

BEKK

*PATTERNS FOR
KEY-VALUE STORES*

*Jfokus 2013
Ole-Martin Mørk
06/02/13*

ABOUT ME

Ole-Martin Mørk
Scientist

Bekk Consulting AS
Oslo, Norway

twitter: olemartin



AGENDA



INTRODUCTION

DIFFERENT KV STORES

GENERAL PATTERNS

RIAK PATTERNS

REDIS PATTERNS

RIAK+REDIS PATTERNS

SUMMARY

KEY	VALUE
session_1	Functional Programming You Already Know
session_2	Fra enkel J2SE til Grid computing med GigaSpaces XAP
session_3	Reinventing software quality
session_4	Taming Java Agents
session_5	Memory Leaks in Java
session_6	Linking data without common identifiers
session_7	Avinstaller Java nå!

PUT

GET

DEL

KEY
VALUE
STORE

```
select value  
from Lookup  
where key=?
```

Speed

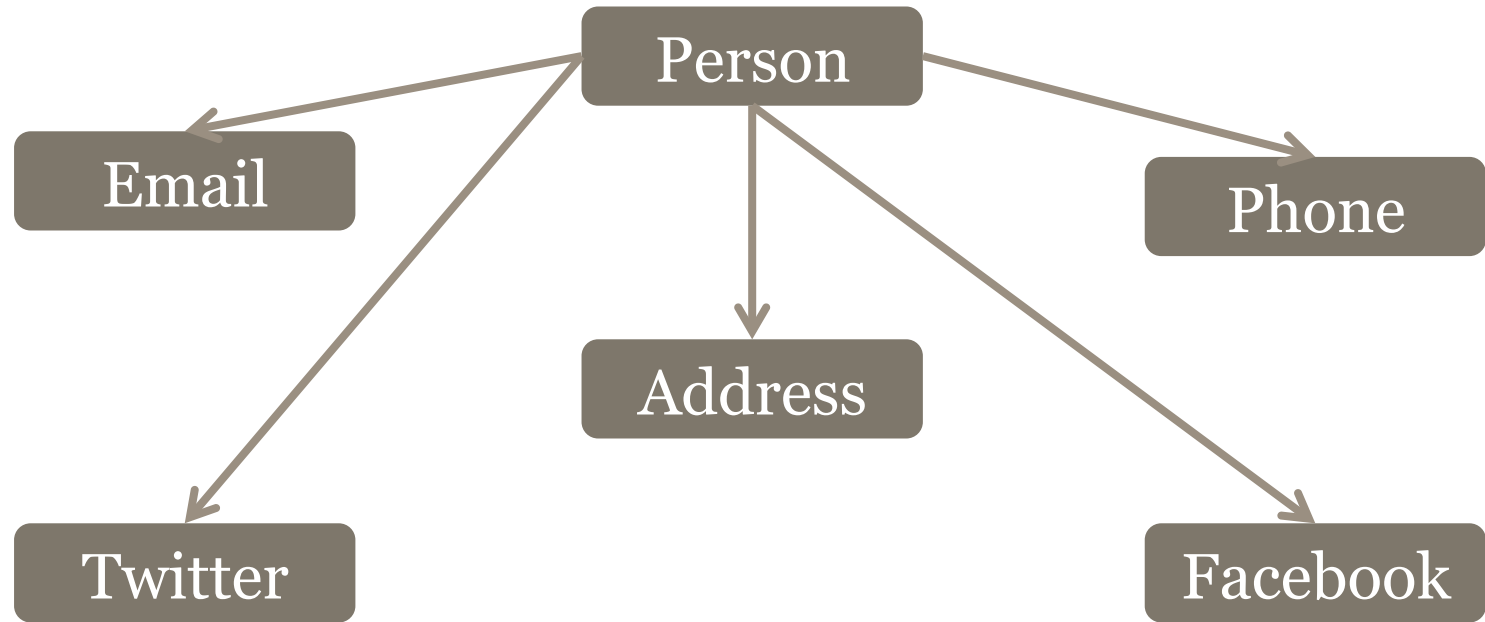




Distribution

When all you
have is a hammer..





name: "Ole-Martin Mørk",

emails: email: "me@ole - martin.net"

type: "work", email: "ole-martin.mork@bekk.no"

phone: type: "mobile", phone: "+4790252338"

addresses:

type: "home"

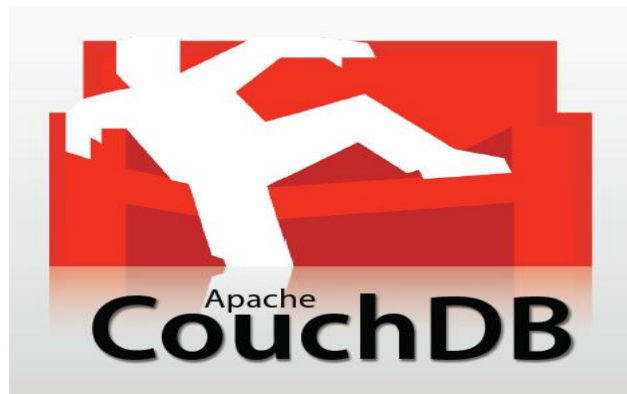
adresse: street: "Lundliveien 30b", town: "Oslo"

type: "work"

address: town: "Oslo", zipcode: "0150"

```
{name: "Ole-Martin Mørk",  
  emails: [ { email: "me@ole - martin.net" },  
            { type: "work", email: "ole-martin.mork@bekk.no" } ],  
  phone: [{ type: "mobile", phone: "+4790252338" } ],  
  addresses: [{  
    type: "home",  
    address: { street: "Lundliveien 30b", town: "Oslo" }  
  }, {  
    type: "work",  
    address: { town: "Oslo", zipcode: "0150"  
  }  
  } ] ] }
```

```
{name: "Ole-Martin Mørk",  
  emails: [ { email: "me@ole - martin.net" },  
            { type: "work", email: "ole-martin.mork@bekk.no" } ],  
  phone: [{ type: "mobile", phone: "+4790252338" } ],  
  addresses: [{  
    type: "home",  
    address: { street: "Lundliveien 30b", town: "Oslo" }  
  }, {  
    type: "work",  
    address: { town: "Oslo", zipcode: "0150" }  
  }  
]
```

