

Couchbase



THE NO MARKETING BS INTRODUCTION TO COUCHBASE, YOUR FAVOURITE NOSQL DATABASE



Thursday, November 3, 11

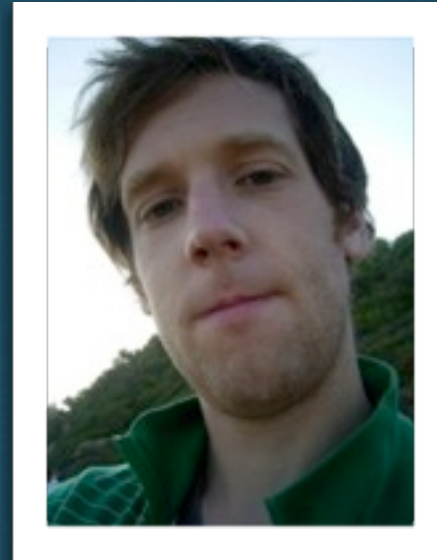
We're a sponsor, sponsor sessions are usually boring, I hate that, not here

YOUR HOST

Jan Lehnardt

jan@apache.org

@janl



Thursday, November 3, 11

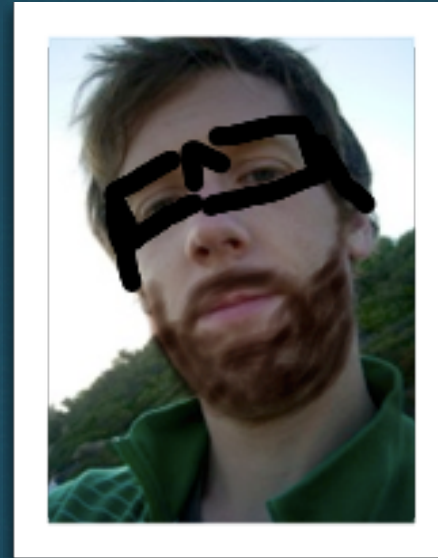
Get in touch any time, ask, interrupt, etc.

YOUR HOST

Jan Lehnardt

jan@apache.org

@janl



Thursday, November 3, 11

Get in touch any time, ask, interrupt, etc.

**MEMCACHED
+ MEMBASE
+ COUCHDB

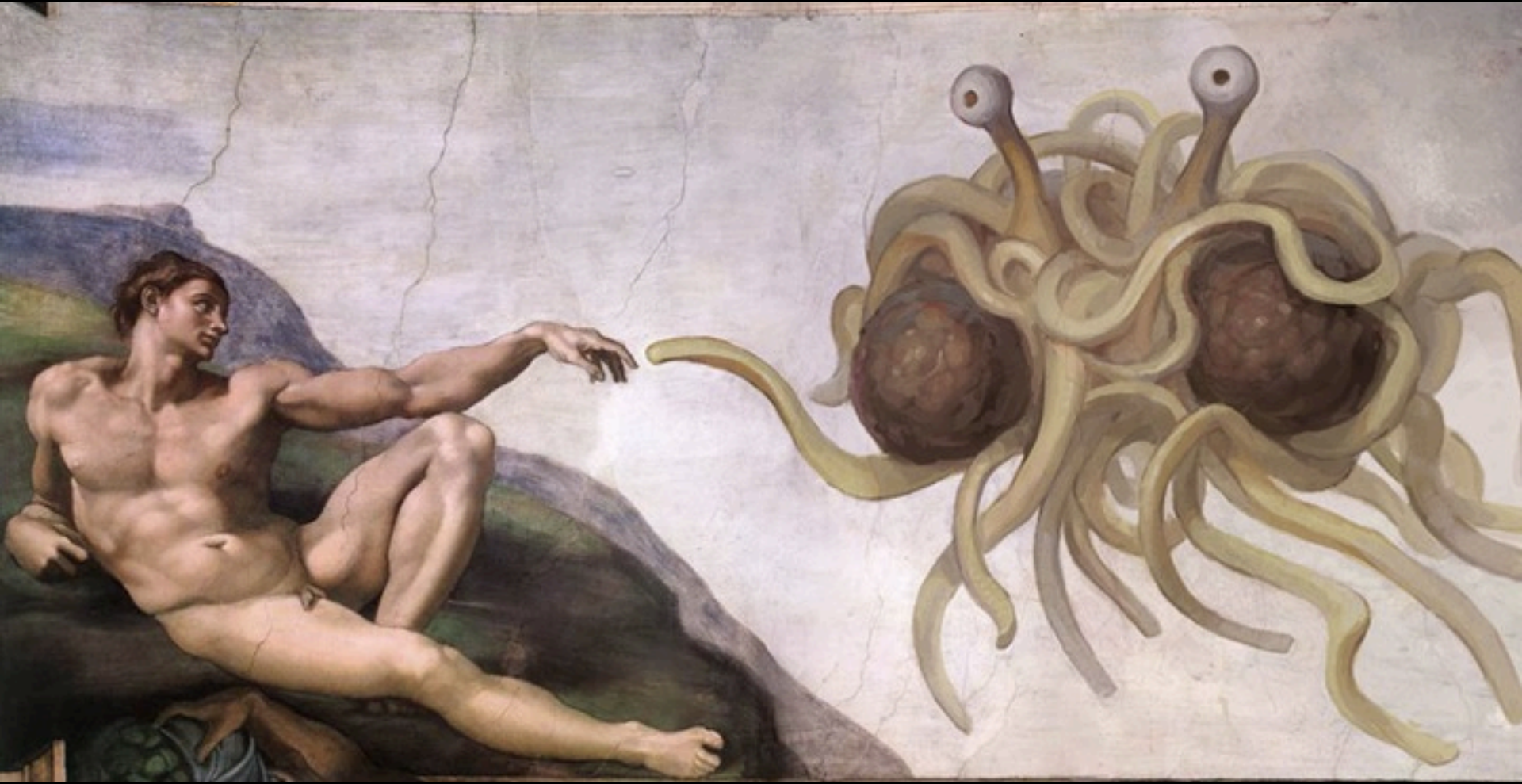
= COUCHBASE**



Thursday, November 3, 11

Understanding Couchbase by understanding its genesis.

?!



MEMCACHED



Thursday, November 3, 11

memcached makes chops of RAM

Gives the chops addresses

And makes them available to the network

MEMCACHED

RAM



Thursday, November 3, 11

memcached makes chops of RAM

Gives the chops addresses

And makes them available to the network

MEMCACHED



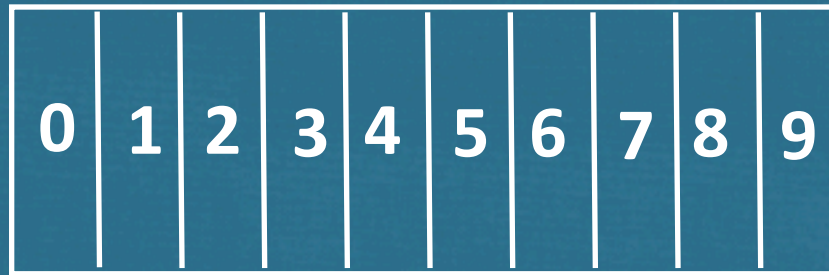
Thursday, November 3, 11

memcached makes chops of RAM

Gives the chops addresses

And makes them available to the network

MEMCACHED



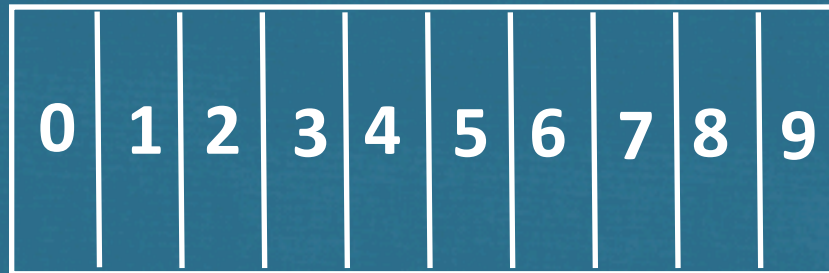
Thursday, November 3, 11

memcached makes chops of RAM

Gives the chops addresses

And makes them available to the network

MEMCACHED



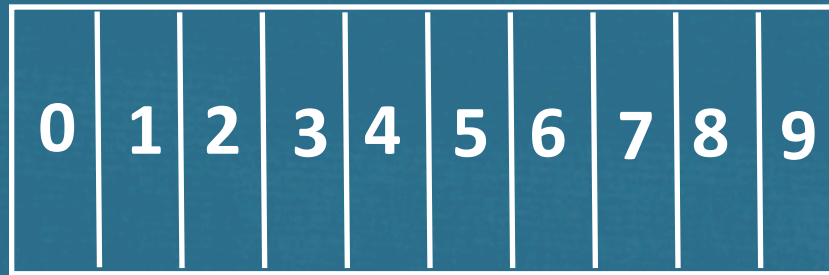
Thursday, November 3, 11

memcached makes chops of RAM

Gives the chops addresses

And makes them available to the network

MEMCACHED



**The
Network**



Thursday, November 3, 11

memcached makes chops of RAM
Gives the chops addresses
And makes them available to the network

MEMCACHED



7

Thursday, November 3, 11

DO NOT underestimate the simple

MEMCACHED

- Pro



7

Thursday, November 3, 11

DO NOT underestimate the simple

MEMCACHED

- Pro
 - Simple



7

Thursday, November 3, 11

DO NOT underestimate the simple

MEMCACHED

- Pro
 - Simple
 - Very fast



7

Thursday, November 3, 11

DO NOT underestimate the simple

MEMCACHED

- Pro
 - Simple
 - Very fast

- Contra



7

Thursday, November 3, 11

DO NOT underestimate the simple

MEMCACHED

- Pro
 - Simple
 - Very fast
- Contra
 - Not very flexible



7

Thursday, November 3, 11

DO NOT underestimate the simple

MEMCACHED

- Pro
 - Simple
 - Very fast
- Contra
 - Not very flexible
 - No querying



7

Thursday, November 3, 11

DO NOT underestimate the simple

MEMCACHED

- Pro
 - Simple
 - Very fast
- Contra
 - Not very flexible
 - No querying
 - Memory bound



7

Thursday, November 3, 11

DO NOT underestimate the simple

MEMCACHED

- Pro
 - Simple
 - Very fast
- Contra
 - Not very flexible
 - No querying
 - Memory bound
 - No persistence



7

Thursday, November 3, 11

DO NOT underestimate the simple

MEMCACHED

- Pro

- Simple
- Very fast

Do Not Underestimate

- Contra

- Not very flexible
- No querying
- Memory bound
- No persistence



7

Thursday, November 3, 11

DO NOT underestimate the simple

MEMCACHED

- Pro
 - Simple
 - Very fast

Do Not Underestimate

Simple



7

Thursday, November 3, 11

DO NOT underestimate the simple

**“ALL PROBLEMS IN COMPUTER
SCIENCE CAN BE SOLVED BY
ANOTHER LEVEL OF INDIRECTION”**

— “David Wheeler”

“...except for the problem of too many layers of indirection.”

