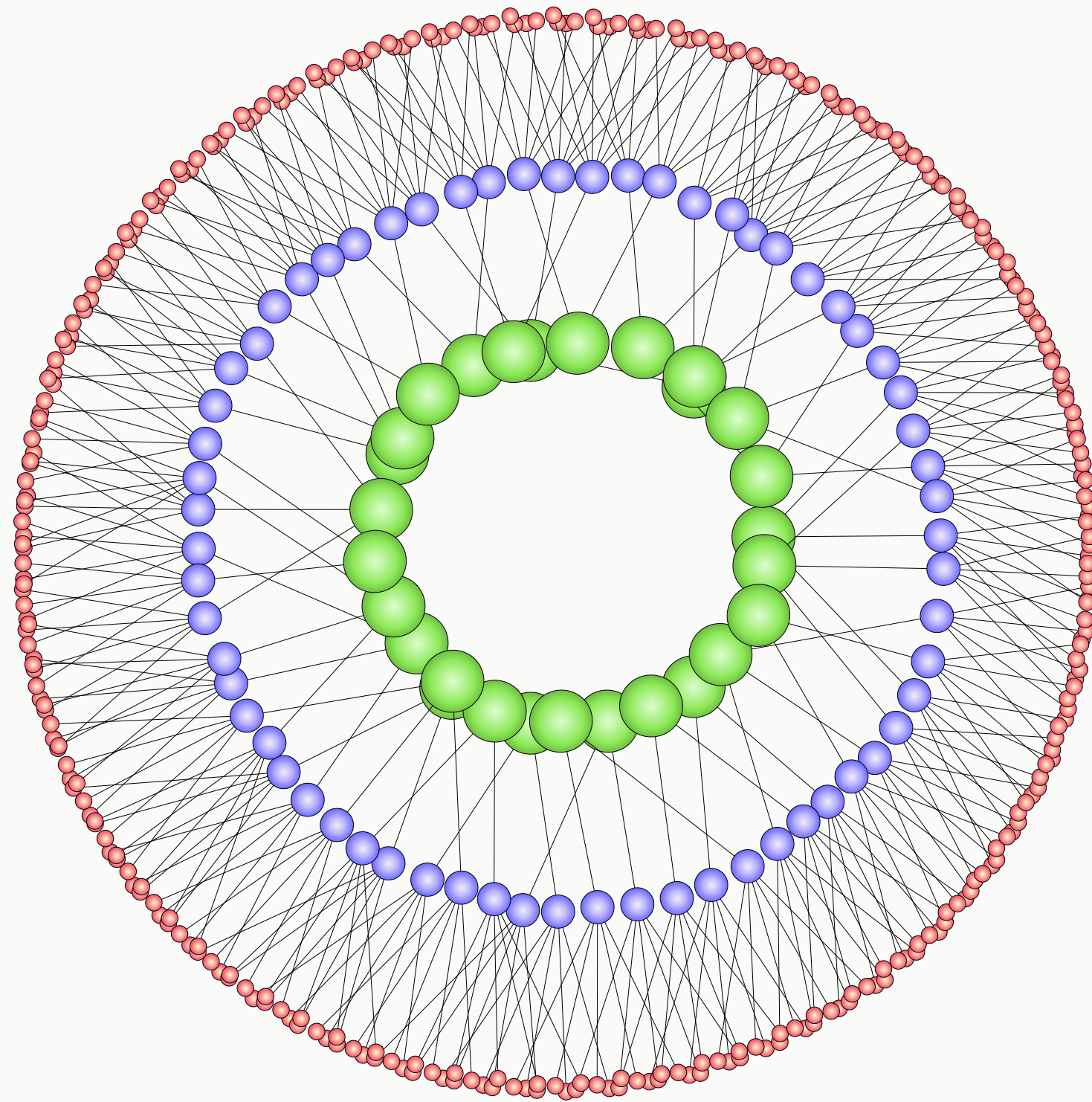
Webscale



Is it webscale?



When it Rains



Webscale!