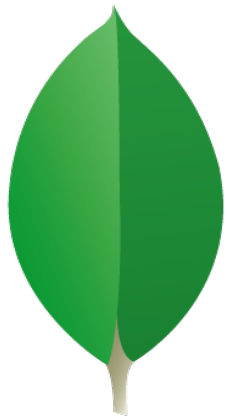


Cassandra



A P A C H E
HBASE

Demo



mongoDB

**GRIPPING
COMPULSIVELY WATCHABLE**

VANITY FAIR

**SUPERB
BRILLIANT & INTRIGUING**

JASON SOLOMONS, THE OBSERVER

HAYLEY ATWELL
COLIN FARRELL
SALLY HAWKINS
EWAN MCGREGOR
TOM WILKINSON

THE NEW THRILLER FROM THE DIRECTOR OF **MATCH POINT**

A **WOODY ALLEN** FILM

CASSANDRA'S DREAM

FAMILY IS FAMILY BLOOD IS BLOOD

WILD BUNCH PRESENTS AN IBERVILLE PRODUCTION "CASSANDRA'S DREAM"

HAYLEY ATWELL COLIN FARRELL SALLY HAWKINS EWAN MCGREGOR TOM WILKINSON COSTUME DESIGNER JULIET TAYLOR GAIL STEVENS EXECUTIVE PRODUCERS PATRICIA DICERITO MUSIC BY PHILIP GLASS EDITOR JILL TAYLOR

EXECUTIVE PRODUCER ALISA LEPSELTHER PRODUCED BY MARIA DJURKOVIC DIRECTOR OF PHOTOGRAPHY VILMOS ZSIGMOND EXECUTIVE PRODUCERS JACK ROLLINS CHARLES H. JOFFE PRODUCED BY VINCENT MARAWAL BRAHIM CHOUA DANIEL WUHRMANN

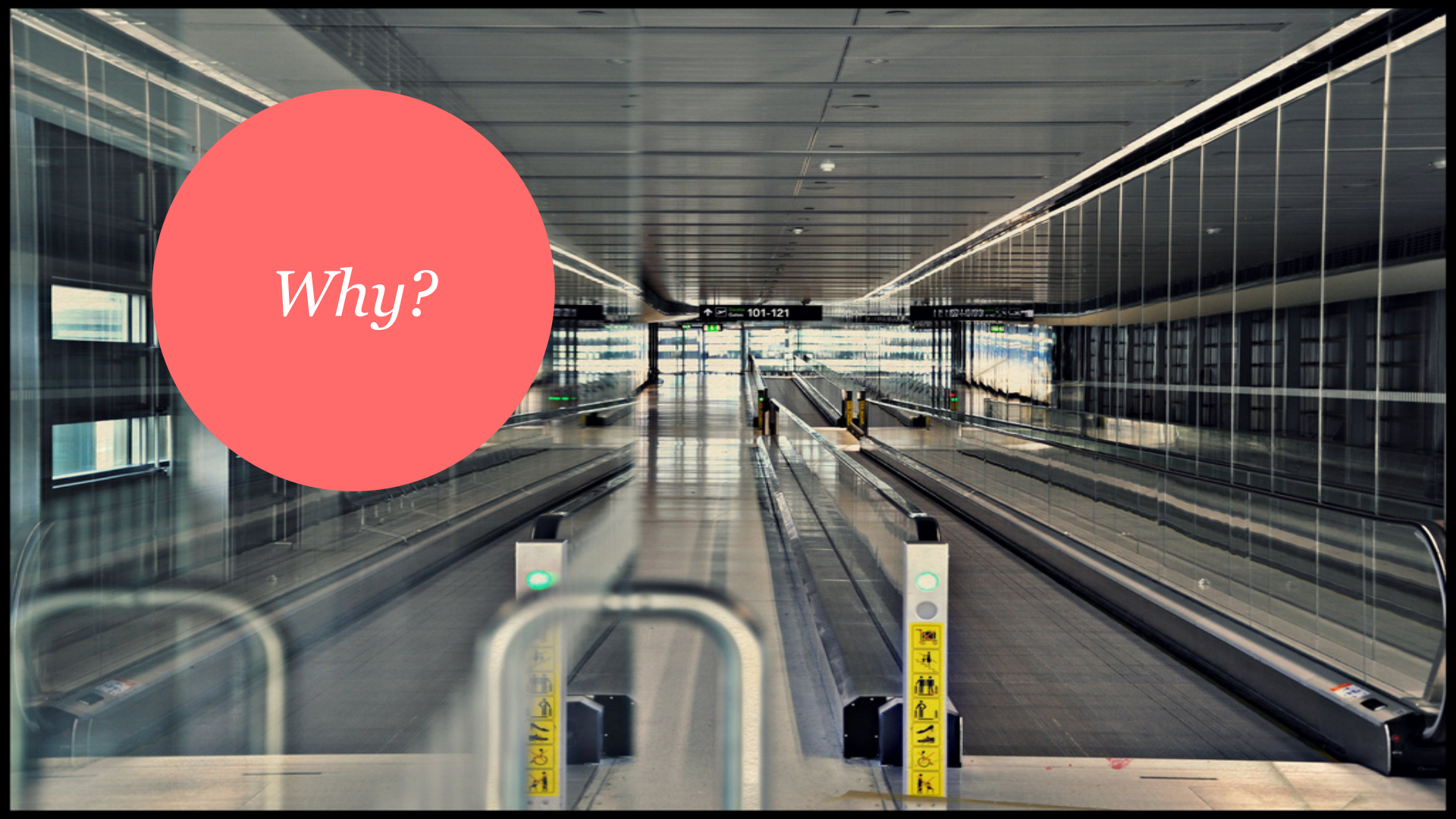
PRODUCED BY HELEN ROBIN NICKY KENTISH BARNES PRODUCED BY LETTY ARONSON STEPHEN TENENBAUM GARETH WILEY DIRECTED BY WOODY ALLEN

www.cassandrasdreammovie.co.uk

WILD BUNCH

OPTIMUM RELEASE

15 CONTAINS STRONG LANGUAGE AND SOME SEXUAL SCENES



Why?

The image shows a perspective view down a long, empty airport security checkpoint. Two parallel metal conveyor belts lead towards a glass-walled screening area in the distance. In the foreground, the tops of two metal detector arches are visible, each with a green light and a yellow sign with pictograms. A large, solid red circle is superimposed on the left side of the image, containing the word 'Why?' in a white, italicized serif font. The ceiling is a complex grid of lights and structural elements, and the floor is a polished, reflective material.