— Kevlin Henney



Thursday, November 3, 11

David Wheeler

Kevlin Henney

according to Wikipedia

MEMBASE 1



9

Thursday, November 3, 11

**Membase collects memcacheds,
gives them addresses,
and makes them available to the network AS ONE.**

MEMBASE 1



Thursday, November 3, 11

**Membase collects memcacheds,
gives them addresses,
and makes them available to the network AS ONE.**

MEMBASE 1



9

Thursday, November 3, 11

**Membase collects memcacheds,
gives them addresses,
and makes them available to the network AS ONE.**

MEMBASE 1

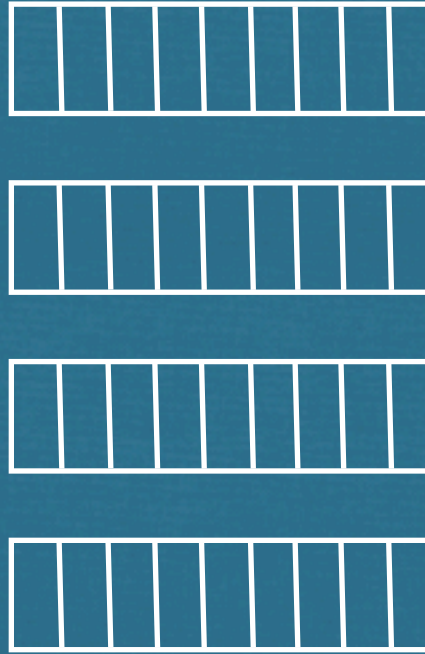


9

Thursday, November 3, 11

**Membase collects memcacheds,
gives them addresses,
and makes them available to the network AS ONE.**

MEMBASE 1

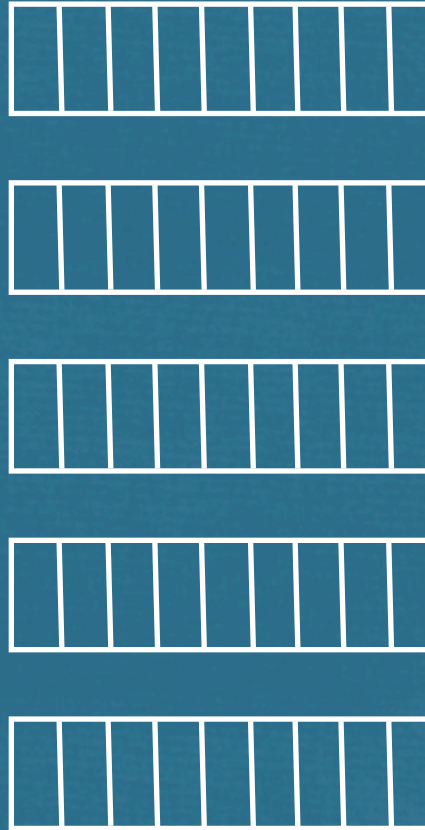


9

Thursday, November 3, 11

**Membase collects memcacheds,
gives them addresses,
and makes them available to the network AS ONE.**

MEMBASE 1

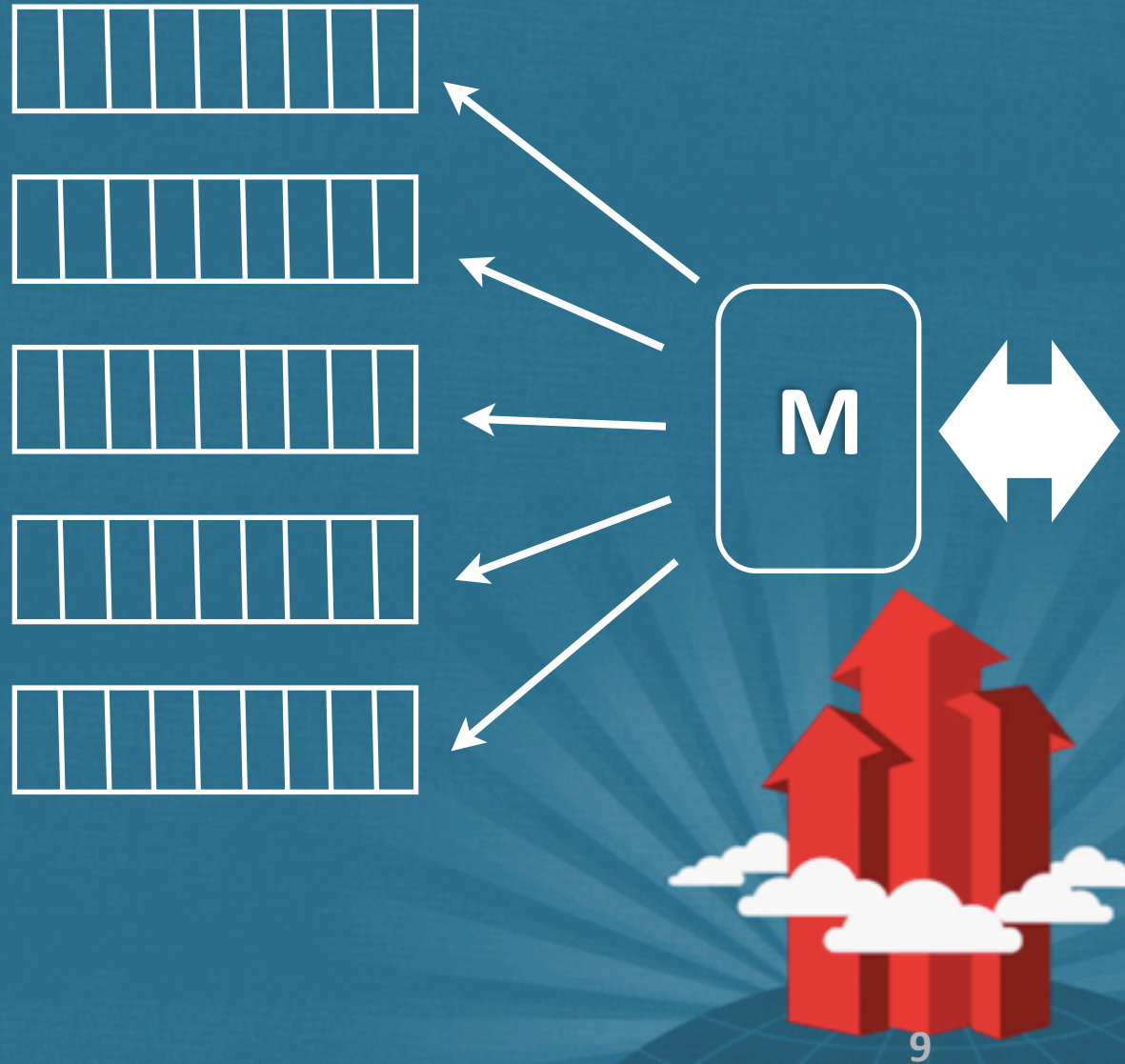


9

Thursday, November 3, 11

**Membase collects memcacheds,
gives them addresses,
and makes them available to the network AS ONE.**

