Rethinking Topology in Cassandra

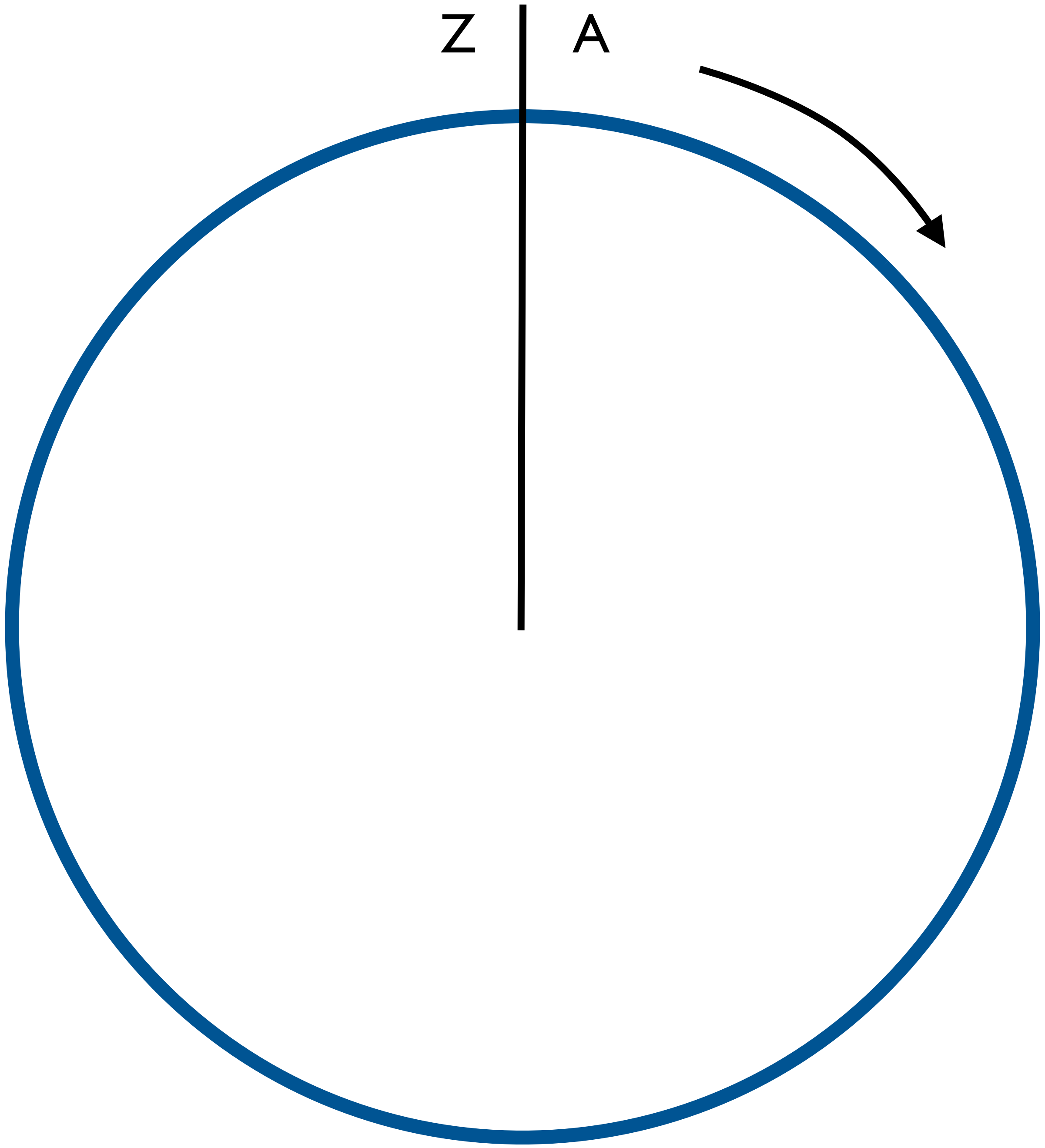
ApacheCon North America
February 28, 2013