PATTERNS

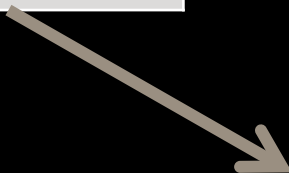
Keys



app:space:what:id

LOOKUPS ON OTHER DATA

user:id:1234



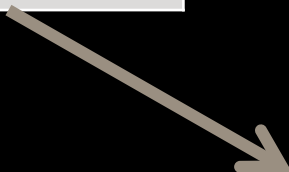
userId	1234
username	olemartin
email	olemartin@bekk.no
twitter	olemartin

user:twitter:olemartin

user:id:1234

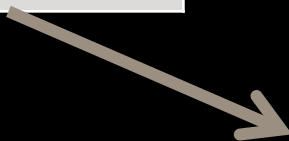
RELATIONS

user:id:1234



userId	1234
username	olemartin
email	olemartin@bekk.no
twitter	olemartin

user:id:4321



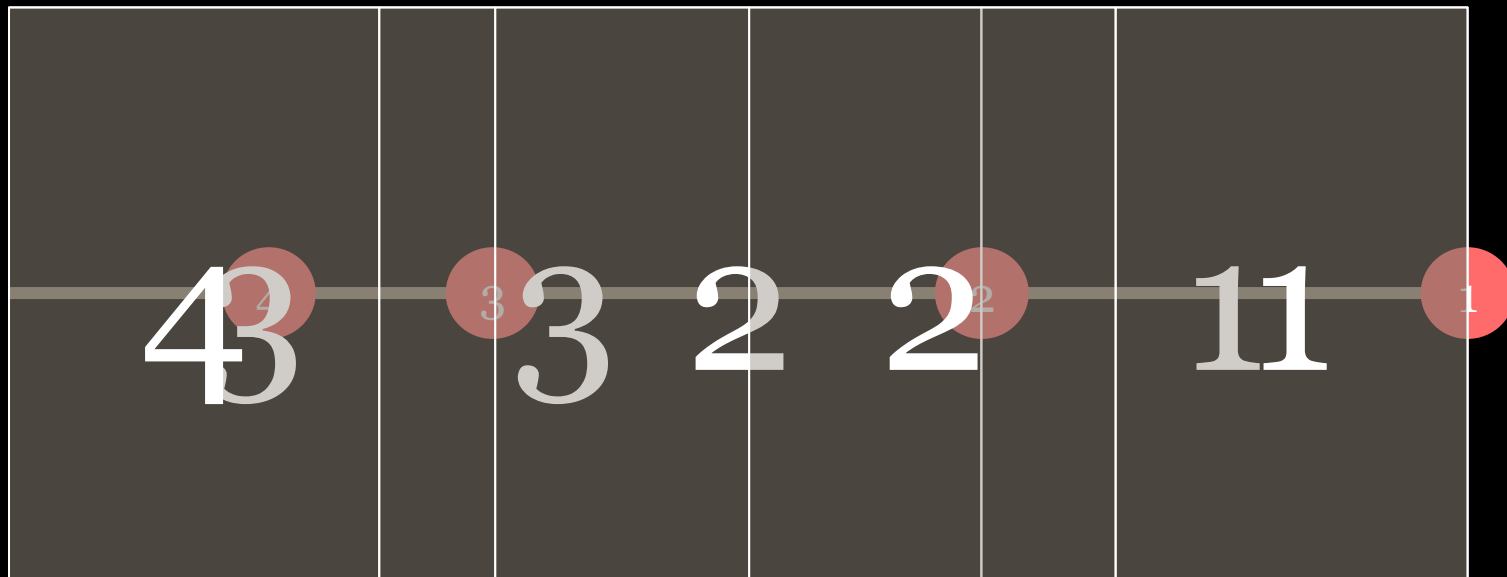
userId	4321
username	grolisbeth
brother_rel	user:id:1234



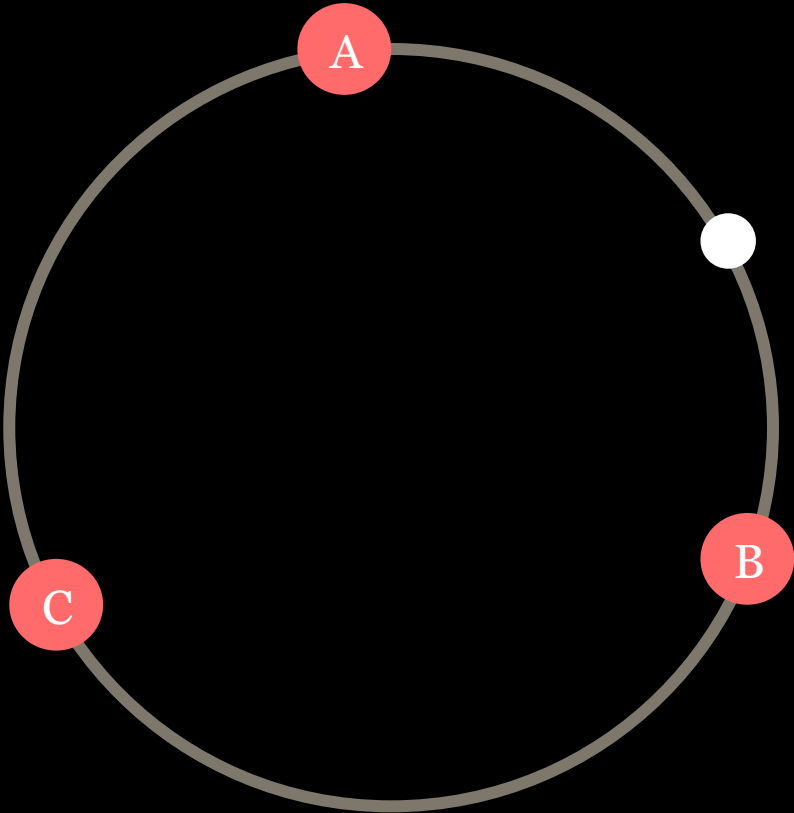
Hash

$$\langle \text{hash} \rangle \text{ modulo } \langle \text{num servers} \rangle = \text{serverId}$$