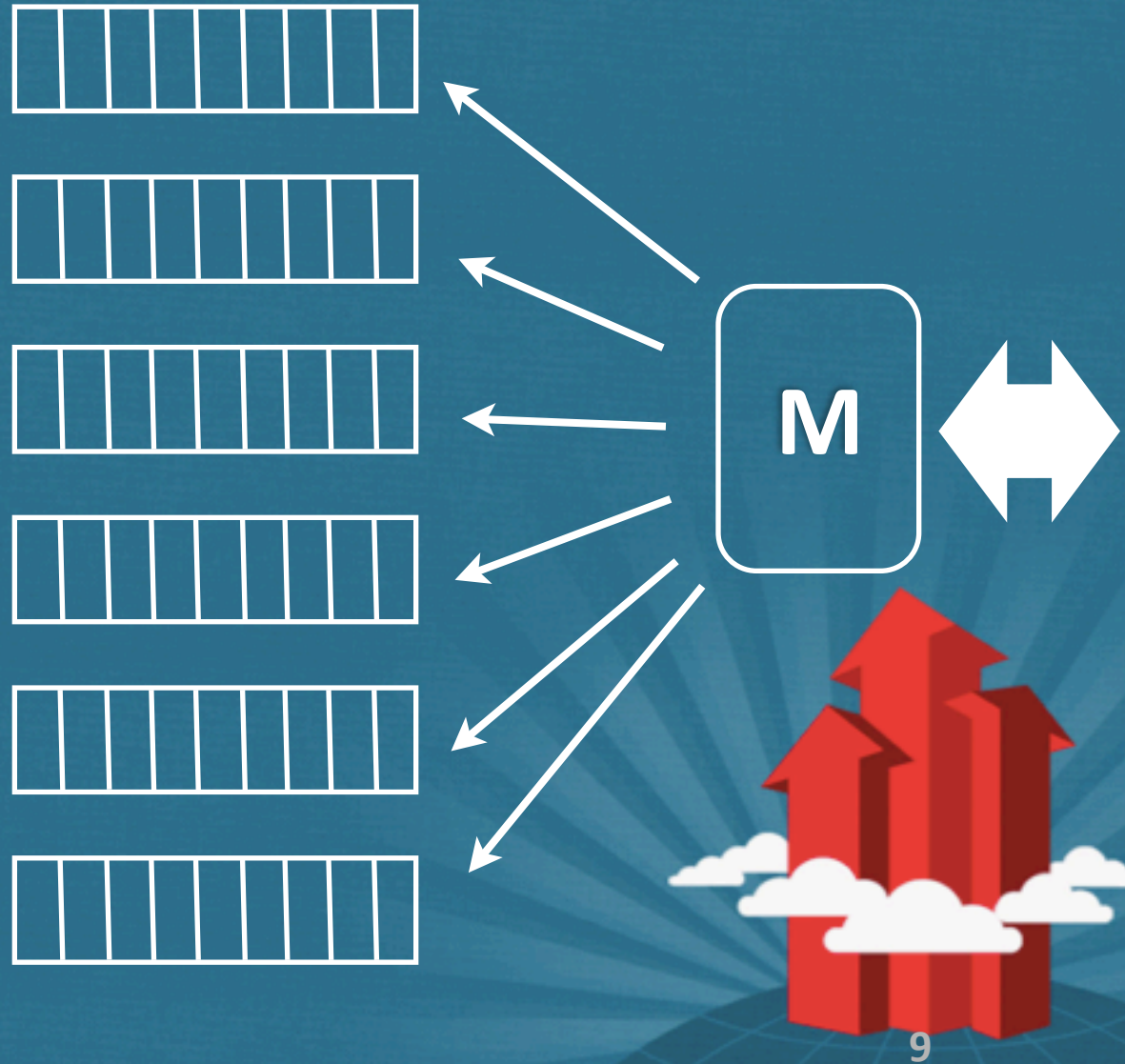
MEMBASE 1



Thursday, November 3, 11

**Membase collects memcacheds,
gives them addresses,
and makes them available to the network AS ONE.**

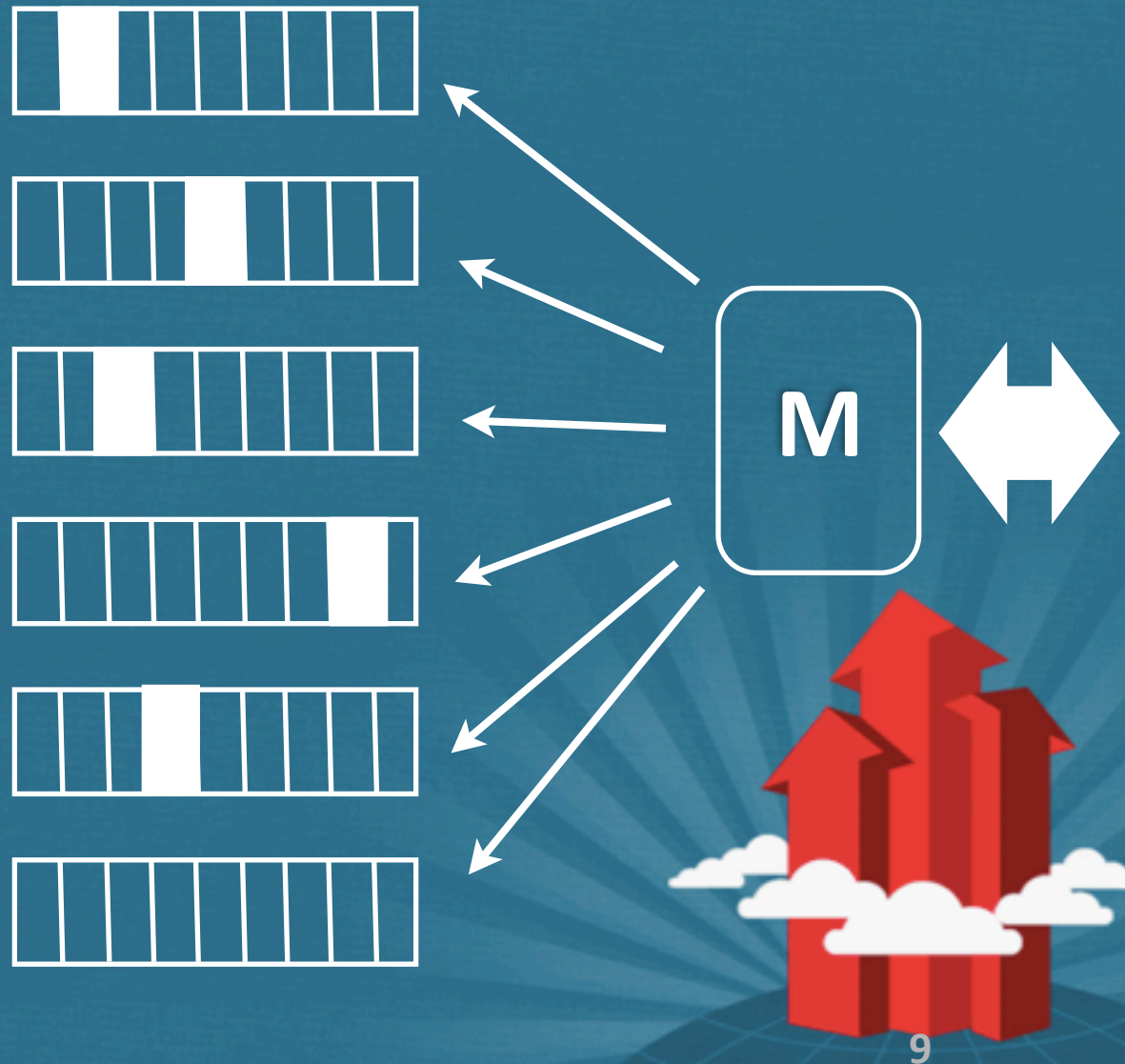
MEMBASE 1



Thursday, November 3, 11

**Membase collects memcacheds,
gives them addresses,
and makes them available to the network AS ONE.**

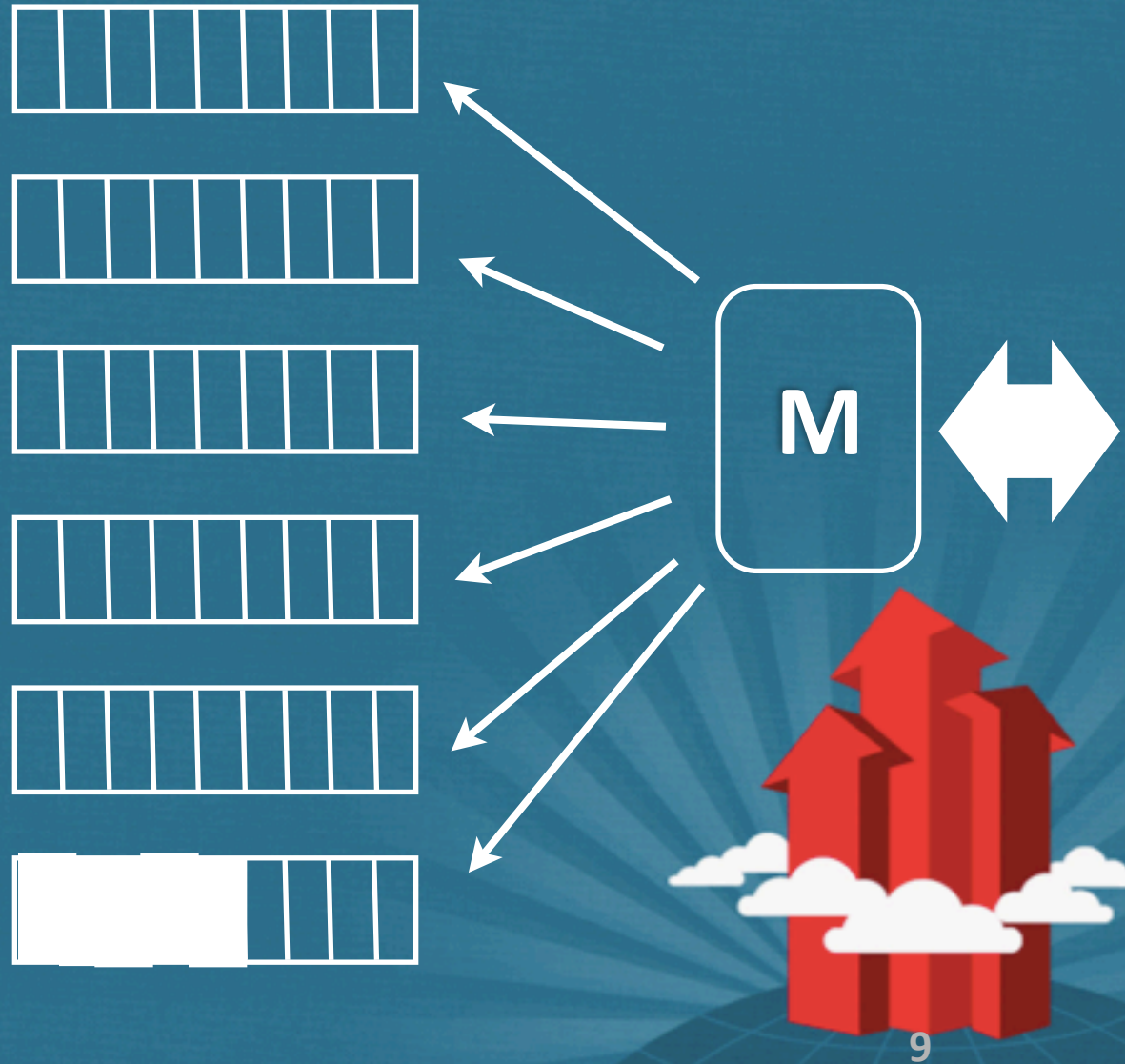
MEMBASE 1



Thursday, November 3, 11

**Membase collects memcacheds,
gives them addresses,
and makes them available to the network AS ONE.**

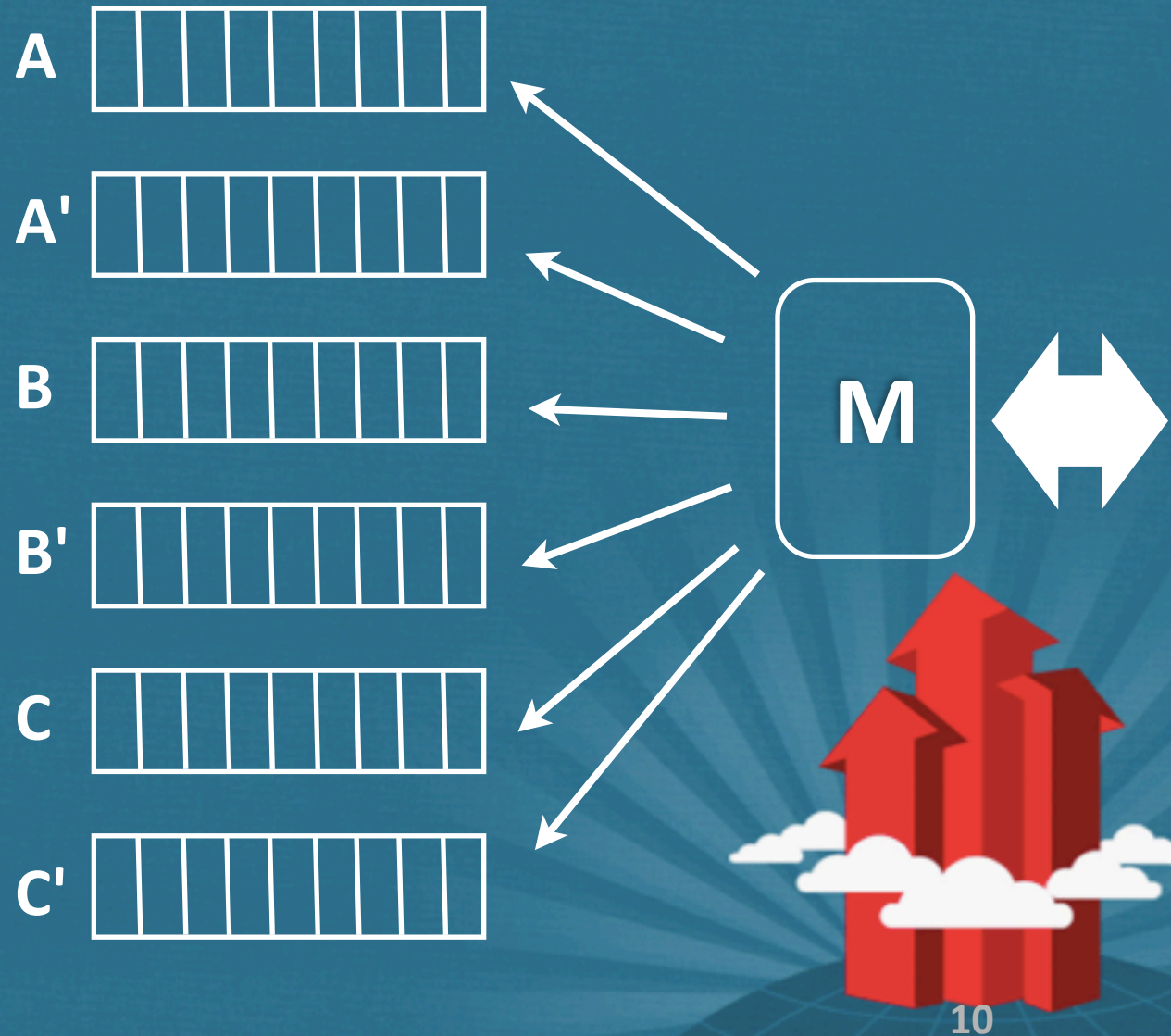
MEMBASE 1



Thursday, November 3, 11

**Membase collects memcacheds,
gives them addresses,
and makes them available to the network AS ONE.**

MEMBASE 2



10

Thursday, November 3, 11

It can do more sophisticated things like keeping a backup set (or multiple) on different nodes for fault tolerance

MEMBASE 2

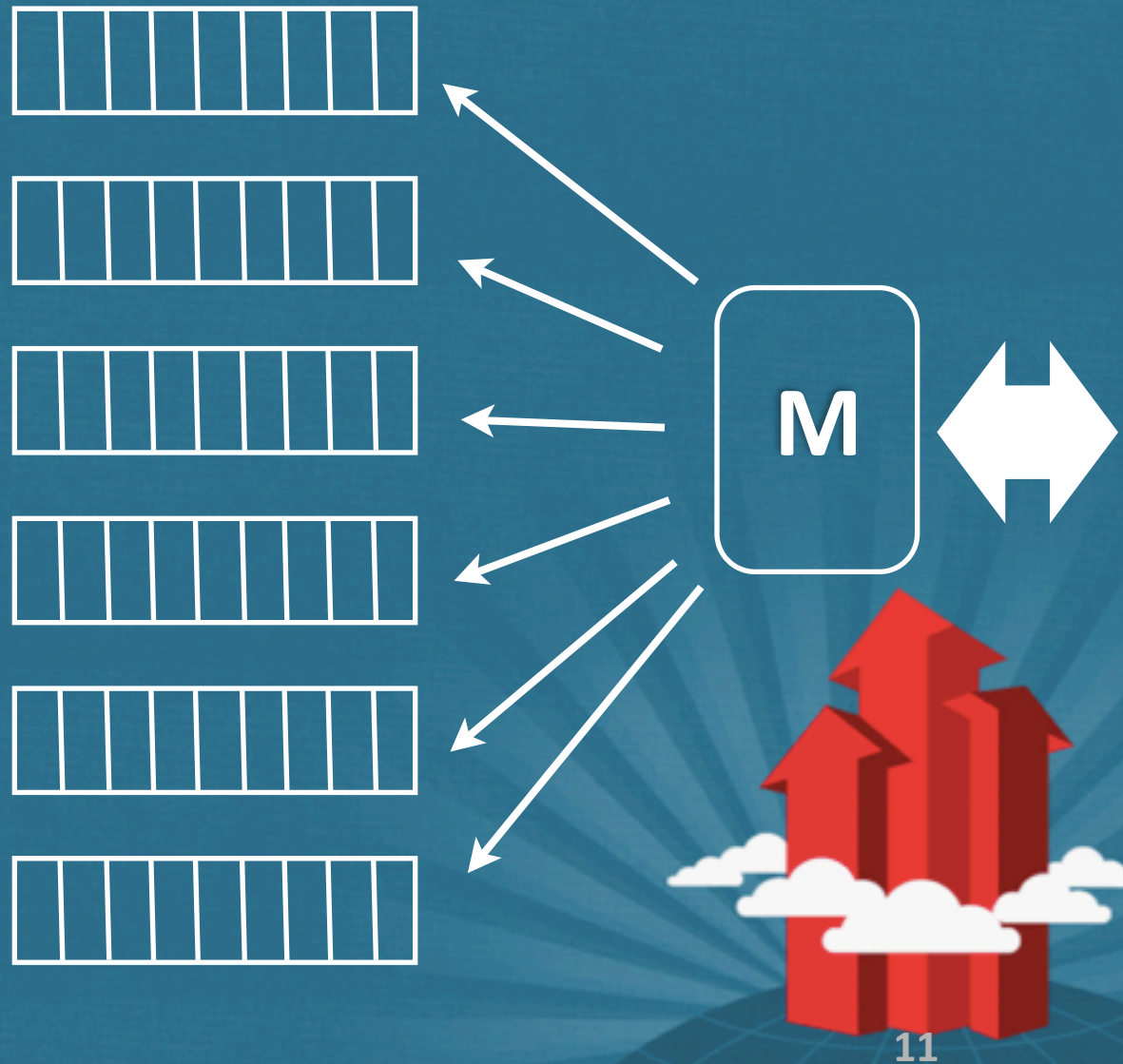
A



Thursday, November 3, 11

It can do more sophisticated things like keeping a backup set (or multiple) on different nodes for fault tolerance