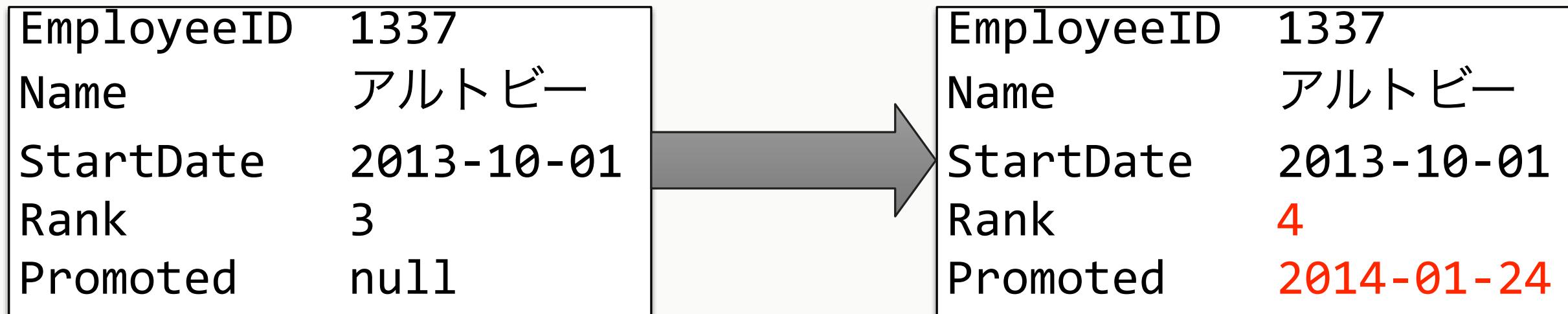
Theory & Practice

**In theory there is no difference
between theory and practice. In
practice there is.**

-Yogi Berra

Read-Modify-Write

```
UPDATE Employees SET Rank=4, Promoted=2014-01-24  
WHERE EmployeeID=1337;
```