Eric Evans
eevans@acunu.com
@jericevans

DHT 101

DHT 101

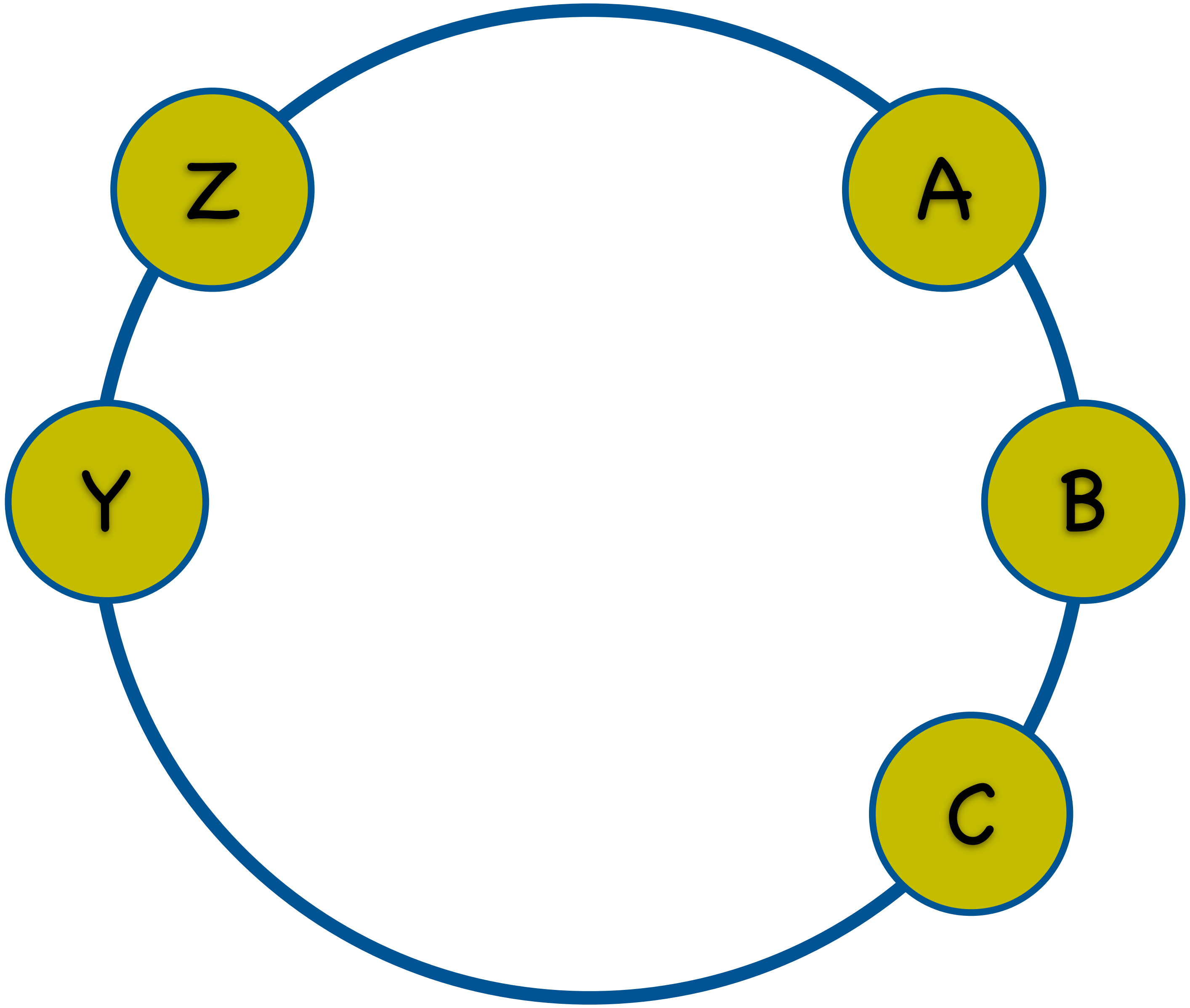
partitioning



Acunu

DHT 101

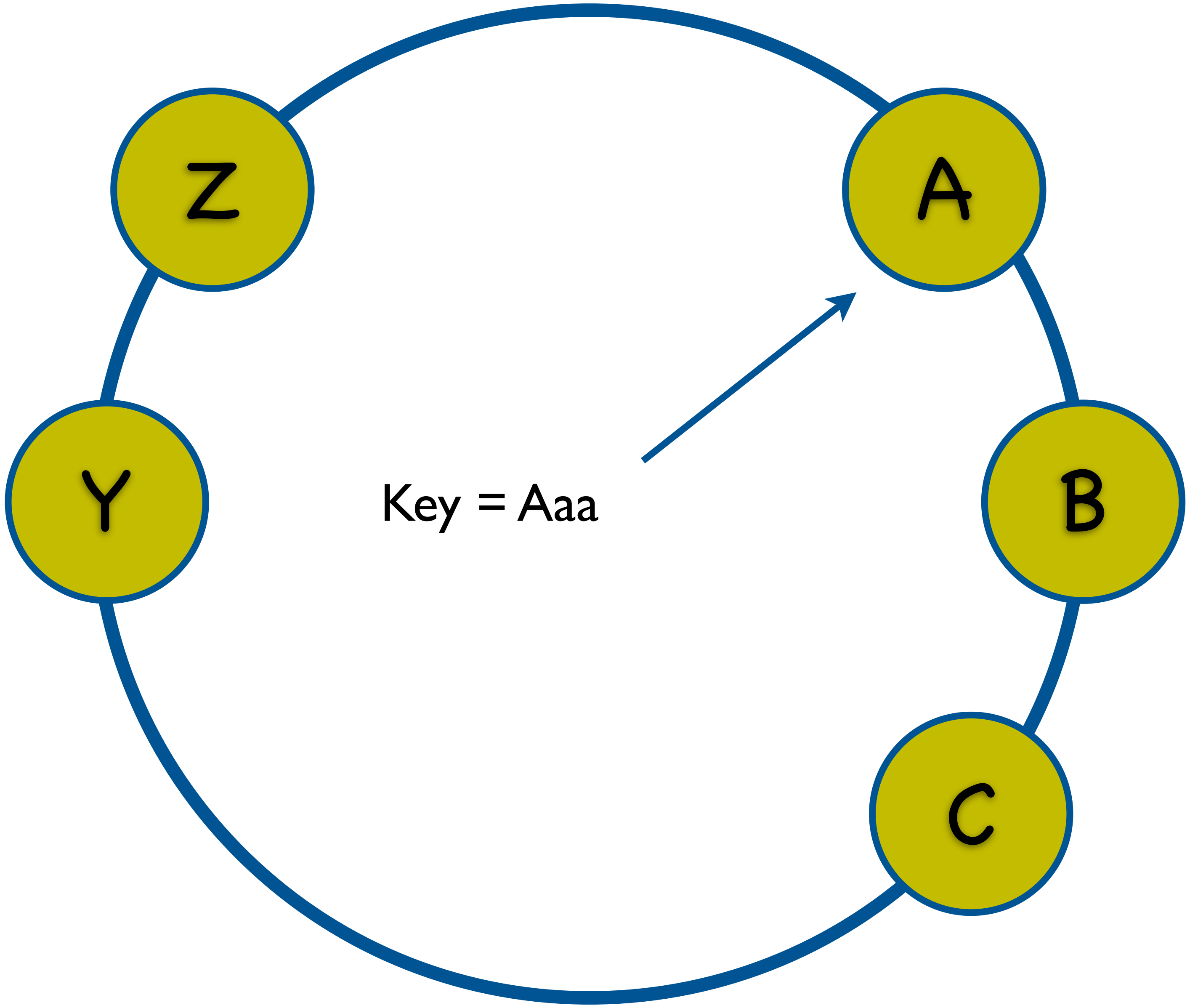
partitioning