MEMBASE 3



11

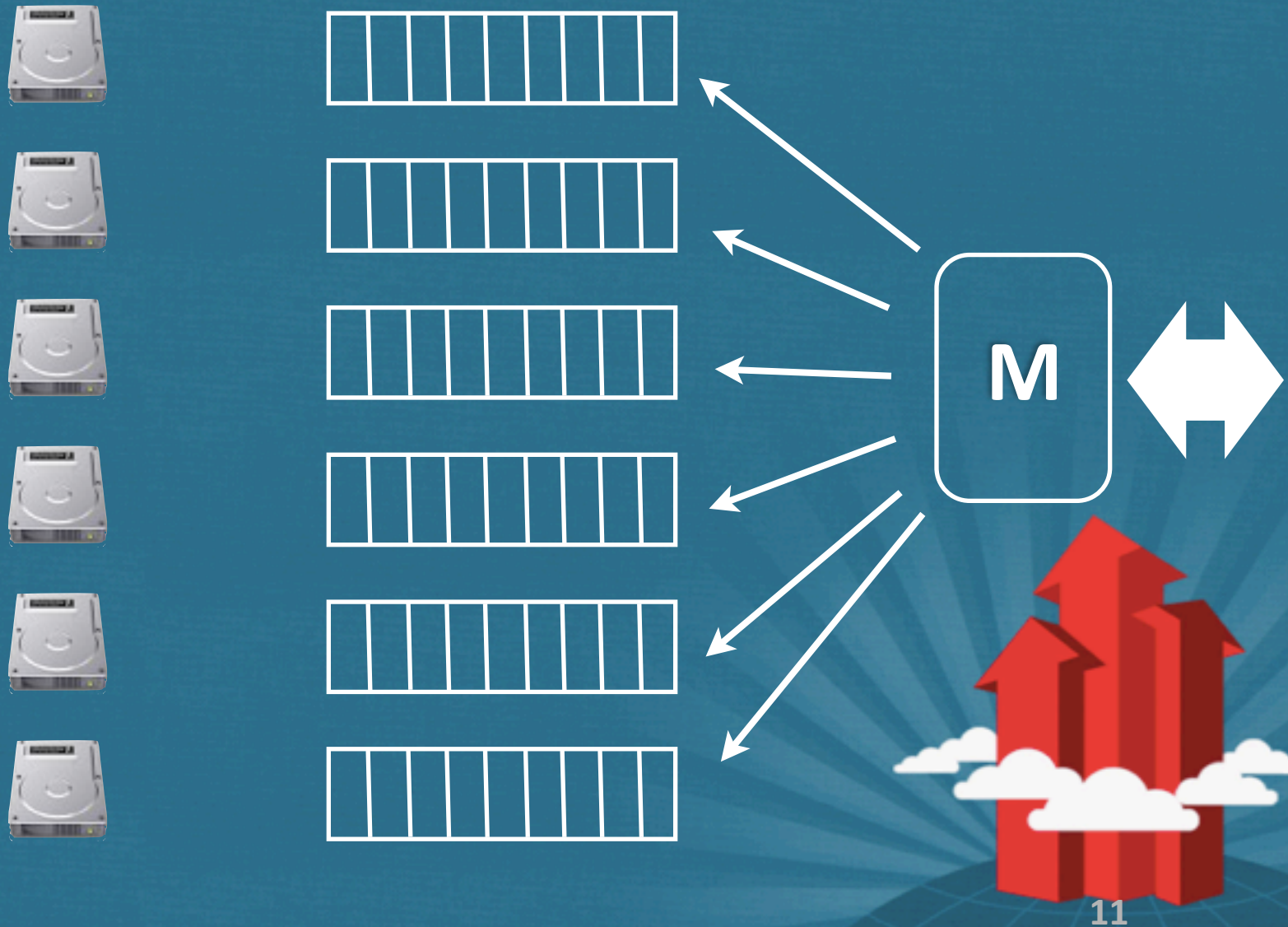
Thursday, November 3, 11

All your memcached machines come with some sort of disk that sits around, why not use it?

Memcached keeps a copy of the data on disk, fetches on demand.

Datasets > RAM possible

MEMBASE 3



11

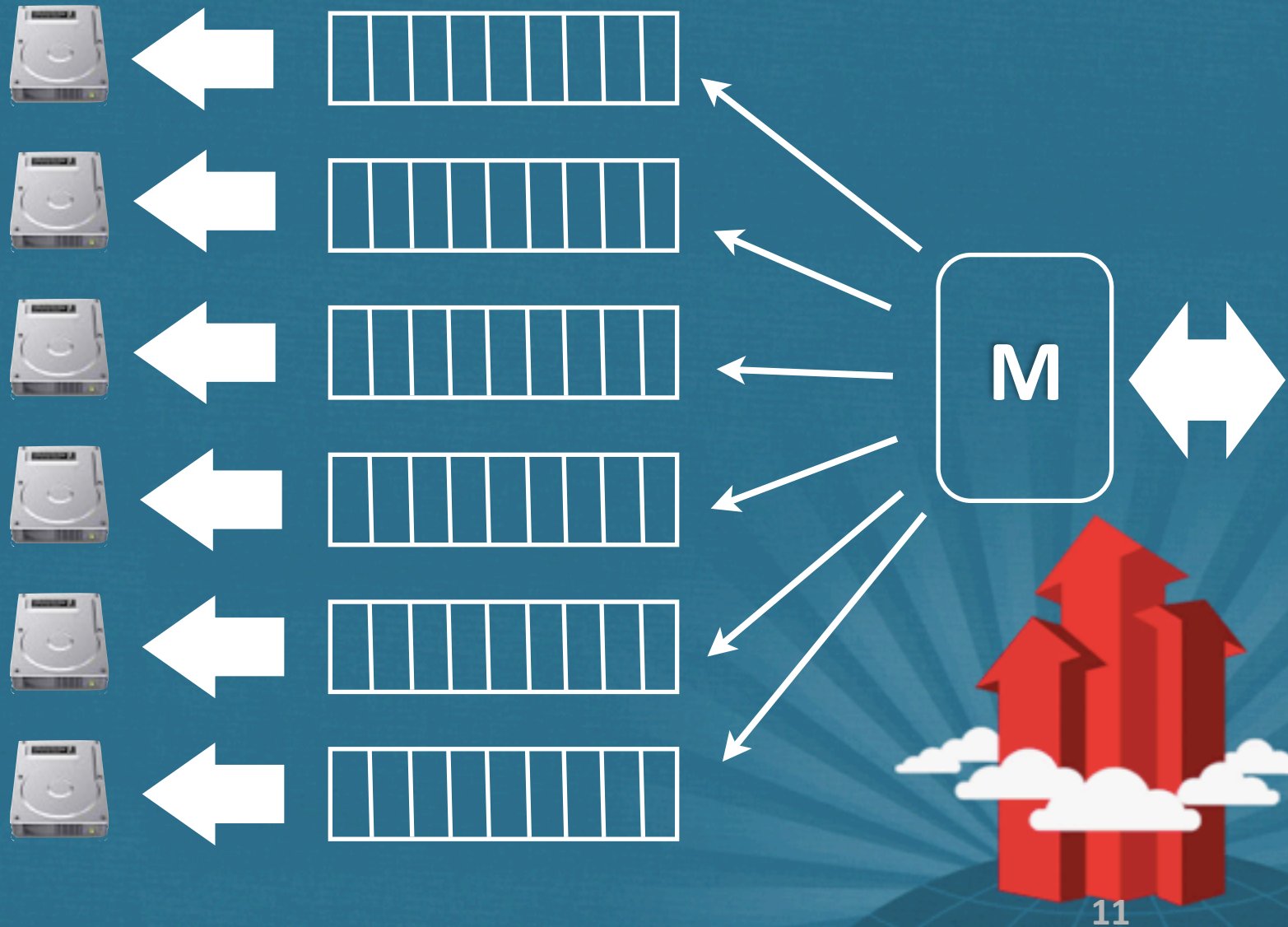
Thursday, November 3, 11

All your memcached machines come with some sort of disk that sits around, why not use it?

Memcached keeps a copy of the data on disk, fetches on demand.

Datasets > RAM possible

MEMBASE 3



11

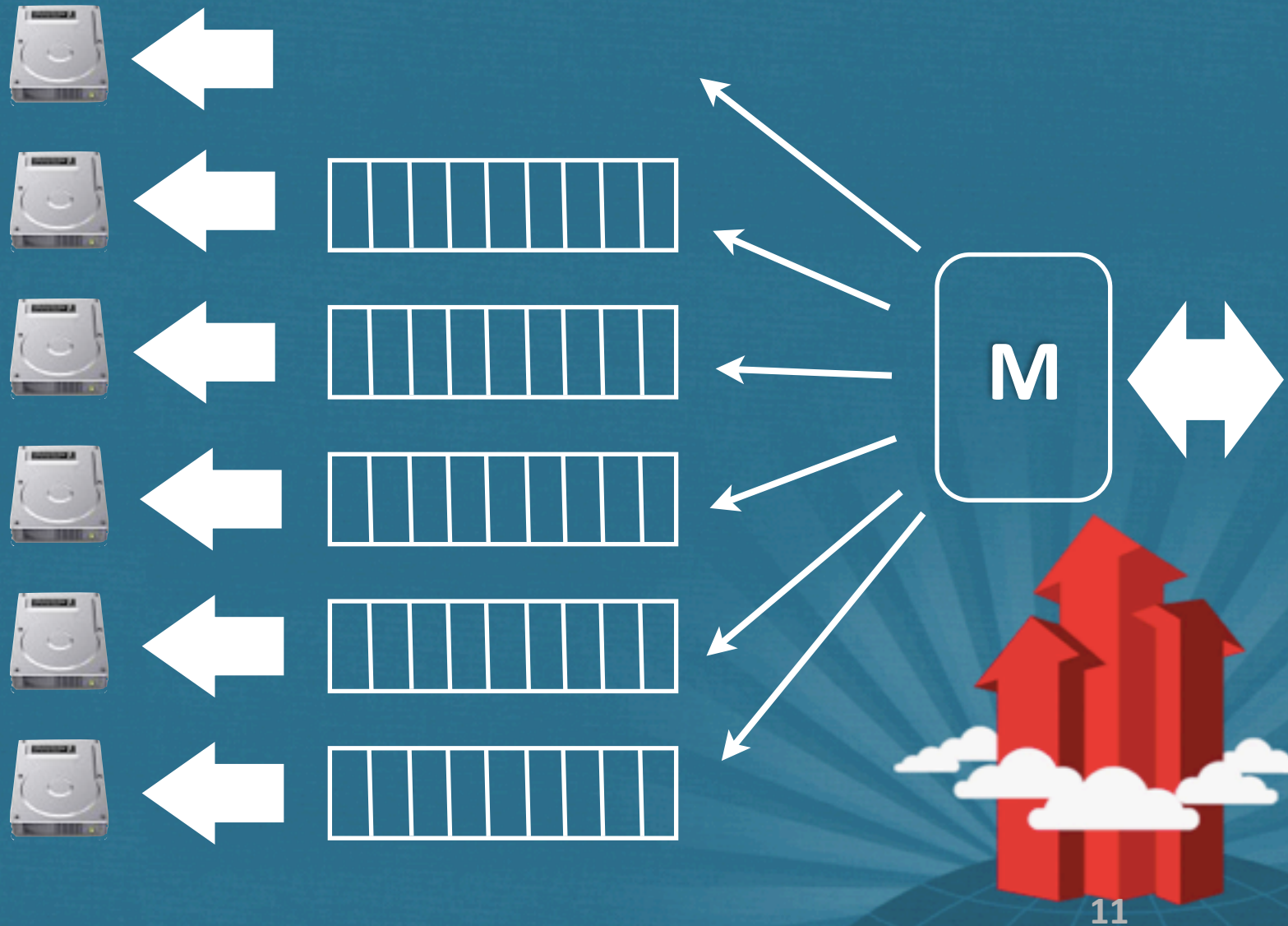
Thursday, November 3, 11

All your memcached machines come with some sort of disk that sits around, why not use it?