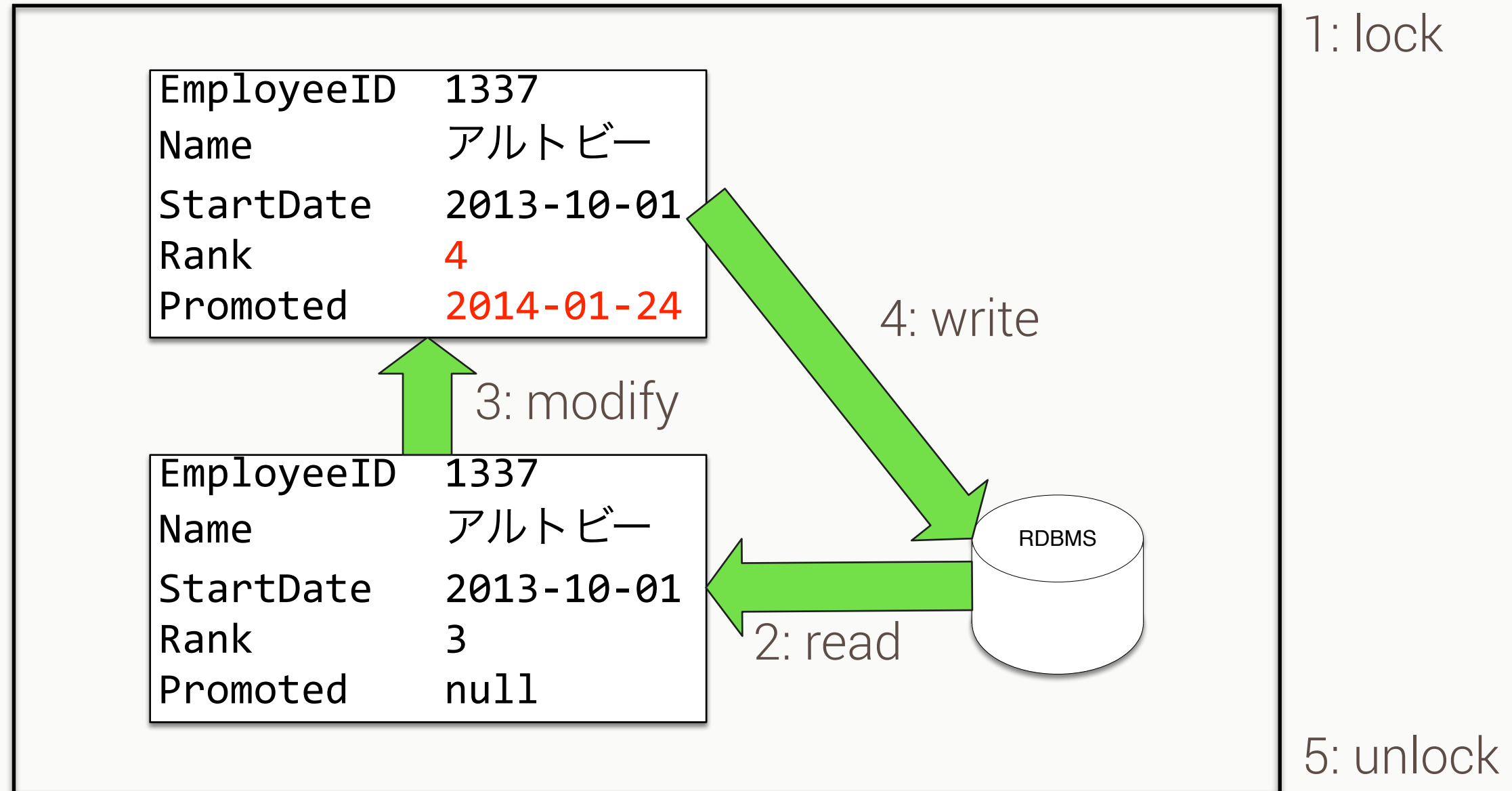


Read-Modify-Write



Read-Modify-Write

```
UPDATE Employees SET Rank=4, Promoted=2014-01-24  
WHERE EmployeeID=1337;
```