Acunu

DHT 101

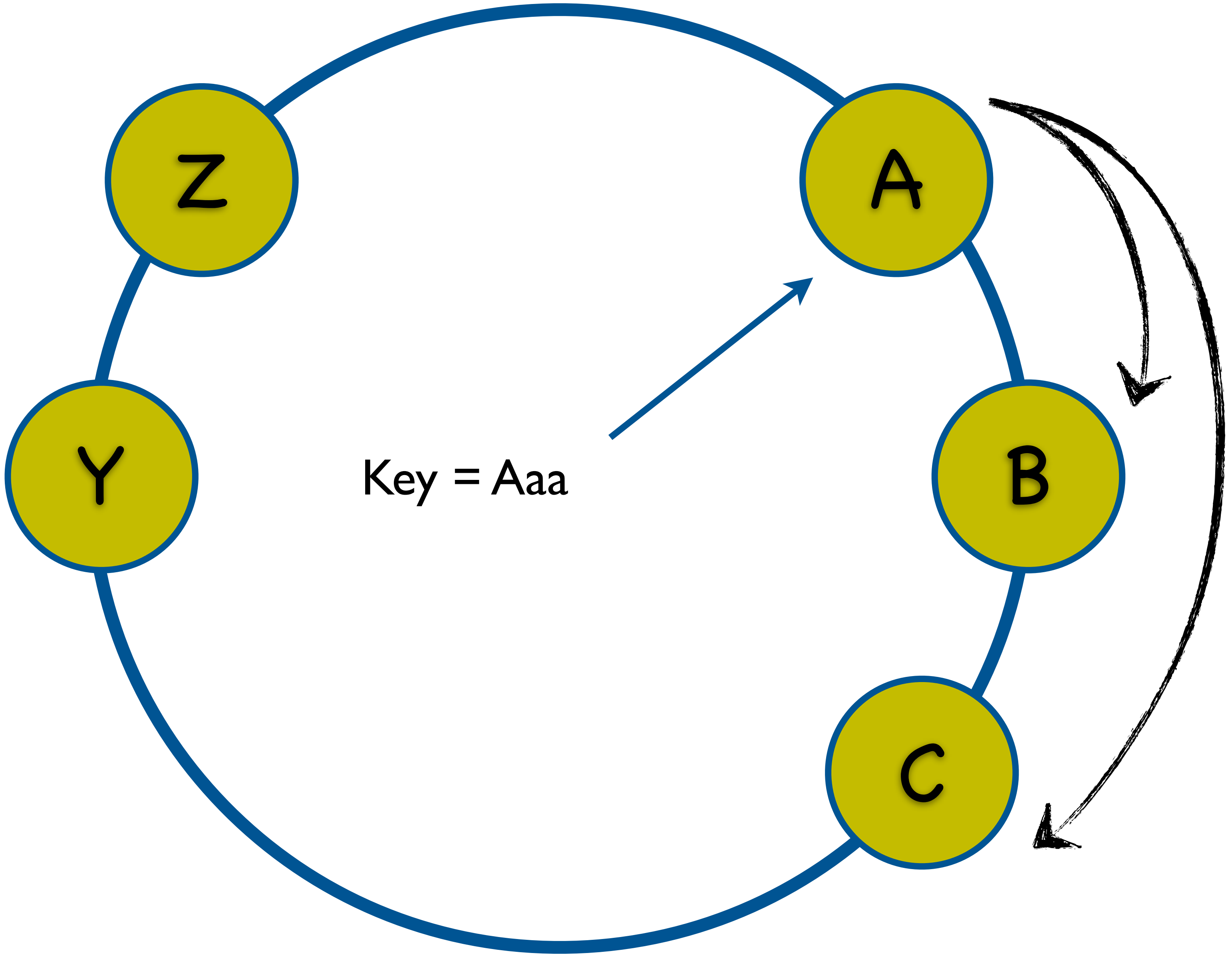
partitioning



Acunu

DHT 101

replica placement



Acunu

DHT 101

consistency

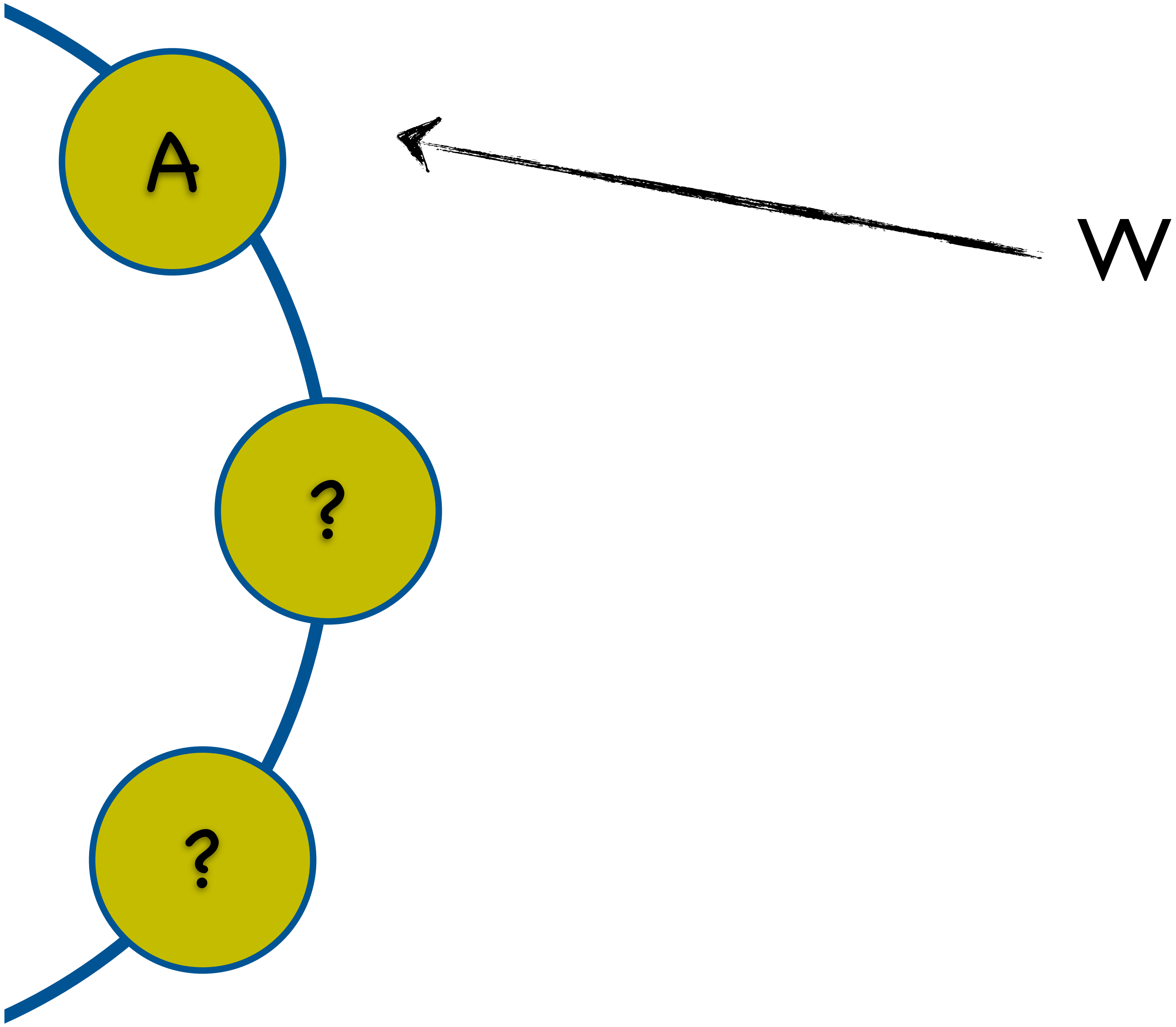