Memcached keeps a copy of the data on disk, fetches on demand.

Datasets > RAM possible

MEMBASE 3



11

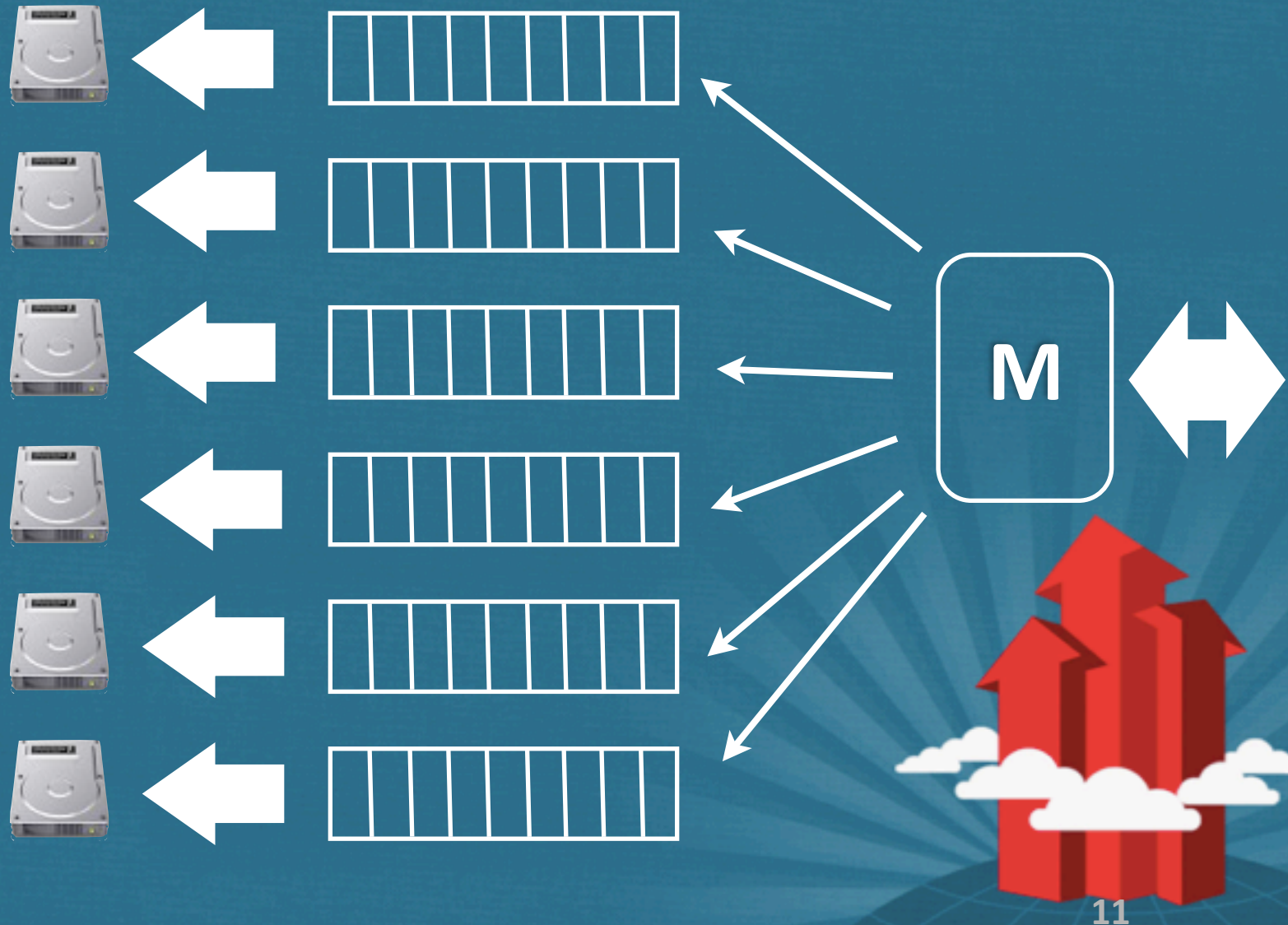
Thursday, November 3, 11

All your memcached machines come with some sort of disk that sits around, why not use it?

Memcached keeps a copy of the data on disk, fetches on demand.

Datasets > RAM possible

MEMBASE 3



11

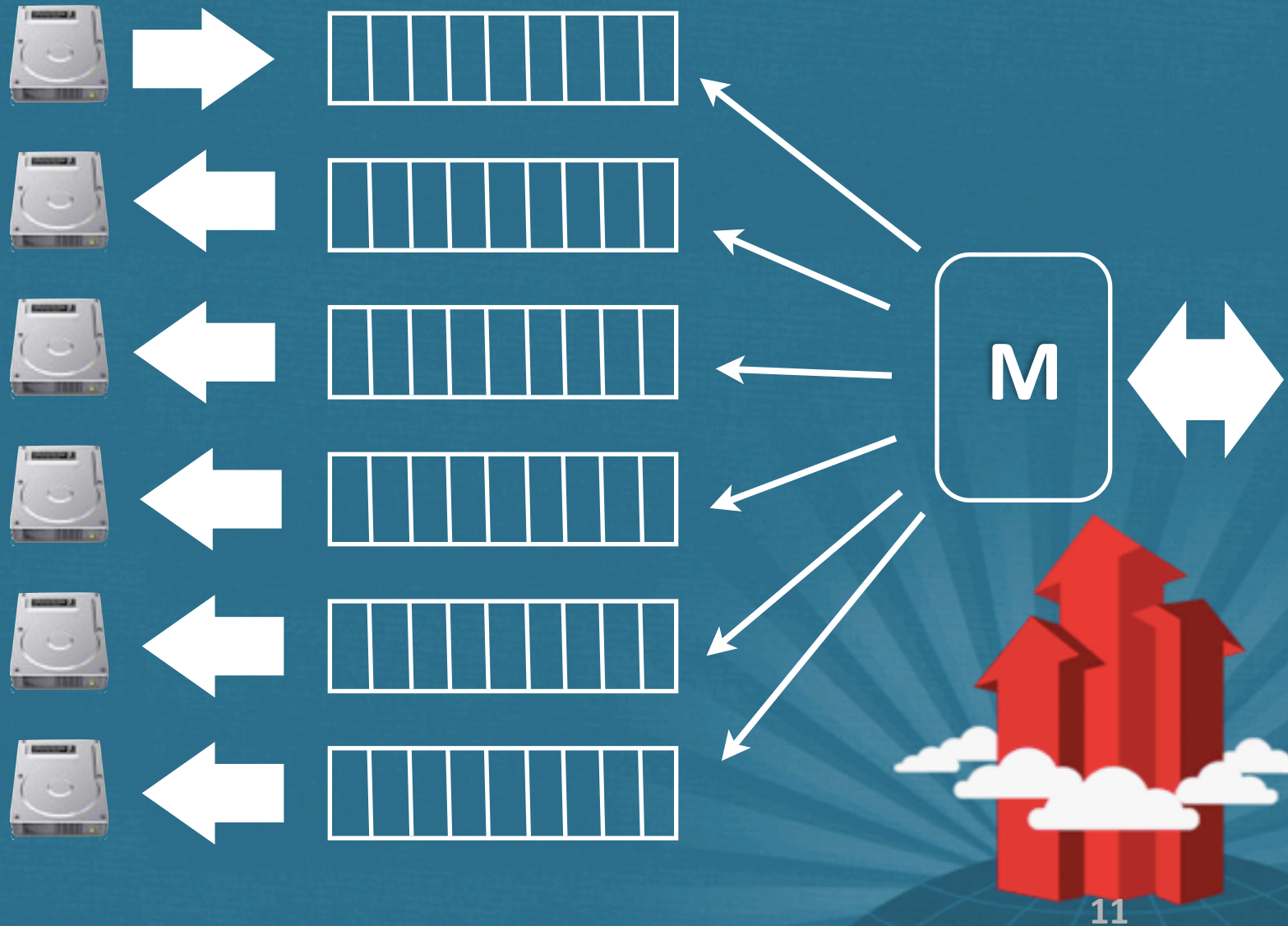
Thursday, November 3, 11

All your memcached machines come with some sort of disk that sits around, why not use it?

Memcached keeps a copy of the data on disk, fetches on demand.

Datasets > RAM possible

MEMBASE 3



11

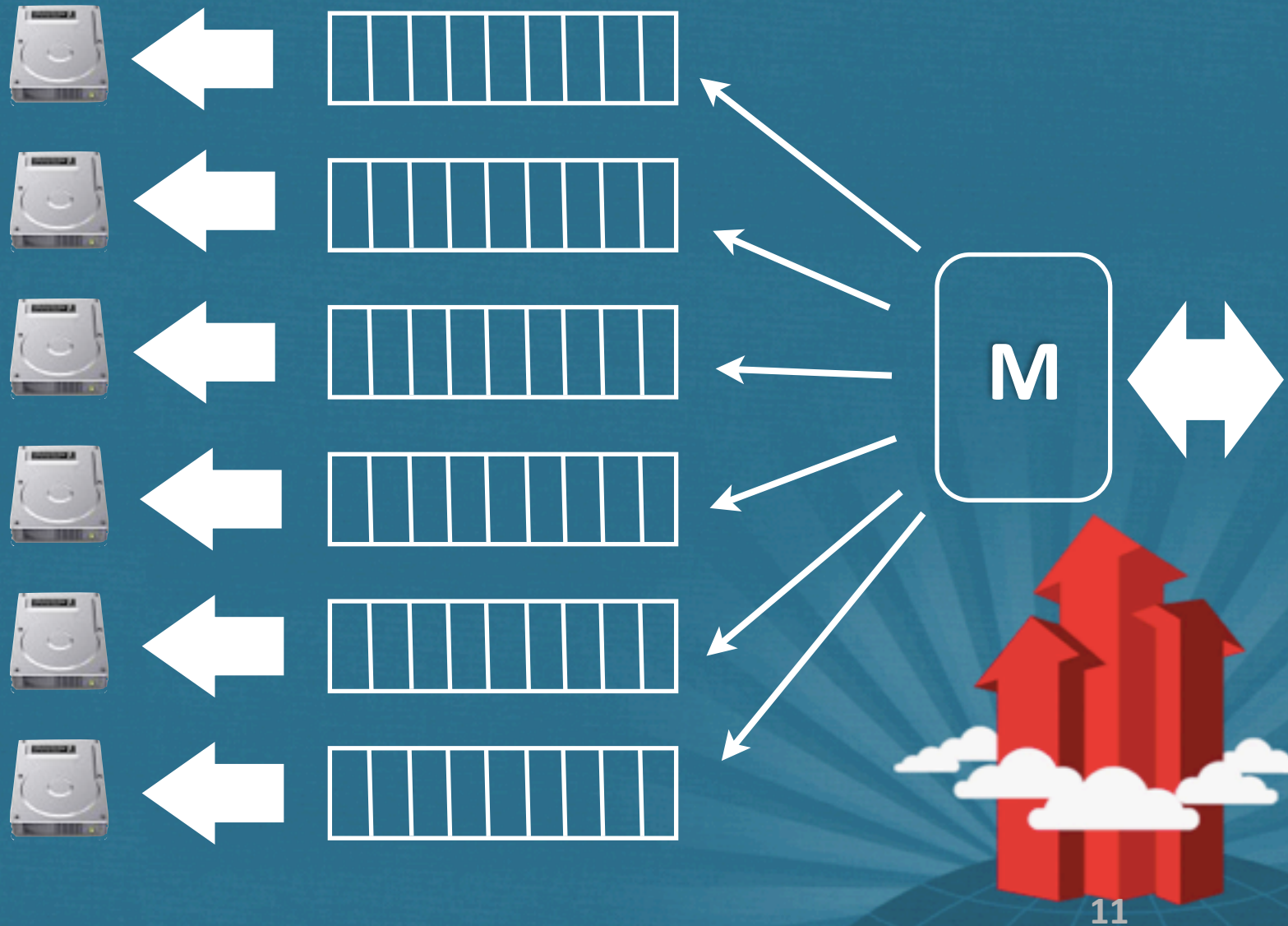
Thursday, November 3, 11

All your memcached machines come with some sort of disk that sits around, why not use it?