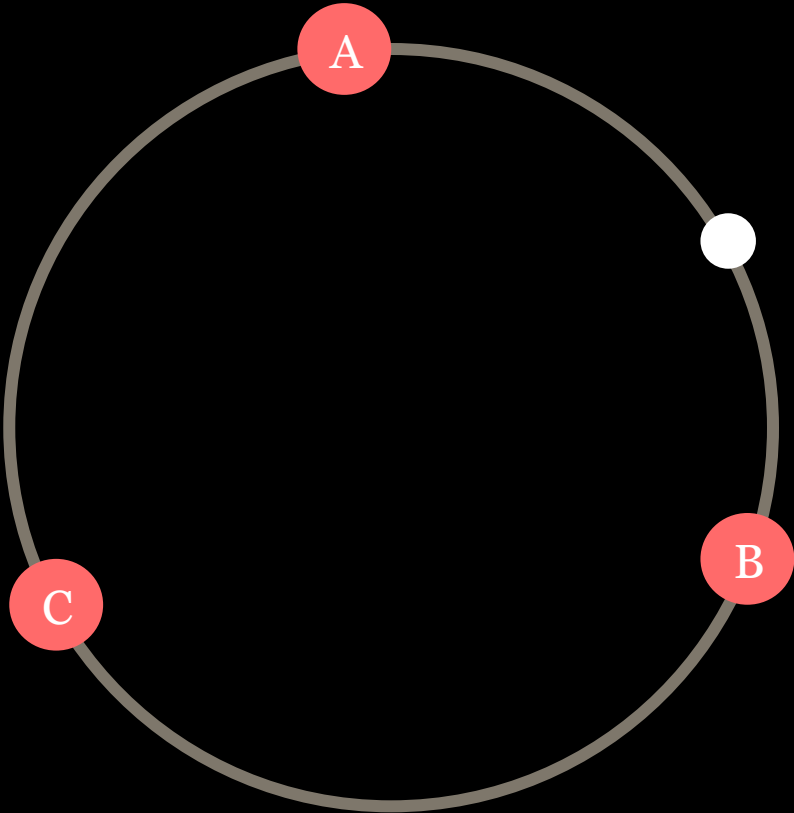
HASHING



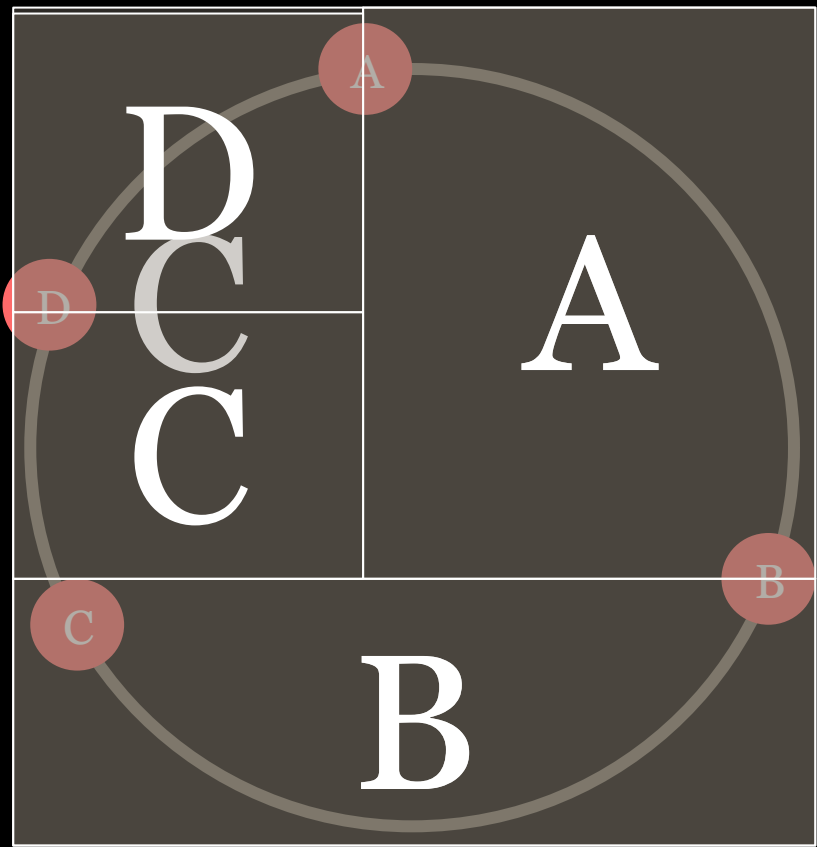
CONSISTENT HASHING



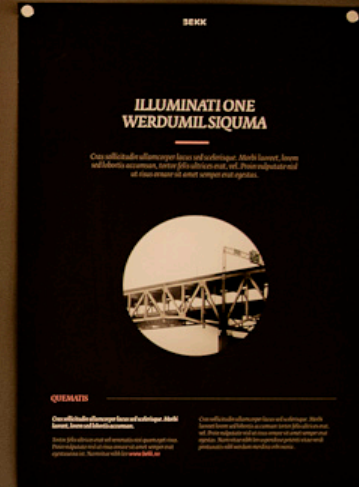
CONSISTENT HASHING



CONSISTENT HASHING



Use cases



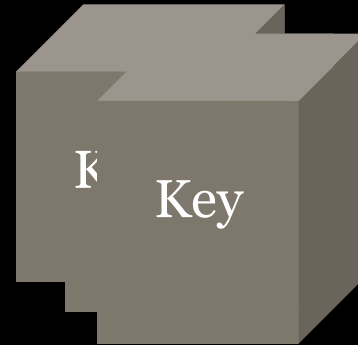
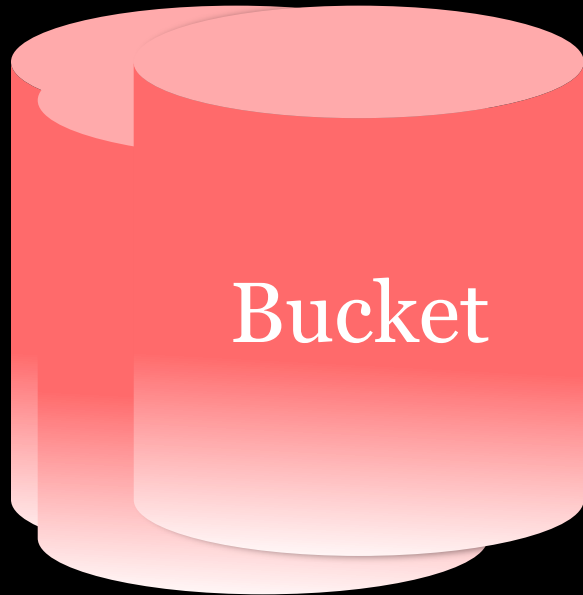
EXAMPLES ON APPLICATIONS SUITED FOR K/V