Consistency

Availability

Partition tolerance

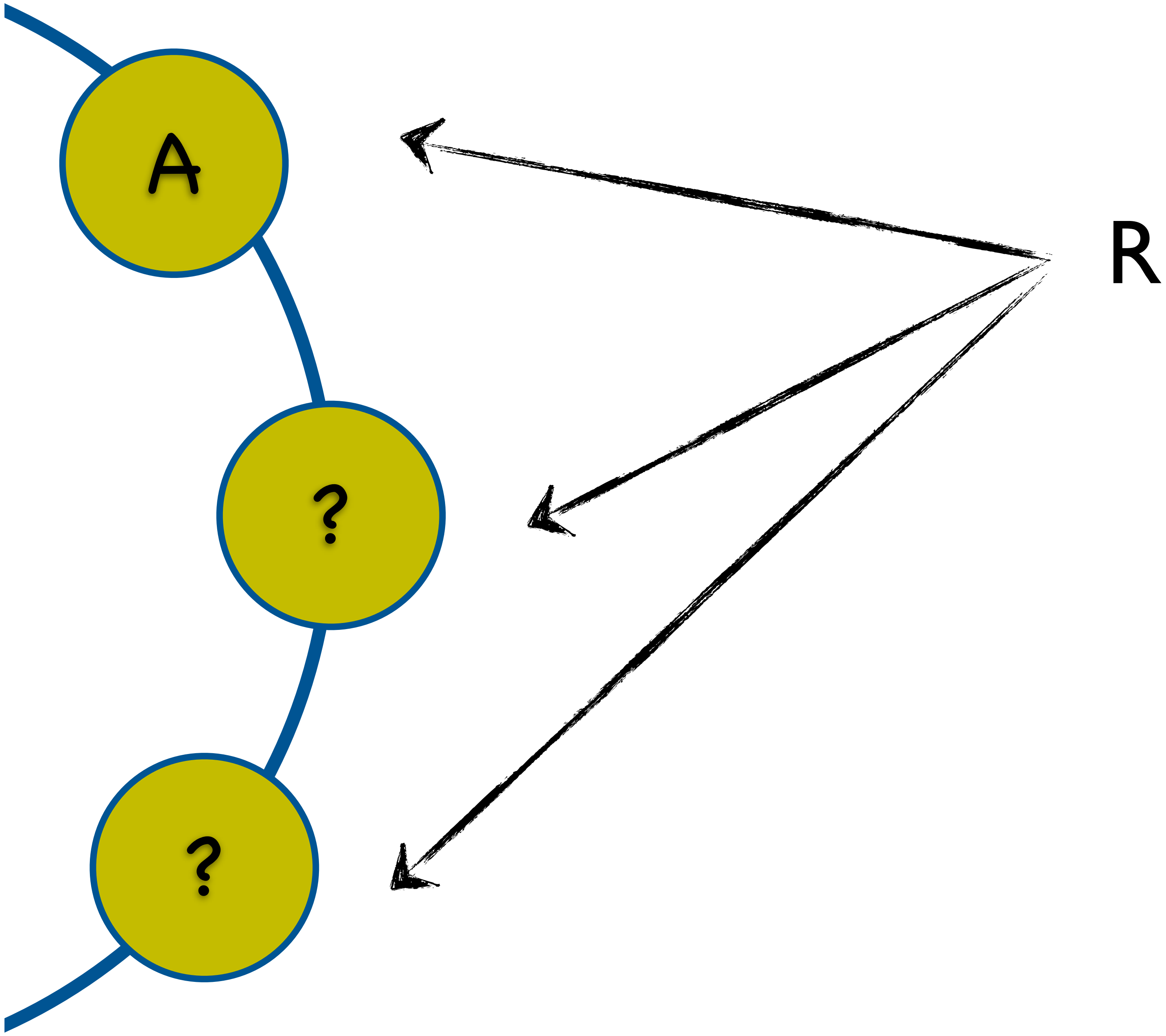
DHT 101

scenario: consistency level = one



DHT 101

scenario: consistency level = all