Memcached keeps a copy of the data on disk, fetches on demand.

Datasets > RAM possible

MEMBASE 3



11

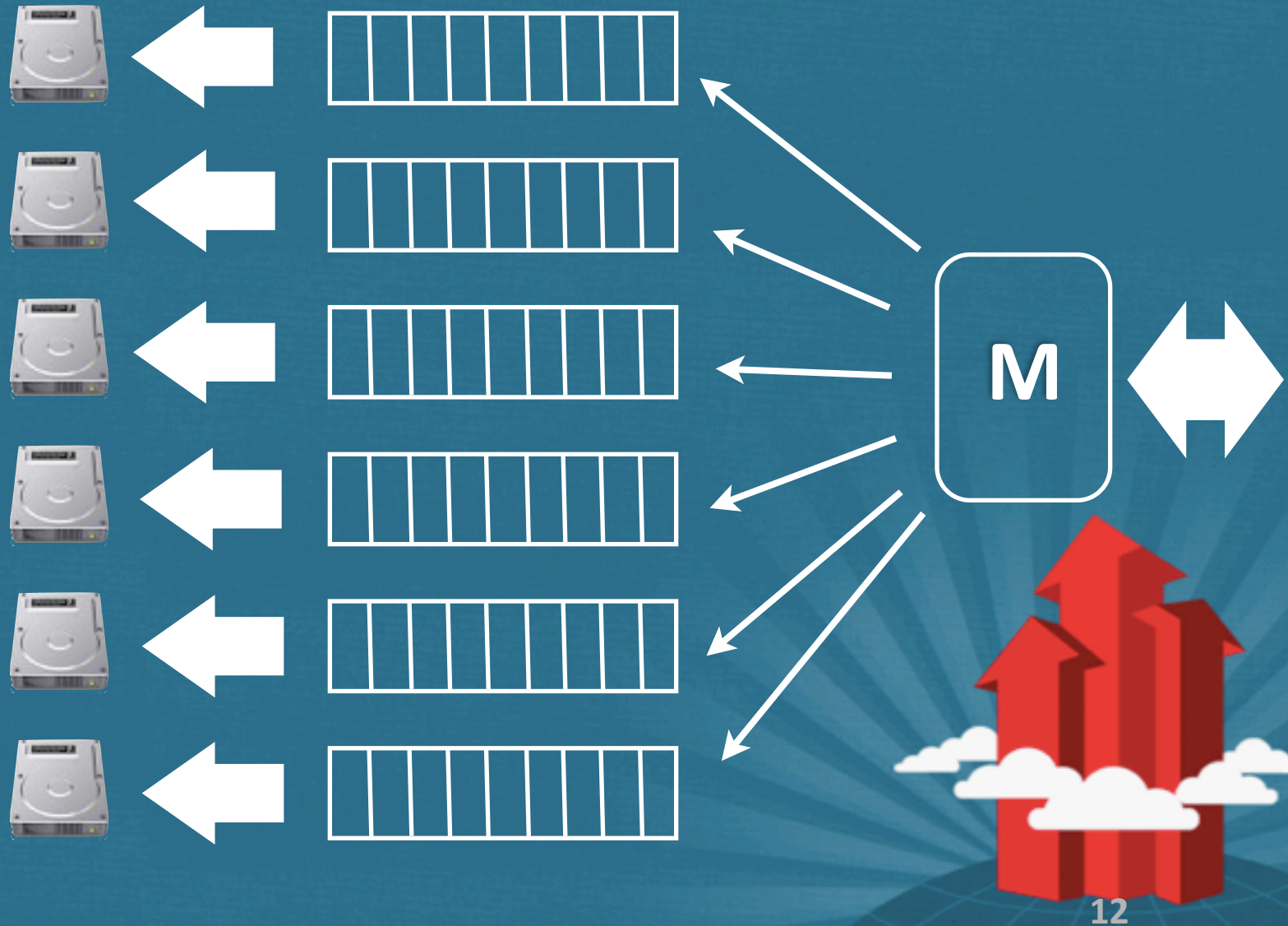
Thursday, November 3, 11

All your memcached machines come with some sort of disk that sits around, why not use it?

Memcached keeps a copy of the data on disk, fetches on demand.

Datasets > RAM possible

MEMBASE 4

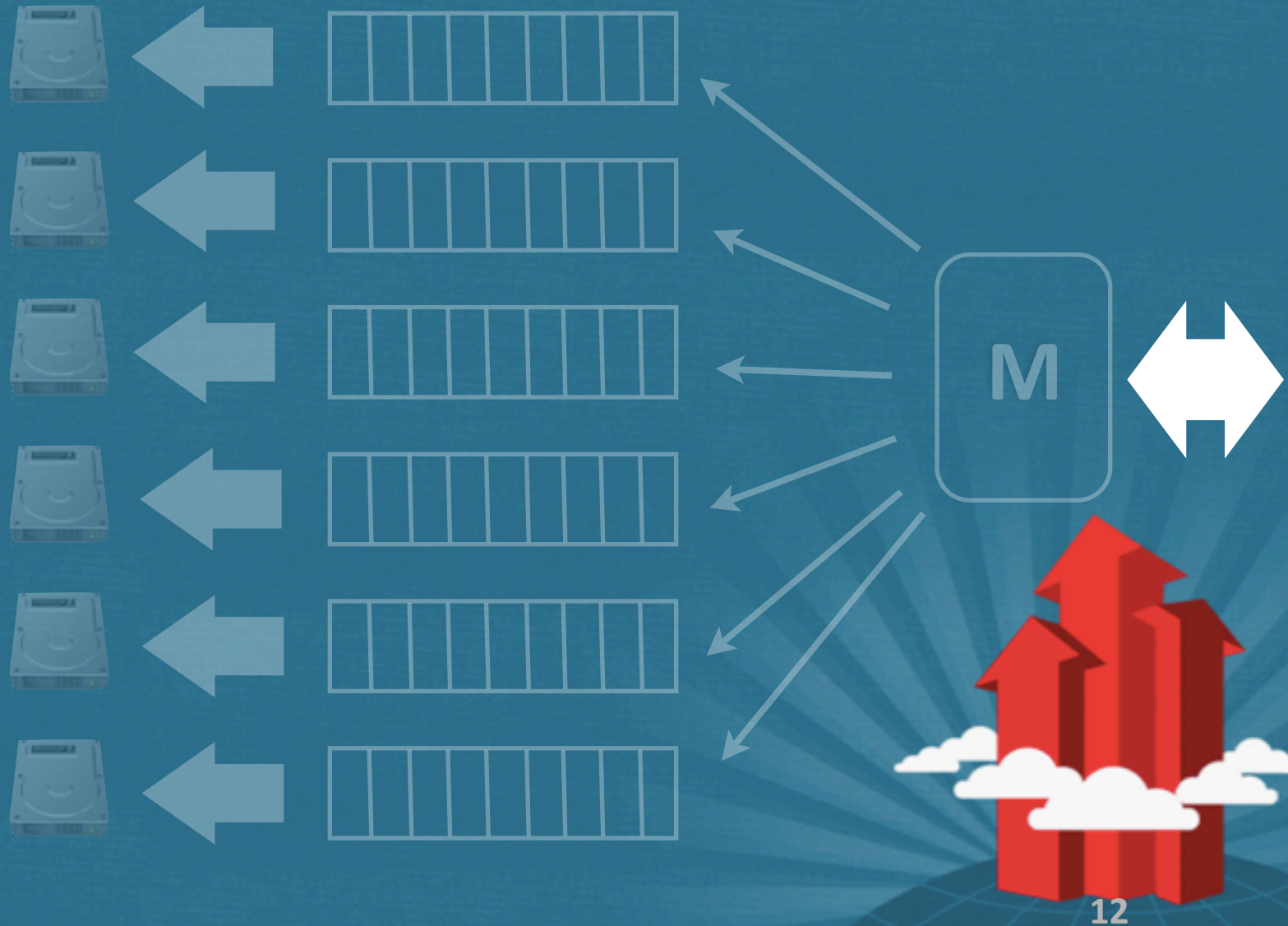


12

Thursday, November 3, 11

ALL OF THE ABOVE without interfering with the application response time

MEMBASE 4

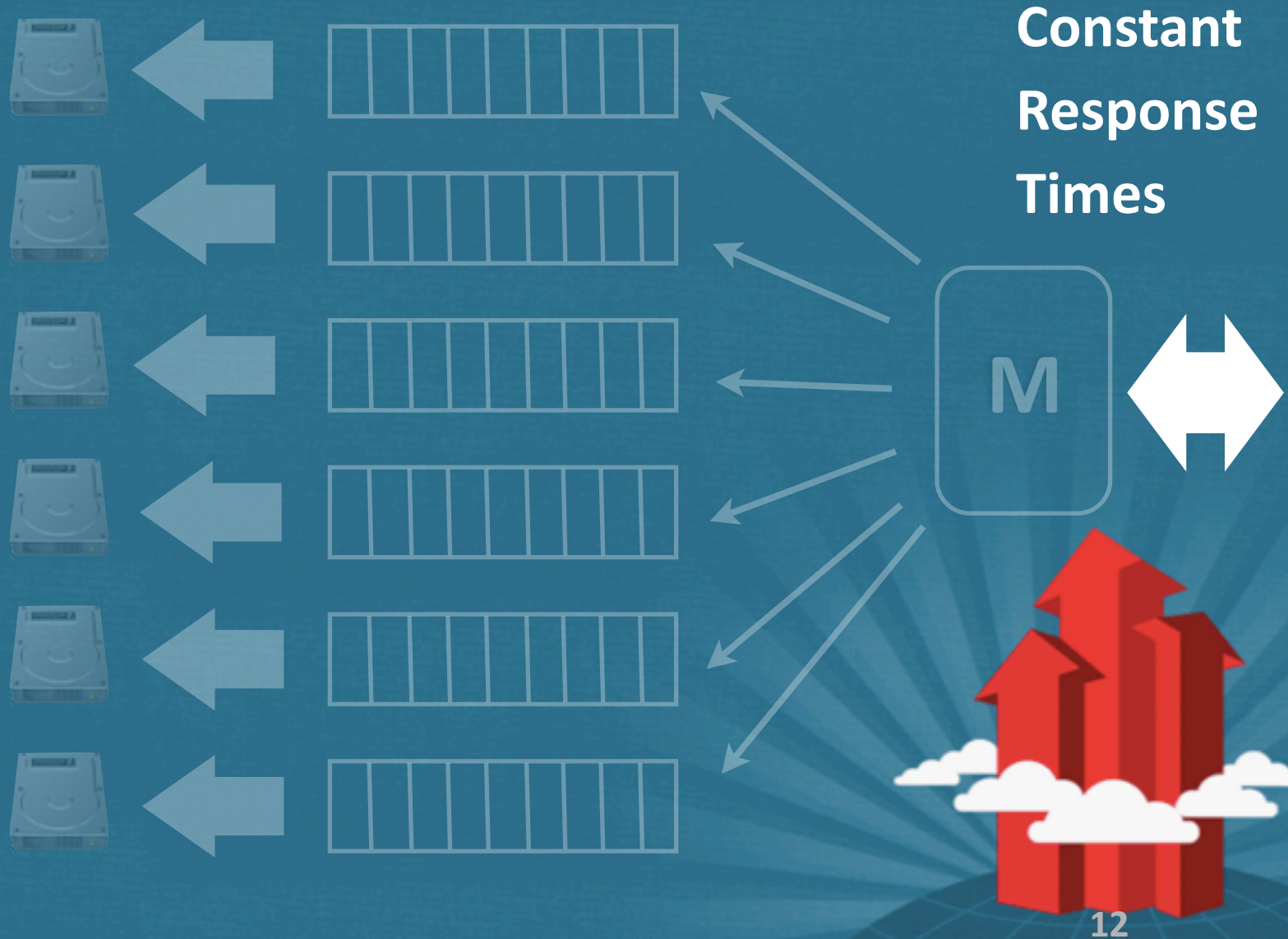


12

Thursday, November 3, 11

ALL OF THE ABOVE without interfering with the application response time

MEMBASE 4



Thursday, November 3, 11

ALL OF THE ABOVE without interfering with the application response time

MEMBASE



13

Thursday, November 3, 11

Time is marching on.