TNSTAAFL

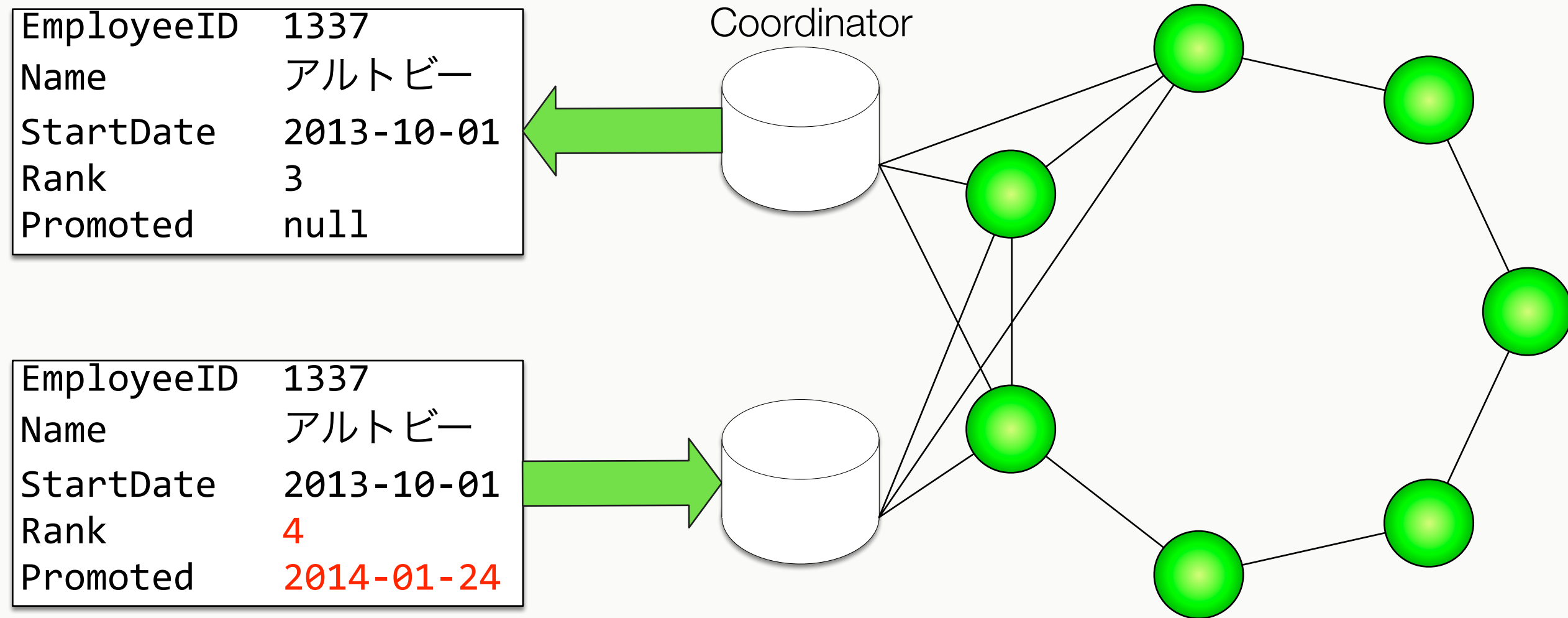
無償の昼食なんてものではありません

Safety



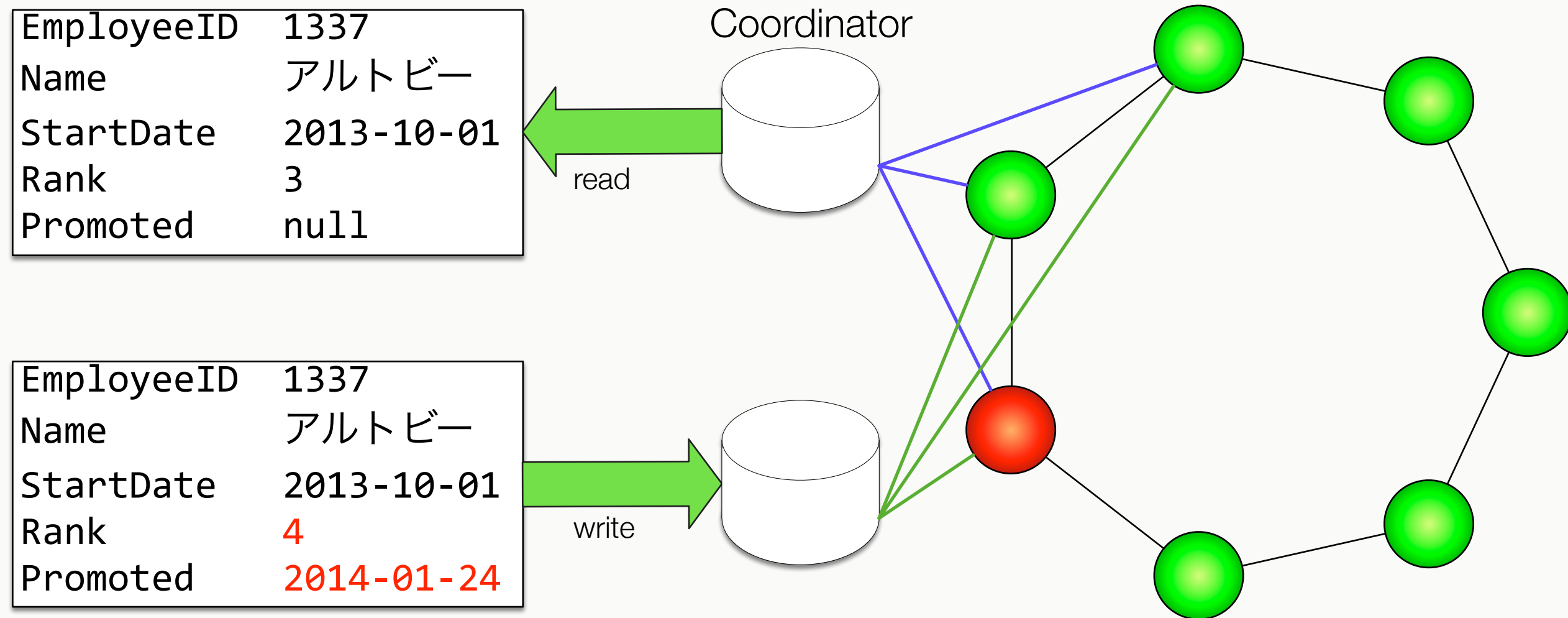
Eventual Consistency

```
UPDATE Employees SET Rank=4, Promoted=2014-01-24  
WHERE EmployeeID=1337;
```



DATASTAX

```
UPDATE Employees SET Rank=4, Promoted=2014-01-24  
WHERE EmployeeID=1337;
```