RDBMS + KEY/VALUE









Consistency

Availability

Partition

N W R

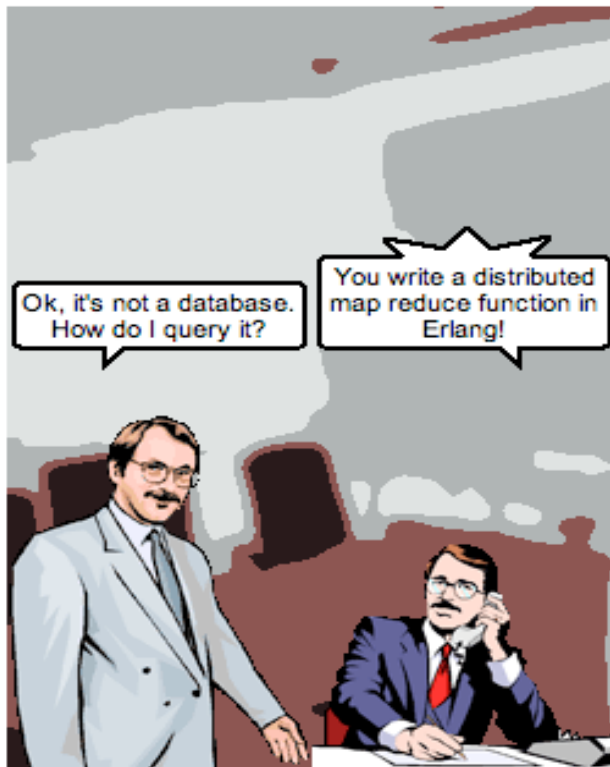
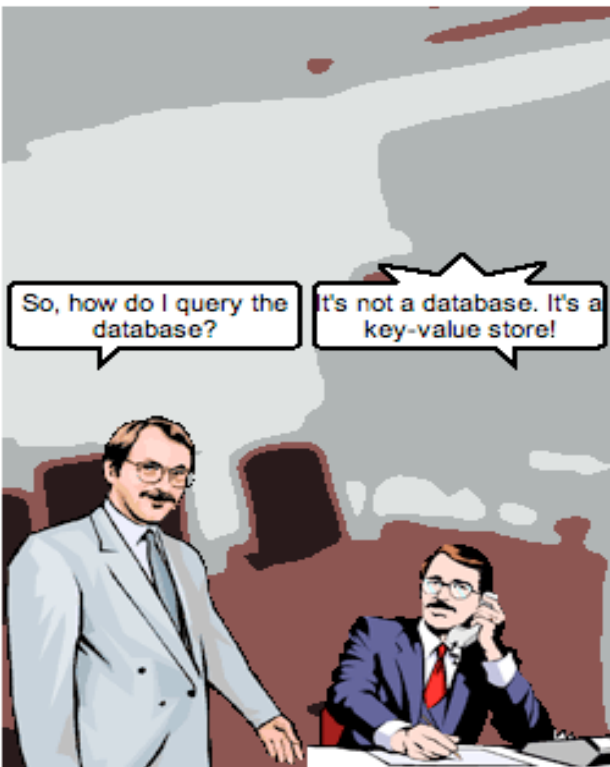
N > W