More demand for more features.

MEMBASE

- **Pro:**
 - Very fast
 - Seamless scaling
 - Persistence
 - Monitoring tools



13

Thursday, November 3, 11

Time is marching on.

More demand for more features.

MEMBASE

- **Pro:**
 - Very fast
 - Seamless scaling
 - Persistence
 - Monitoring tools
- **Contra:**
 - No querying, just KV storage
 - No Choice of CAP, just CP
 - No mobile sync



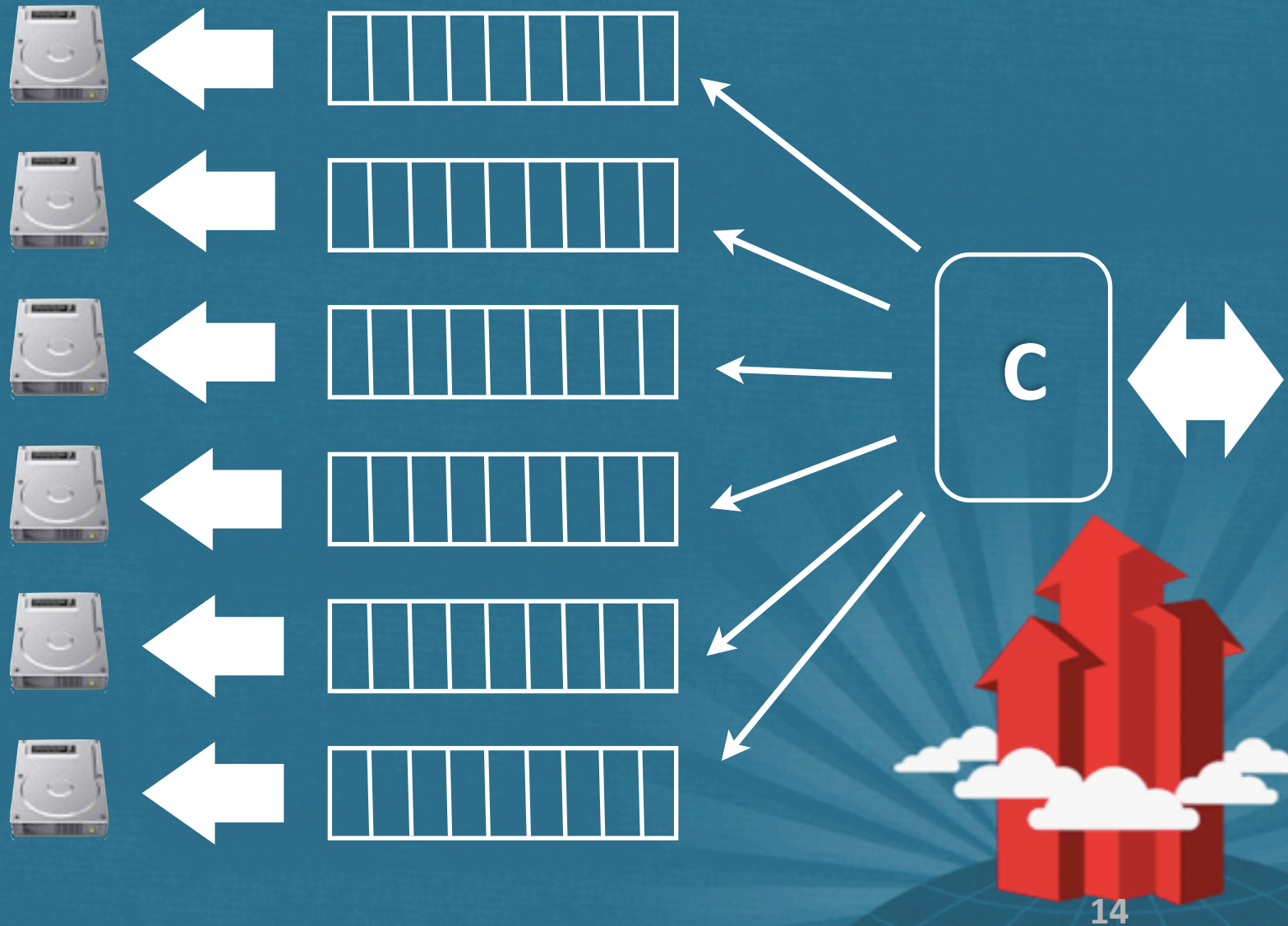
13

Thursday, November 3, 11

Time is marching on.

More demand for more features.

COUCHBASE 1

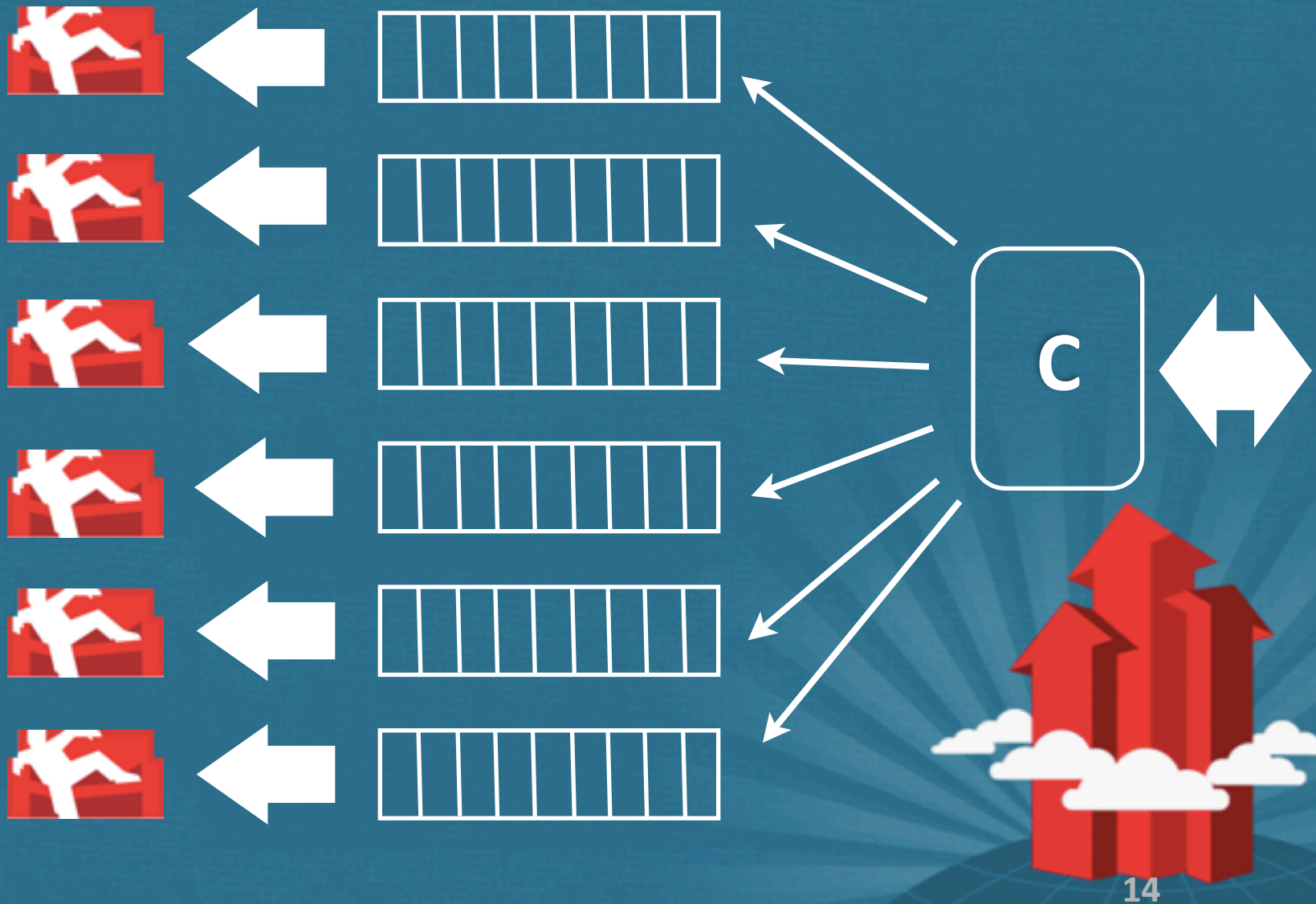


Thursday, November 3, 11

swap in da couch!

Earn: distributed querying, made simple, secondary indexes aggregation think GROUP BY

COUCHBASE 1



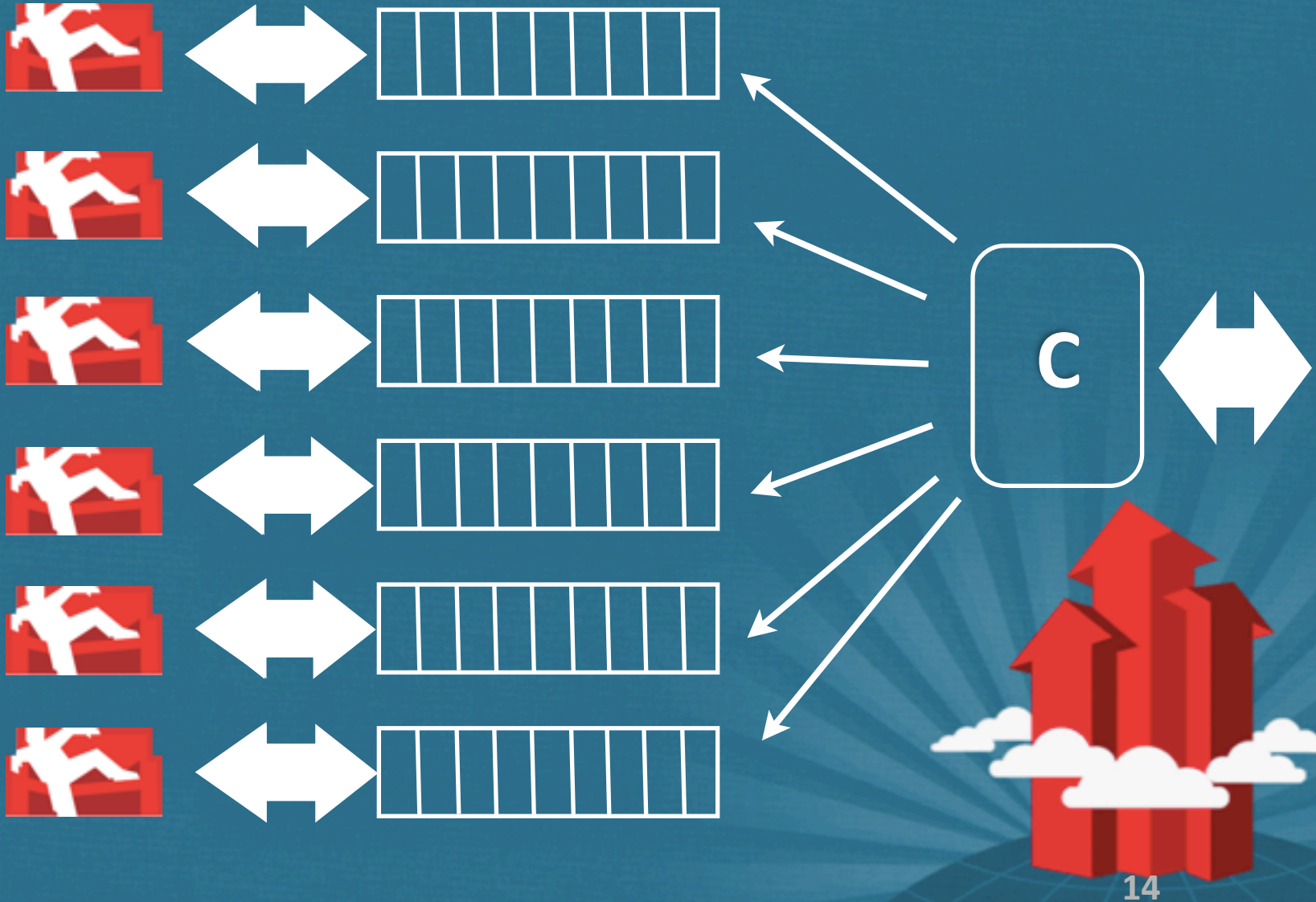
14

Thursday, November 3, 11

swap in da couch!