Overwriting

```
CREATE TABLE host_lookup (  
    name    varchar,  
    id      uuid,  
    PRIMARY KEY(name)  
);
```

```
INSERT INTO host_uuid (name,id) VALUES  
("www.tobert.org", "463b03ec-fcc1-4428-bac8-80cccee1c2f77");
```

```
INSERT INTO host_uuid (name,id) VALUES  
("tobert.org",      "463b03ec-fcc1-4428-bac8-80cccee1c2f77");
```

```
INSERT INTO host_uuid (name,id) VALUES  
("www.tobert.org", "463b03ec-fcc1-4428-bac8-80cccee1c2f77");
```

```
SELECT id FROM host_lookup WHERE name="tobert.org";
```


Key/Value

```
CREATE TABLE keyval (  
    key VARCHAR,  
    value blob,  
    PRIMARY KEY(key)  
);
```

```
INSERT INTO keyval (key,value) VALUES (?, ?);
```

```
SELECT value FROM keyval WHERE key=?;
```

Journaling / Logging / Time-series

```
CREATE TABLE tsdb (  
    time_bucket timestamp,  
    time          timestamp,  
    value         blob,  
    PRIMARY KEY(time_bucket, time)  
);
```

```
INSERT INTO tsdb (time_bucket, time, value) VALUES (  
    "2014-10-24",          -- 1-day bucket (UTC)  
    "2014-10-24T12:12:12Z", -- ALWAYS USE UTC  
    '{ "foo": "bar" }'  
);
```


Journaling / Logging / Time-series

2014-01-24	2014-01-24T12:12:12Z	2014-01-24T21:21:21Z
	{“key”: “value”}	{“key”: “value”}
2014-01-25	2014-01-25T13:13:13Z	
	{“key”: “value”}	

```
{ “2014-01-24” => {  
  “2014-01-24T12:12:12Z” => {  
    {“foo”: “bar”}  
  }  
}
```

Content Addressable Storage

```
CREATE TABLE objects (  
    cid      varchar,  
    content  blob,  
    PRIMARY KEY(cid)  
);
```

```
INSERT INTO objects (cid,content) VALUES (?, ?);
```

```
SELECT content FROM objects WHERE cid=?;
```


Content Addressable Storage

```
require 'cql'
require 'digest/sha1'

dbh = Cql::Client.connect(hosts: ['127.0.0.1'])
dbh.use('cas')

data = { :timestamp => 1390436043, :value => 1234 }

cid = Digest::SHA1.new.digest(data.to_s).unpack('H*')

sth = dbh.prepare(
  'SELECT content FROM objects WHERE cid=?')

sth.execute(root_cid).first['content']
```

Safety



Lightweight Transactions

- Cassandra 2.0 and on support LWT based on PAXOS
- PAXOS is a distributed consensus protocol
- Given a constraint, Cassandra ensures correct ordering

Cassandra Collections

```
CREATE TABLE posts (  
  id          uuid,  
  body        varchar,  
  created     timestamp,  
  authors     set<varchar>,  
  tags        set<varchar>,  
  PRIMARY KEY(id)  
);
```

```
INSERT INTO posts (id,body,created,authors,tags) VALUES (  
  ea4aba7d-9344-4d08-8ca5-873aa1214068,  
  ‘アルトビーの犬はばかね’,  
  ‘now’,  
  [‘アルトビー’, ‘ィオートビー’],  
  [‘dog’, ‘silly’, ‘犬’, ‘ばか’]  
);
```



Cassandra Collections

```
CREATE TABLE metrics (  
    bucket    timestamp,  
    time      timestamp,  
    value     blob,  
    labels    map<varchar,varchar>,  
    PRIMARY KEY(bucket)  
);
```

Lightweight Transactions

```
UPDATE users
  SET username='tobert'
  WHERE id=68021e8a-9eb0-436c-8cdd-aac629788383
     IF username='renice';

INSERT INTO users (id, username)
VALUES (68021e8a-9eb0-436c-8cdd-aac629788383, 'renice')
IF NOT EXISTS;
```


CASSANDRA SUMMIT 2014

September 10 – 11

THE LARGEST GATHERING OF CASSANDRA USERS ON THE PLANET.

SEPTEMBER 10 - 11 | SAN FRANCISCO | www.Datastax.com/CassandraSummit14

THE #1 GLOBAL CASSANDRA EVENT

FREE ADMISSION.
SERIOUSLY.

Conclusion

- The data model is part of the architecture.
- Questions?
 - Come see me at the Datastax booth or mention @ALTobey on Twitter