$\mathbb{N} > \mathbb{R}$



PATTERNS

Fault-tolerance



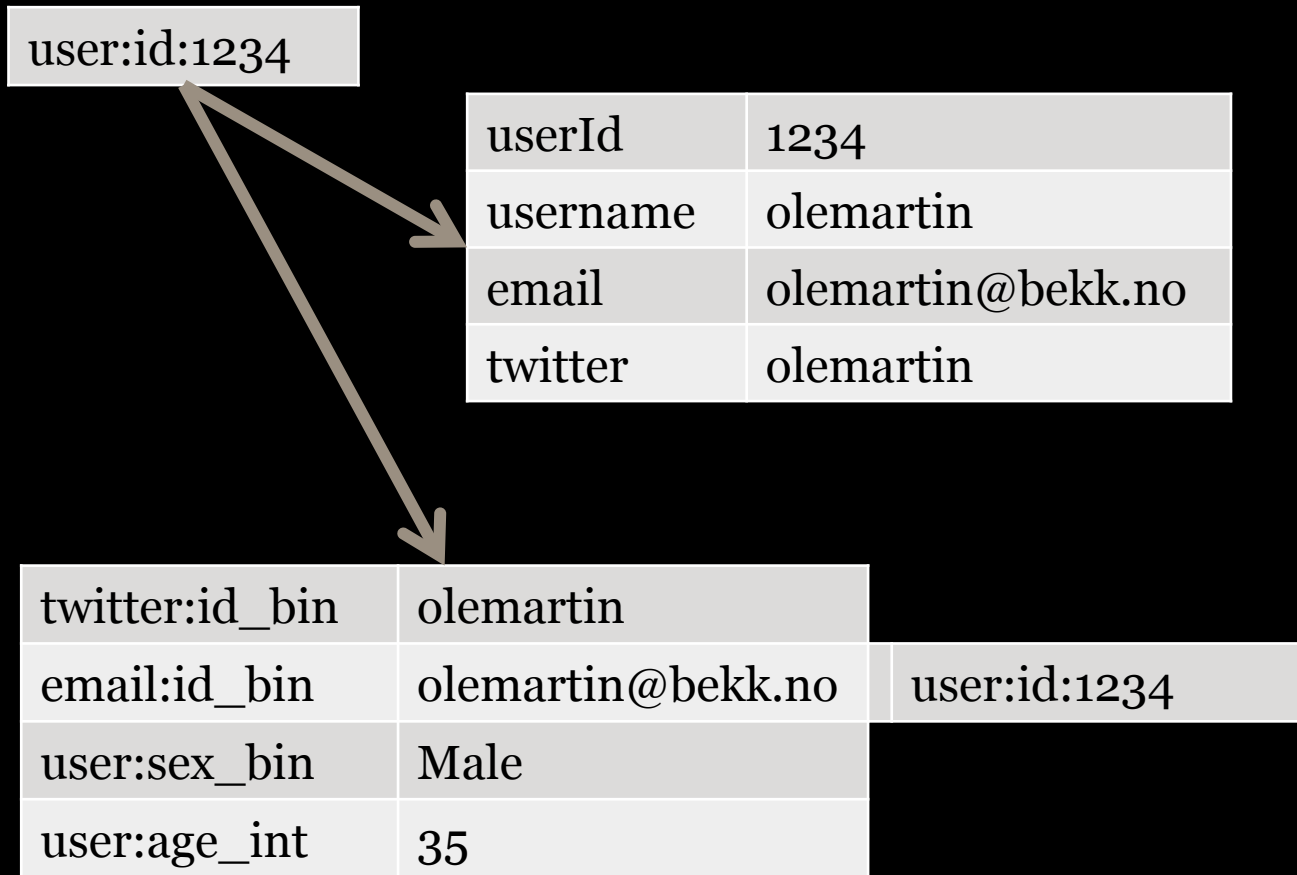
shopping:13092012:userId

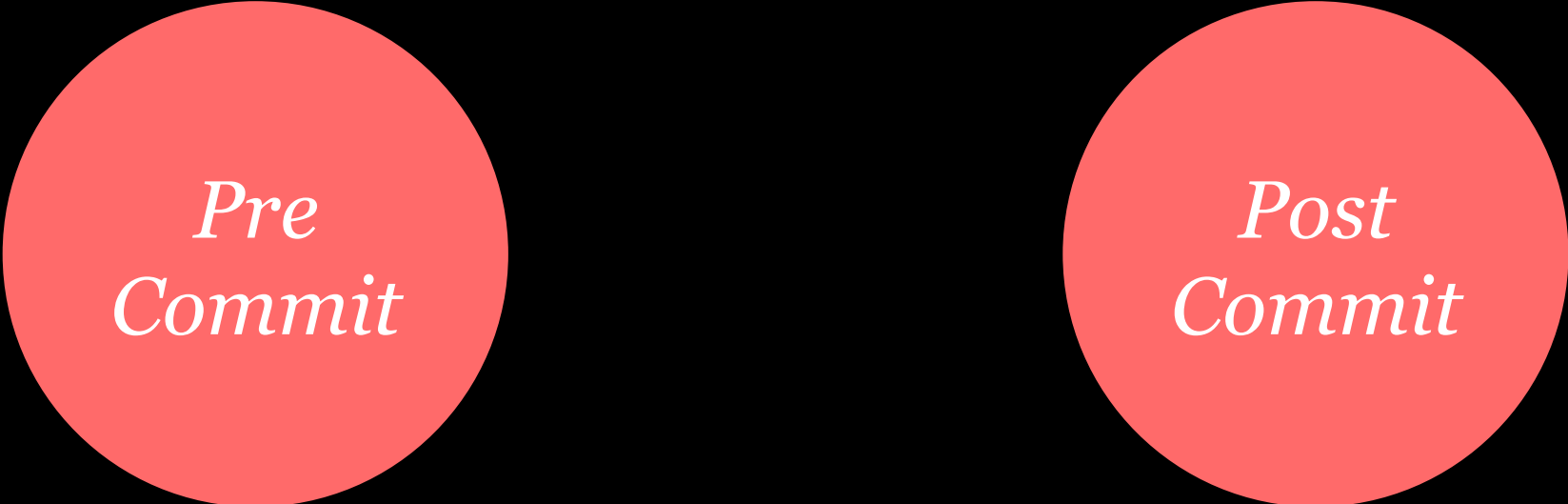
Secondary indexes



SECONDARY INDEXES

4
8





*Pre
Commit*

*Post
Commit*



redis

Hash

Sorted Sets

Lists

Sets

Strings

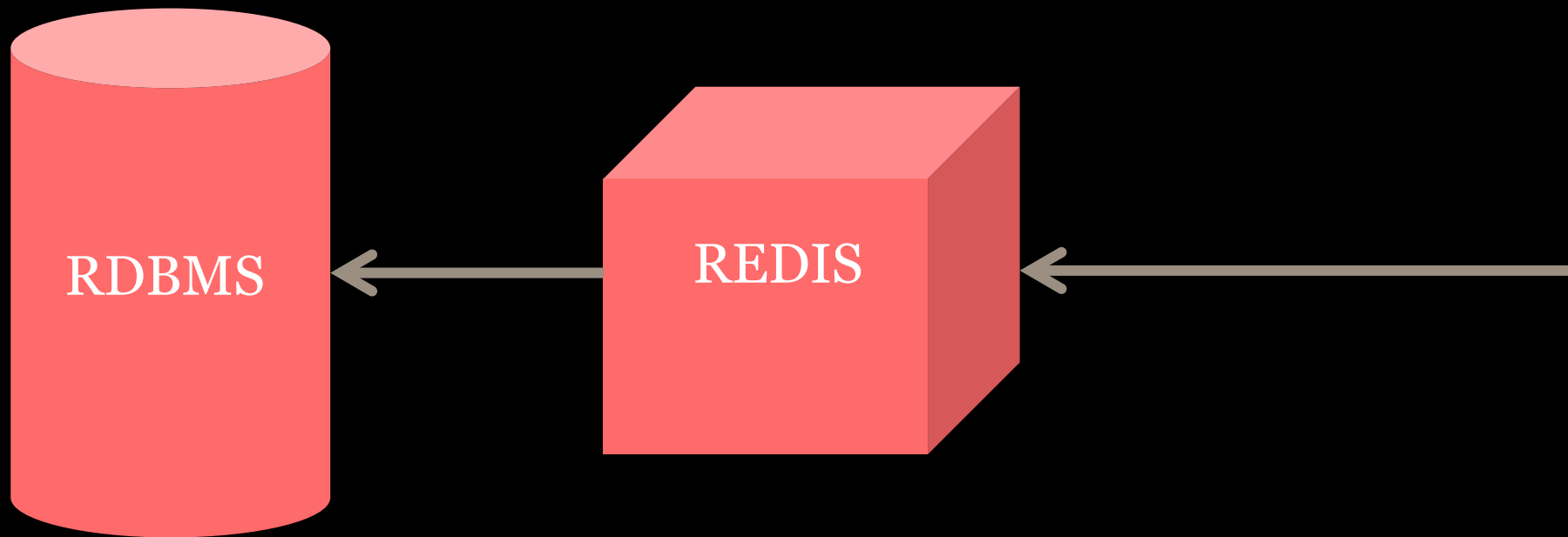
PATTERNS

SELECT * FROM foo

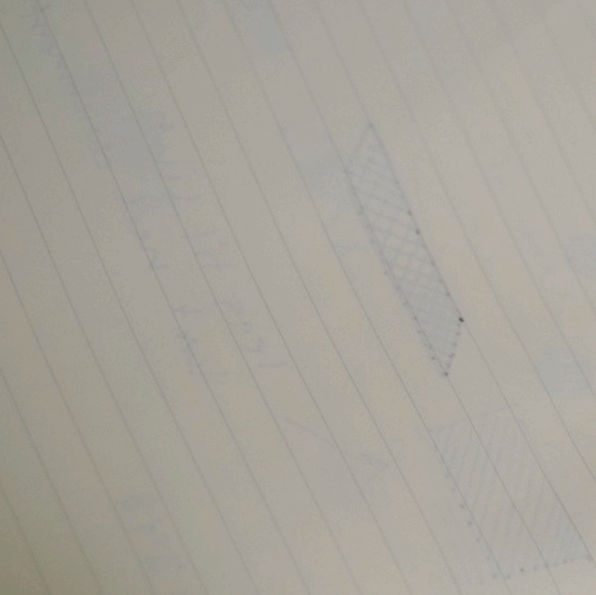
WHERE ...

ORDER BY time DESC LIMIT 10

PATTERNS FOR MILJØER MED RDBMS



Handwritten text at the top of the left page, partially obscured.



Farger: BEKK

Pantone:

- Rød - Red 032 U
- Gul - 401 U
- Grønn - 3265 U
- Mørk brun - 7554 U
- Mellom brun - 404 U
- Lys brun - 402 U
- Sort -
- ? m. slette de røde farger

CMYK:

Rød
gul
grønn

C	0	0	0
M	0	0	0
Y	0	0	0
K	0	0	0

Number of pages read by user

<pages:userId, count>

incr pages:userId

Most read today:

<most_read:300-2012, <url, count>, <url, count>>