Earn: distributed querying, made simple, secondary indexes aggregation think GROUP BY

COUCHBASE 1



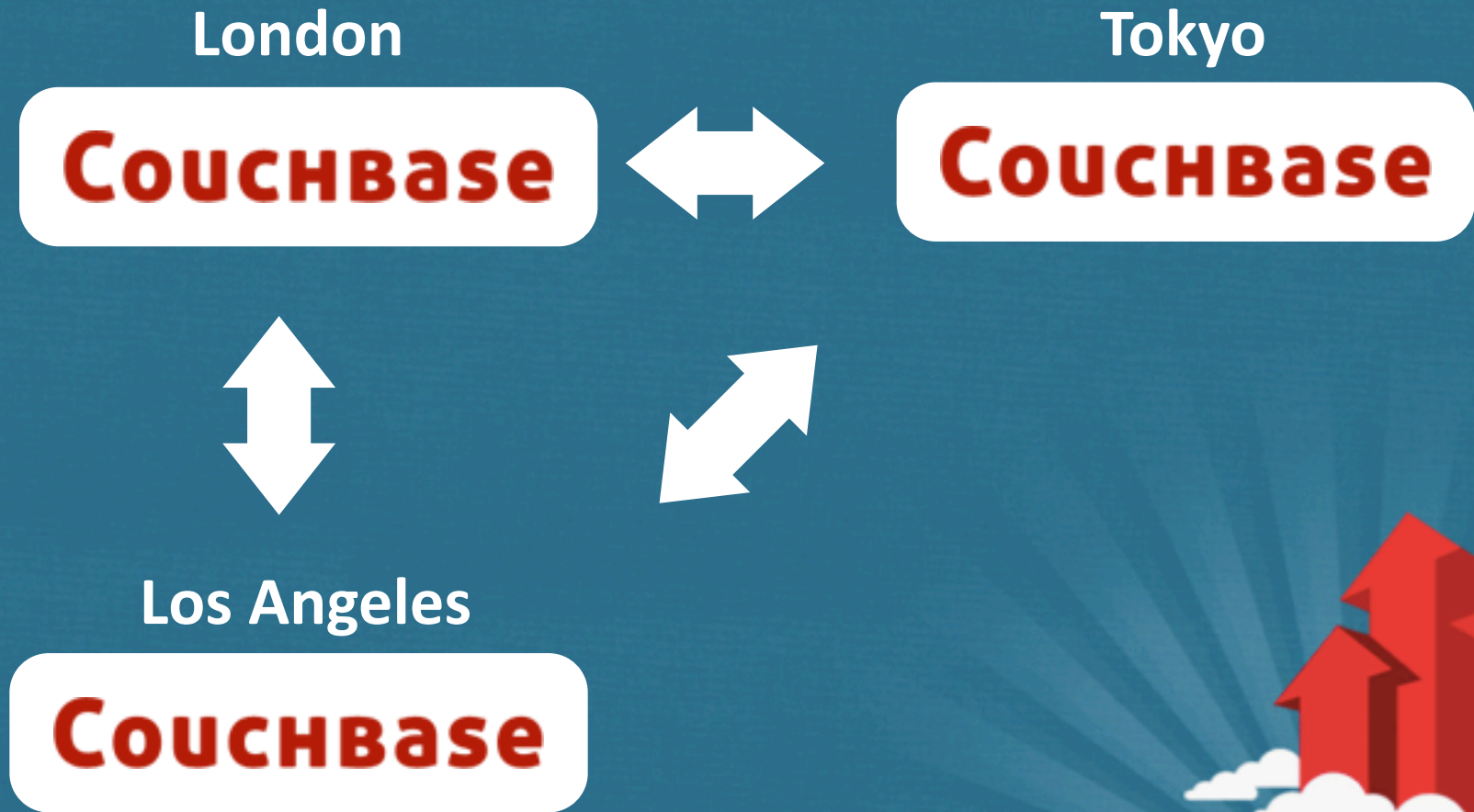
14

Thursday, November 3, 11

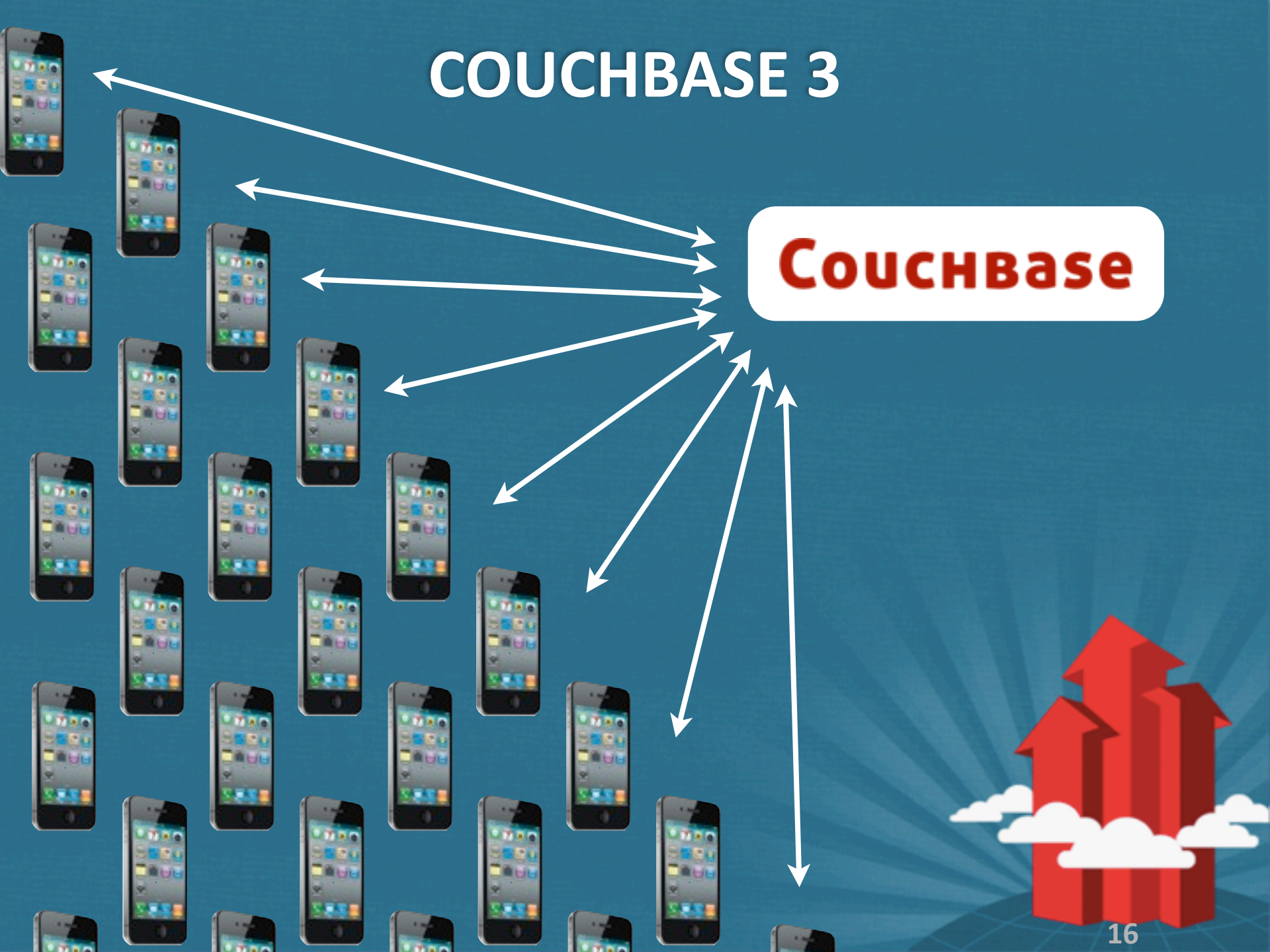
swap in da couch!

Earn: distributed querying, made simple, secondary indexes aggregation think GROUP BY

COUCHBASE 2



COUCHBASE 3



16

Thursday, November 3, 11

Zynga, ITT DoCoMo, Rim

COUCHBASE



17

COUCHBASE

- Pro:



17

COUCHBASE

- **Pro:**
 - **Fast**



17

COUCHBASE

- **Pro:**
 - Fast
 - Querying



COUCHBASE

- **Pro:**
 - Fast
 - Querying
 - Persistence



COUCHBASE

- **Pro:**
 - Fast
 - Querying
 - Persistence
 - Choice of CP and AP



COUCHBASE

- **Pro:**
 - Fast
 - Querying
 - Persistence
 - Choice of CP and AP
 - Mobile sync



COUCHBASE

- **Pro:**
 - Fast
 - Querying
 - Persistence
 - Choice of CP and AP
 - Mobile sync
 - Ops friendly



COUCHBASE

- **Pro:**
 - Fast
 - Querying
 - Persistence
 - Choice of CP and AP
 - Mobile sync
 - Ops friendly
 - Memcached and CouchDB APIs



COUCHBASE

- **Pro:**
 - Fast
 - Querying
 - Persistence
 - Choice of CP and AP
 - Mobile sync
 - Ops friendly
 - Memcached and CouchDB APIs
 - (Drop into existing installations)



COUCHBASE

- **Pro:**
 - Fast
 - Querying
 - Persistence
 - Choice of CP and AP
 - Mobile sync
 - Ops friendly
 - Memcached and CouchDB APIs
 - (Drop into existing installations)
- **Contra:**



COUCHBASE

- **Pro:**
 - Fast
 - Querying
 - Persistence
 - Choice of CP and AP
 - Mobile sync
 - Ops friendly
 - Memcached and CouchDB APIs
 - (Drop into existing installations)
- **Contra:**
 - Still in beta



COUCHBASE

- **Pro:**
 - Fast
 - Querying
 - Persistence
 - Choice of CP and AP
 - Mobile sync
 - Ops friendly
 - Memcached and CouchDB APIs
 - (Drop into existing installations)
- **Contra:**
 - Still in beta
 - Developer Previews are going strong



COUCHBASE

- **Pro:**
 - Fast
 - Querying
 - Persistence
 - Choice of CP and AP
 - Mobile sync
 - Ops friendly
 - Memcached and CouchDB APIs
 - (Drop into existing installations)
- **Contra:**
 - Still in beta
 - Developer Previews are going strong
 - We are really close



COUCHBASE

- **Pro:**

- Fast
- Querying
- Persistence
- Choice of CP and AP
- Mobile sync
- Ops friendly
- Memcached and CouchDB APIs
 - (Drop into existing installations)

- **Contra:**

- Still in beta
 - Developer Previews are going strong
 - We are really close
 - Promised!



COUCHBASE



18

COUCHBASE

- **Built on reliable, proven technologies**



COUCHBASE

- Built on reliable, proven technologies
- Apache 2.0 Licensed Open Source (yay :)



COUCHBASE

- **Built on reliable, proven technologies**
- **Apache 2.0 Licensed Open Source (yay :)**
- **Great community**



COUCHBASE

- **Built on reliable, proven technologies**
- **Apache 2.0 Licensed Open Source (yay :)**
- **Great community**
- **Enterprise support & training available**



Take a seat at

CouchConf

BERLIN

November 7, 2011

<http://www.couchbase.com/couchconf-berlin>

Thursday, November 3, 11

THANKS!



20

Thursday, November 3, 11

COUCHBASE



Thursday, November 3, 11