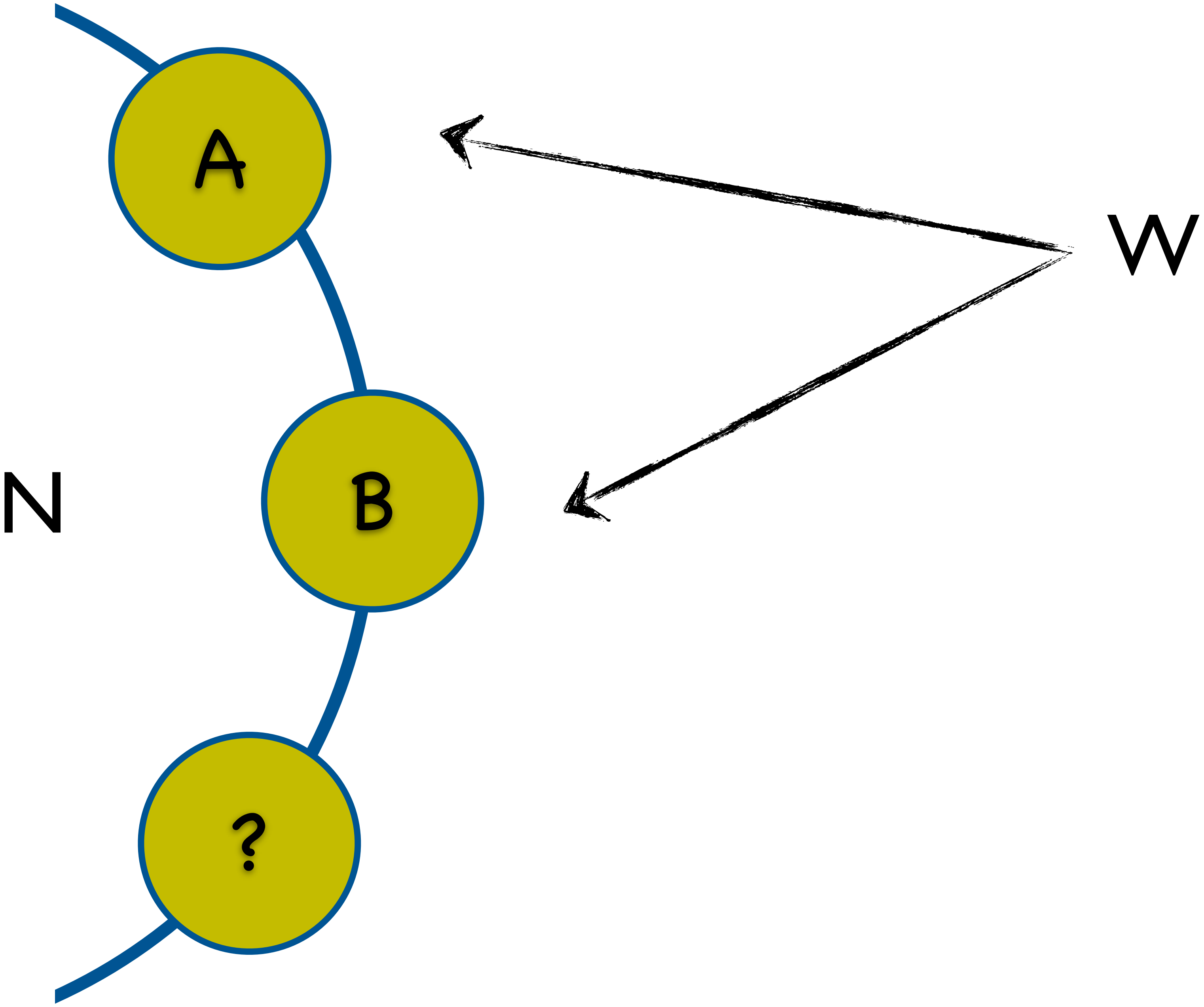


Acunu

DHT 101

scenario: quorum write

R+W > N

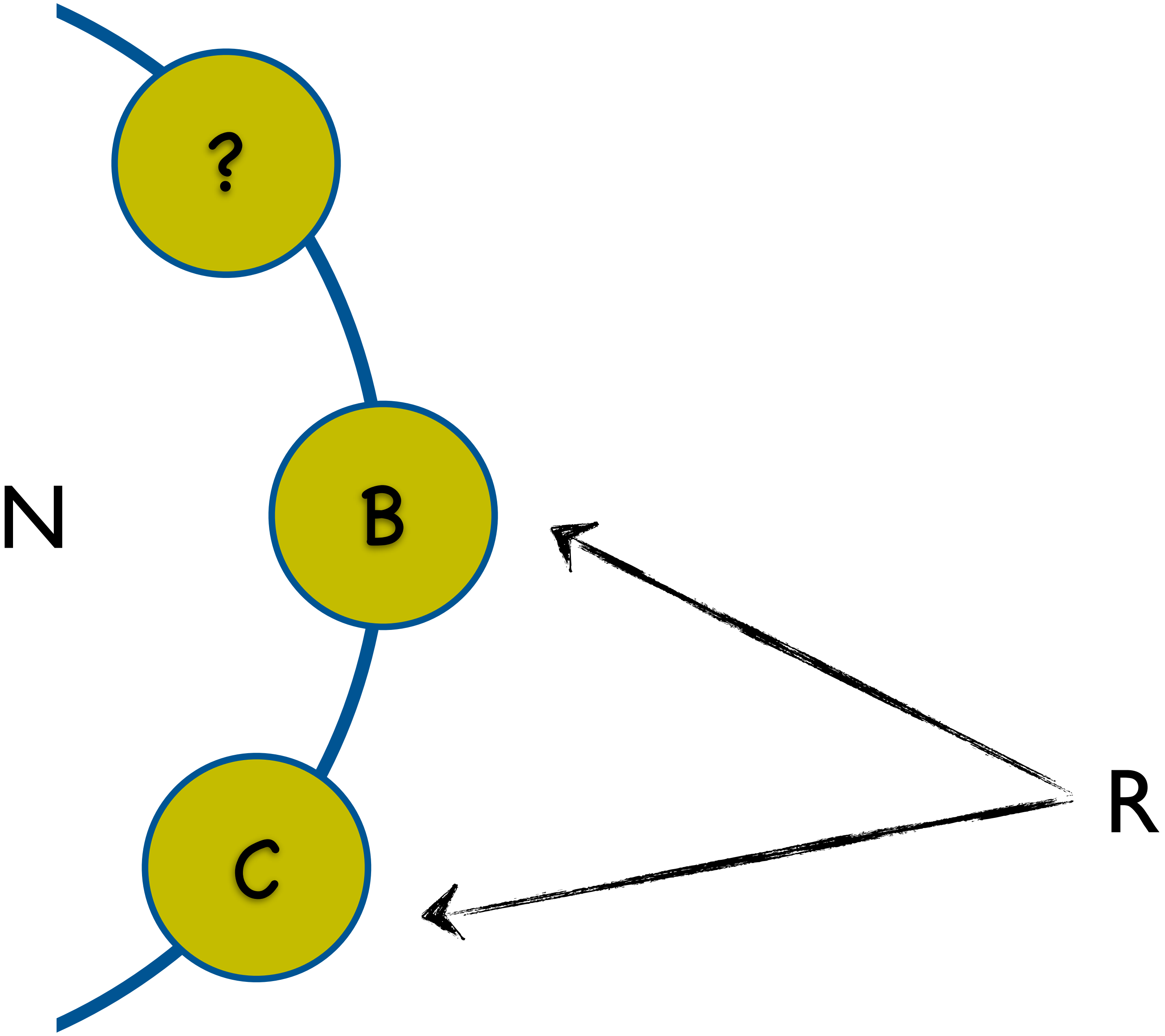


Acunu

DHT 101

scenario: quorum read

R+W > N



Awesome, yes?