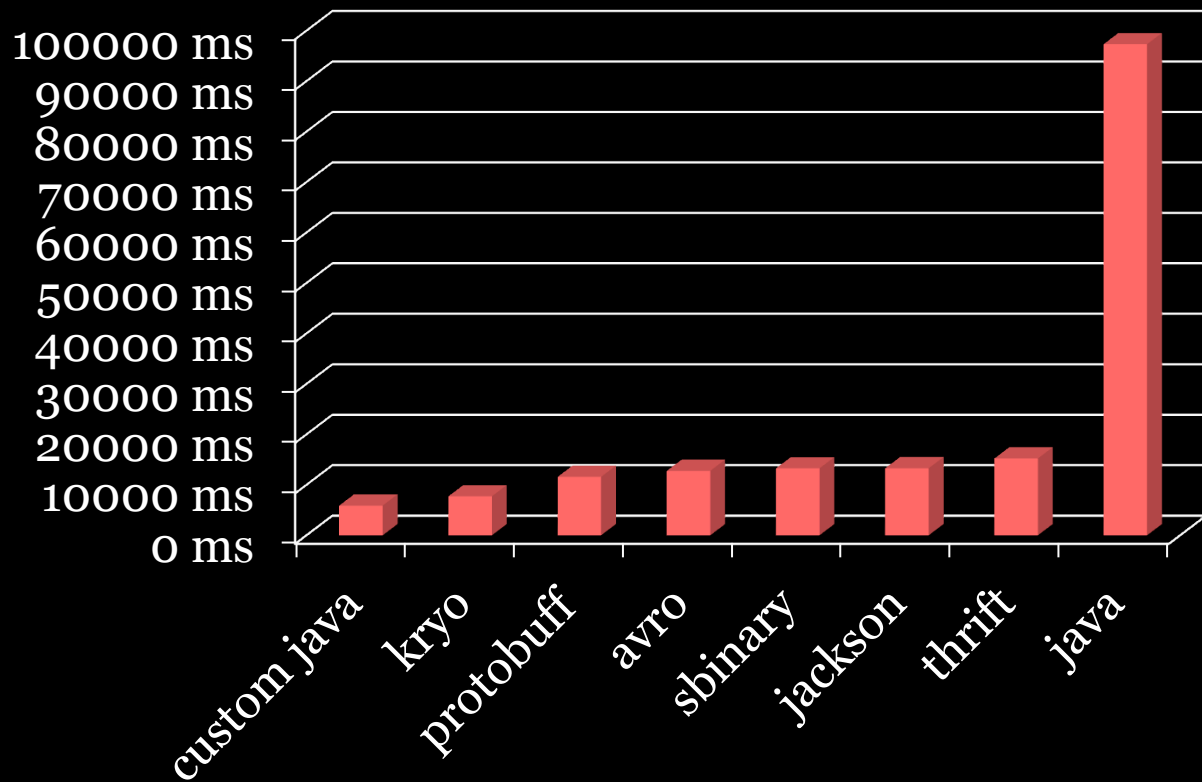
zrangebyscore most_read:300-2012 -inf +inf

Unique visitors:

<page:url:300-2012, <userId>, <userId>, <userId>>

scard page:url:300-2012

SERIALIZATION



KRYO IN REDIS

```
<bean id="redisTemplate"
      class="org.springframework.data.redis.core.RedisTemplate">
  <property name="connectionFactory" ref="connectionFactory"/>
  <property name="defaultSerializer" ref="serializer"/>
</bean>

<bean id="serializer" class="no.bekk.redis.DiggerSerializer"/>
```

KRYO

```
public class DiggerSerializer<T> implements RedisSerializer<T> {  
    private Kryo kryo = new Kryo();  
  
    public DiggerSerializer() {  
        kryo.register(Story.class);  
    }  
  
    public byte[] serialize(T t) {  
        return new ObjectBuffer(kryo).writeClassAndObject(t);  
    }  
  
    public T deserialize(byte[] bytes) {  
        return (T) new ObjectBuffer(kryo).readClassAndObject(bytes);  
    }  
}
```