Acunu

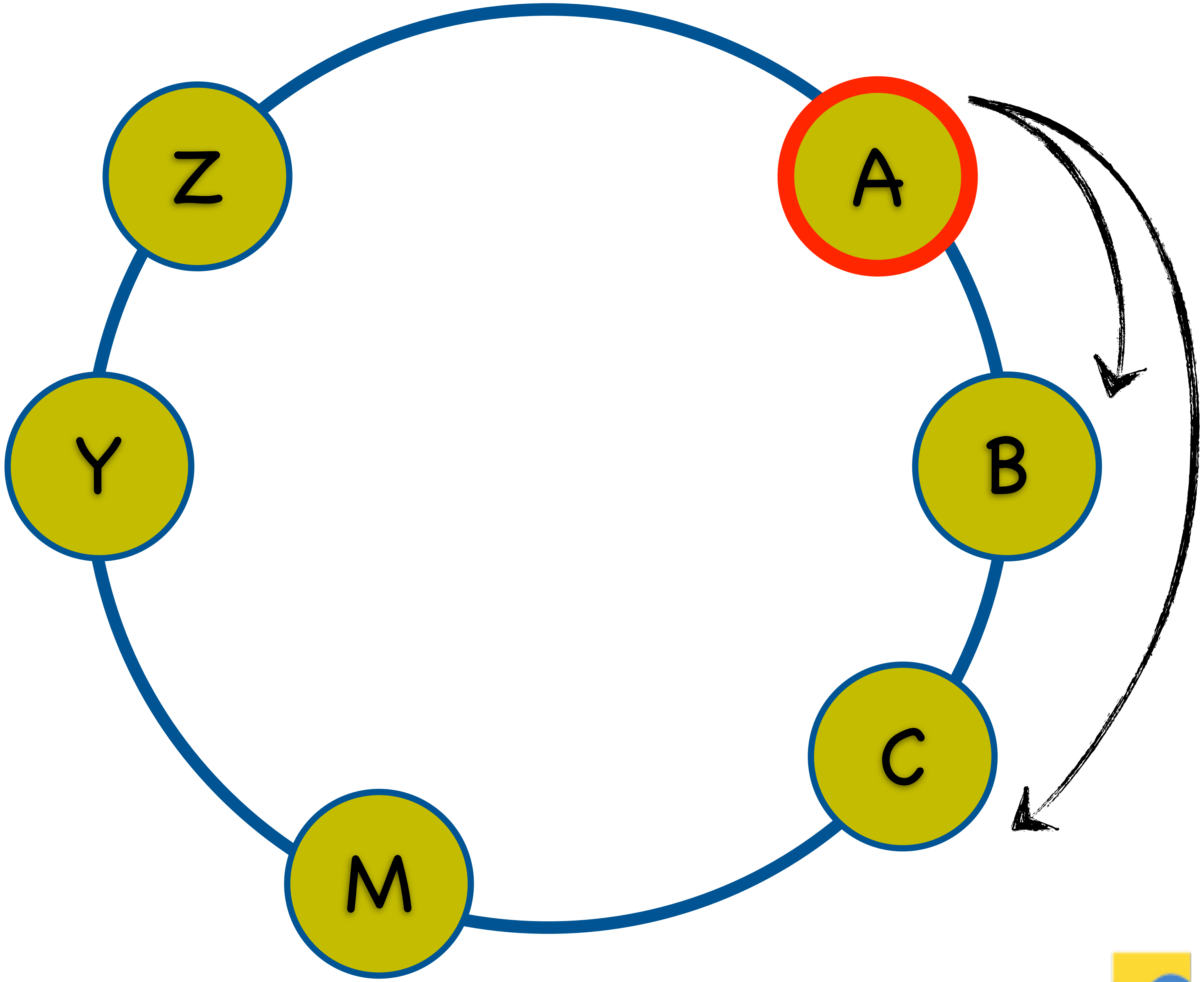
Well...

Acunu

Problem:

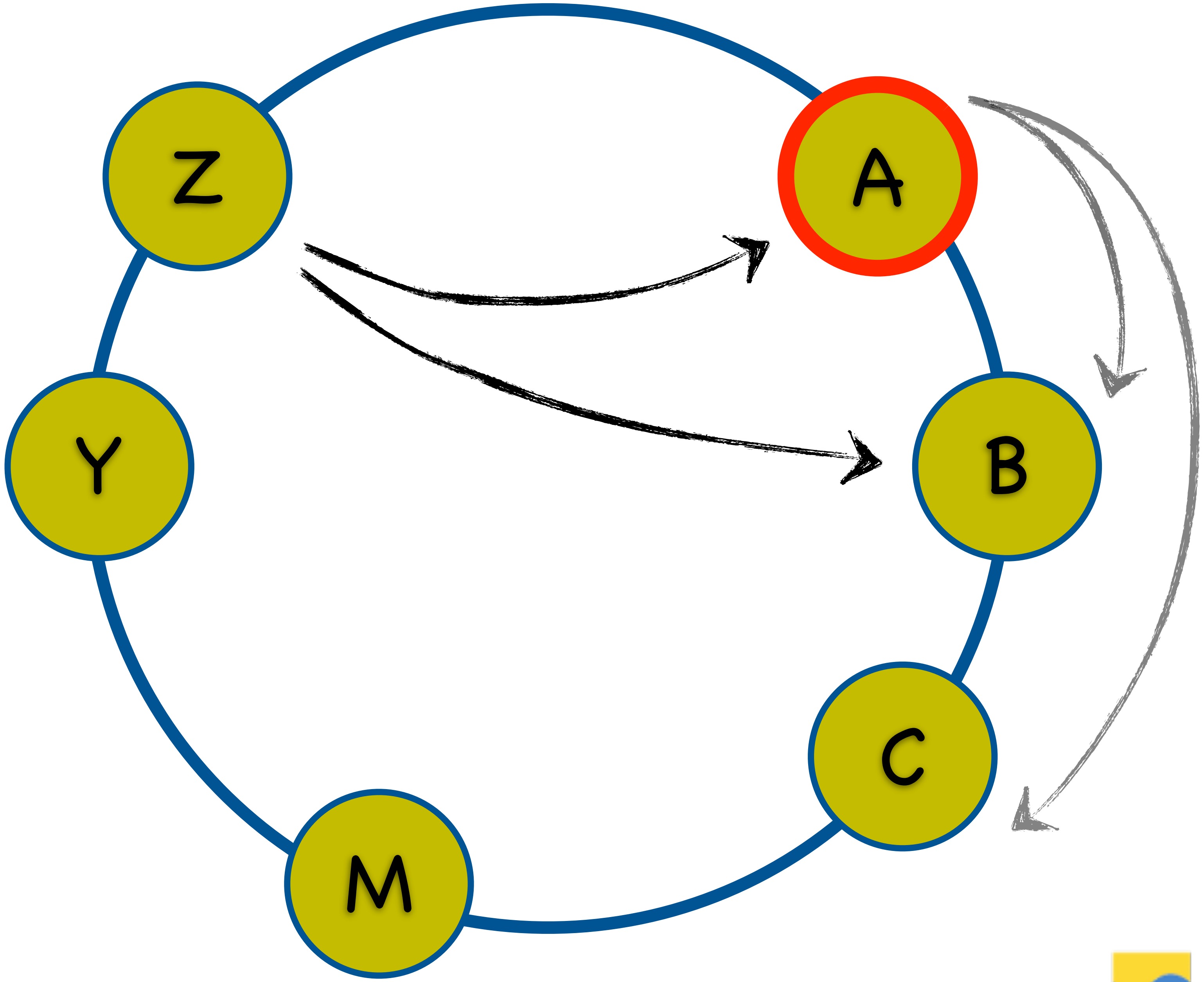
Poor load distribution

Distributing Load

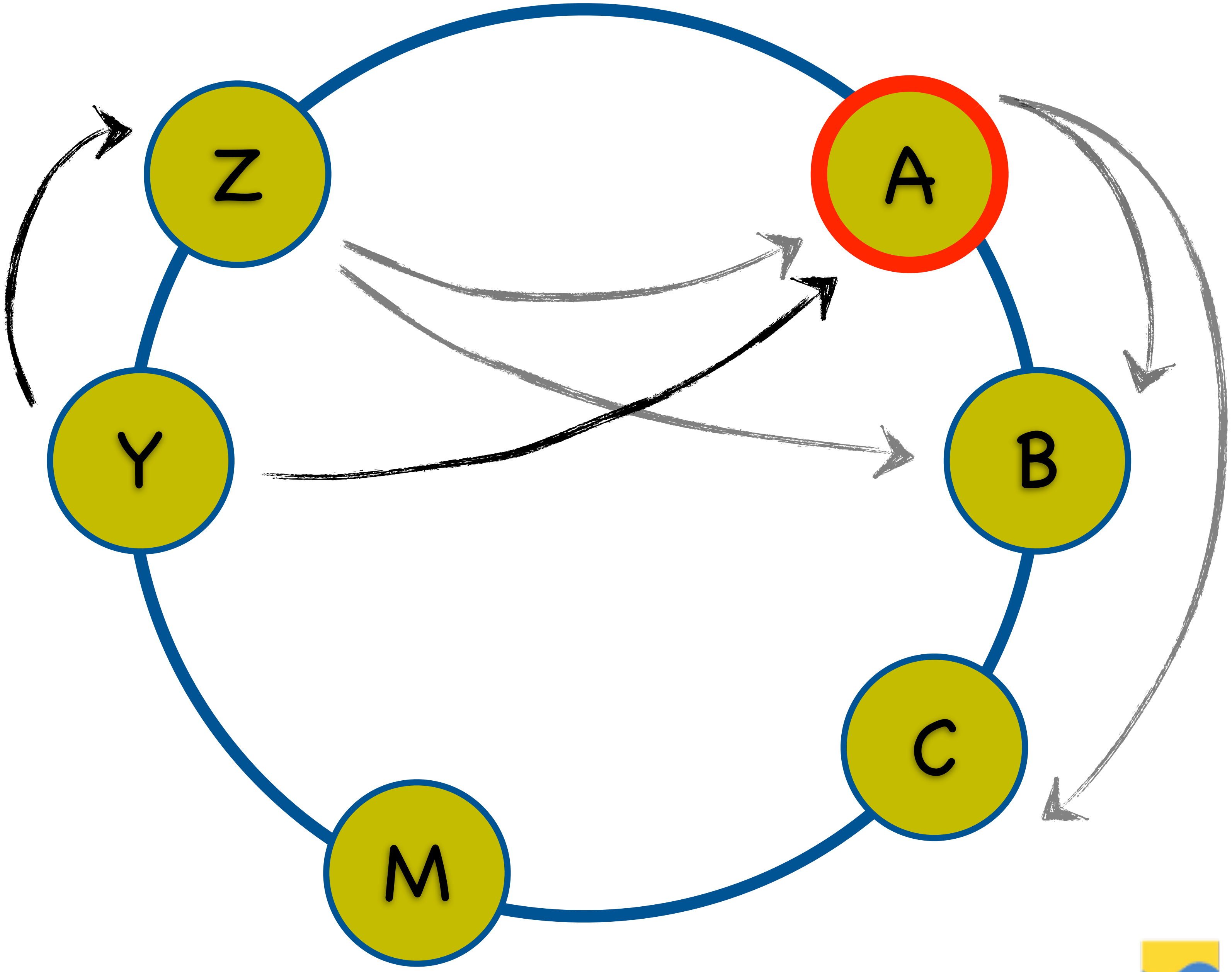


Acunu