REDDIT KLONE

1. story:id – holder styr på id-en til en story

2. story:id, value – holder selve story-objektet

3. story:stories zset – holder styr på antall stemmer

Stories

[Add new story](#)

Fullfartfoto foto rally

Submitted by olemartin

Fotos

(5 votes) [Vote!](#)

VG

Submitted by olemartin

VG er landets største avis

(4 votes) [Vote!](#)



redis

+



riak

=

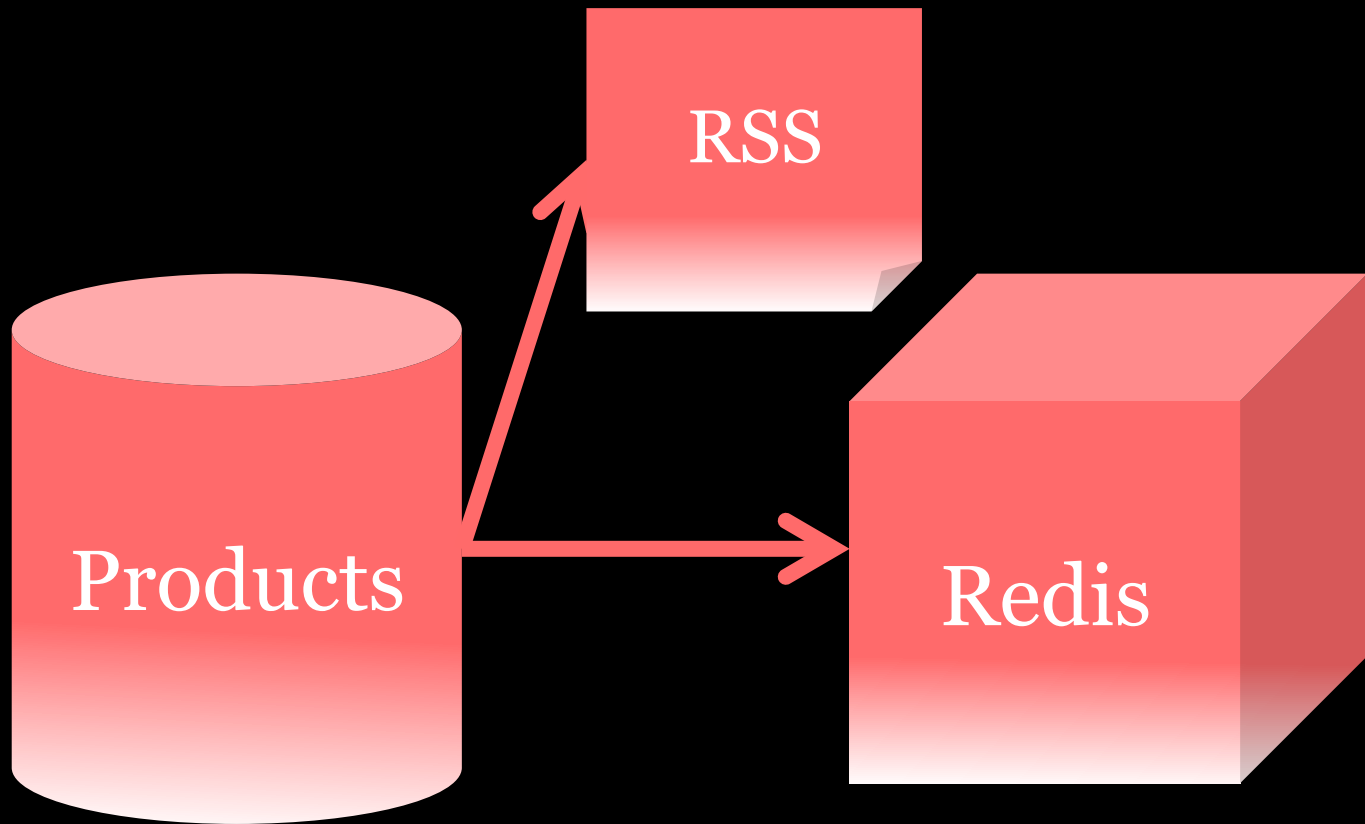


*Product
catalog*





POST COMMIT



REDIS

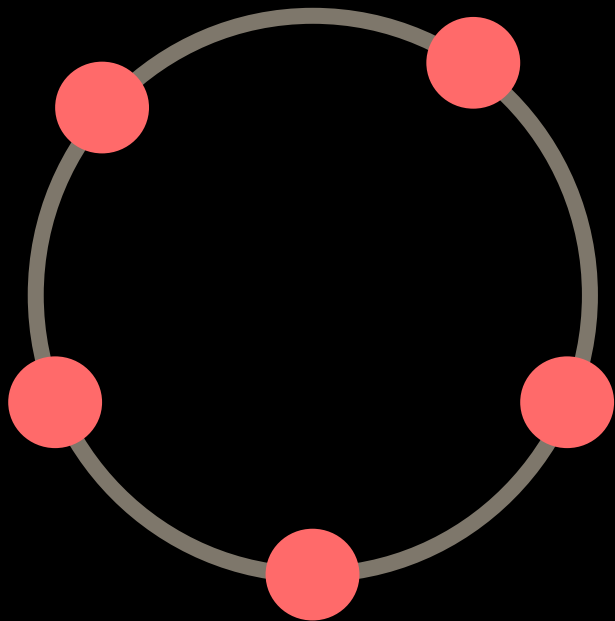
catalog:product:id	<i>binary data</i>
--------------------	--------------------

catalog:product:today:sold_zset	<i><productId, counter></i>
---------------------------------	-----------------------------------

catalog:product:week:sold_zset	<i><productId, counter></i>
--------------------------------	-----------------------------------

catalog:product:year:sold_zset	<i><productId, counter></i>
--------------------------------	-----------------------------------

RESULT





Summary

SUMMARY

1. Lots of different KV-stores are available
2. Good key names
3. Make sure you model your data correctly
4. Consistent hashin
5. Use KV-stores with RDBMS
6. Use KV-stores with data that match
7. Use the datastore where it excels
8. Combine different KV-stores
9. Be creative!



Questions?

BEKK

THANK YOU!

Ole-Martin Mørk
ole-martin.mork@bekk.no
@olemartin