Distributing Load

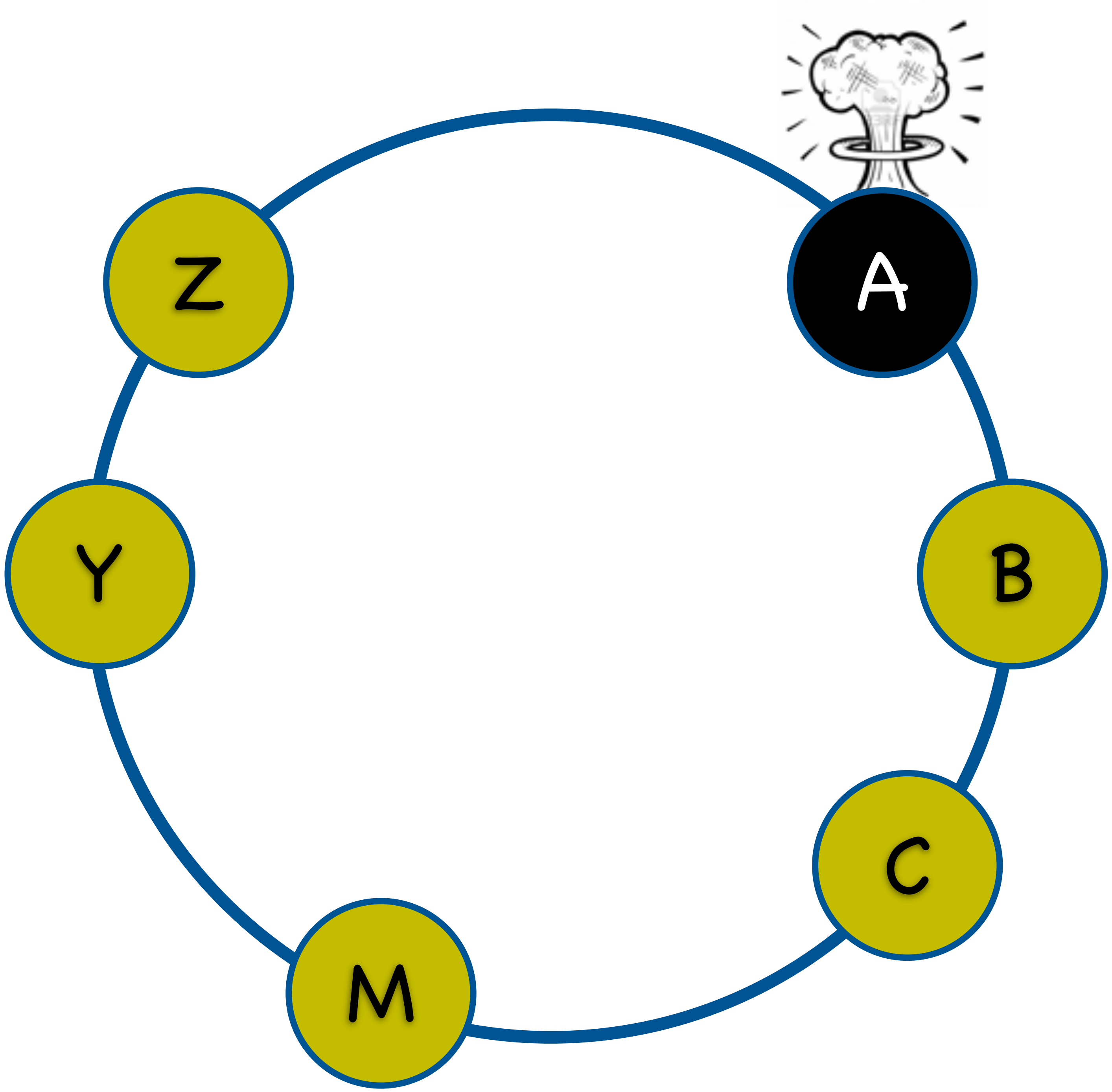


Distributing Load



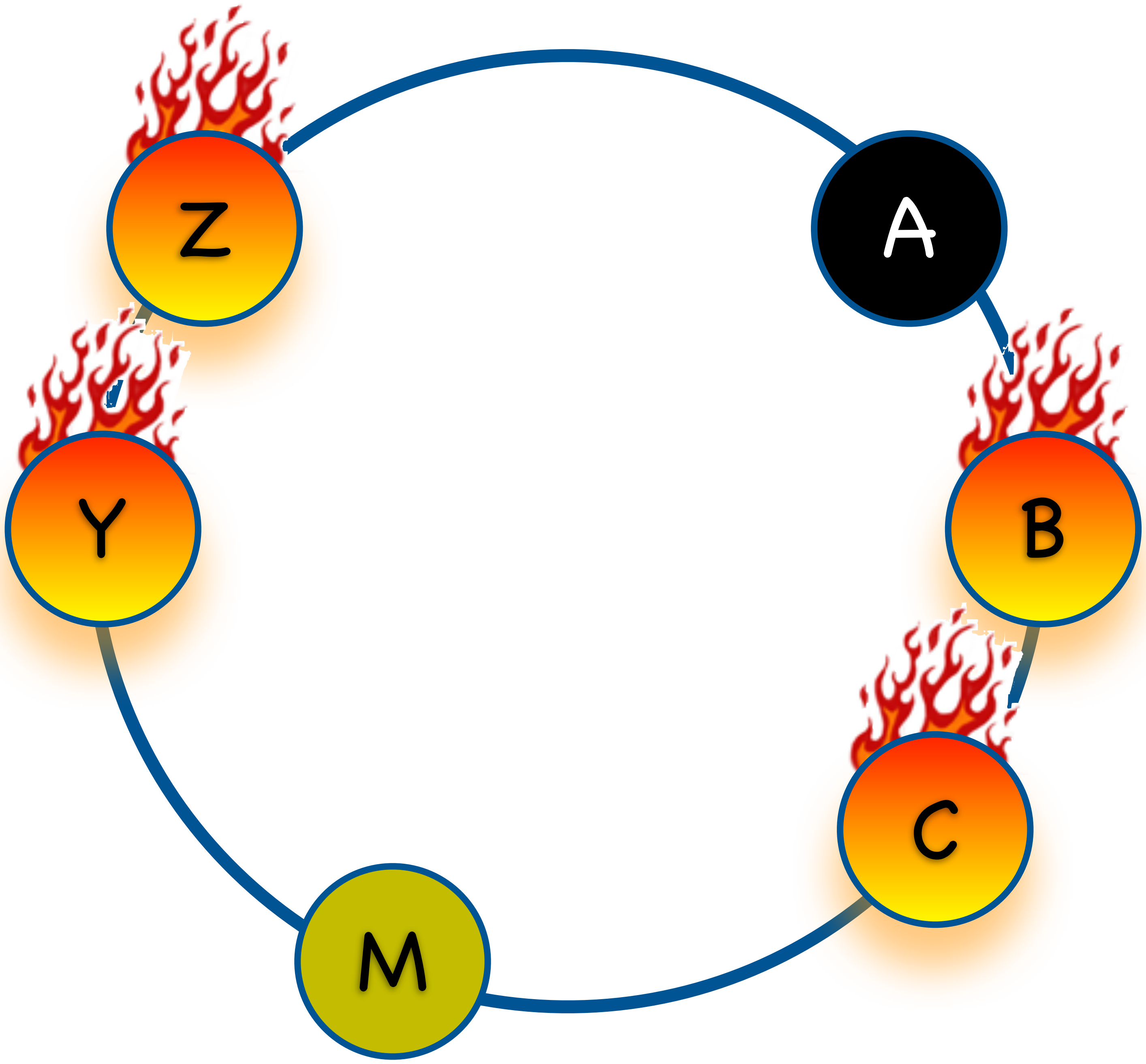
Acunu

Distributing Load



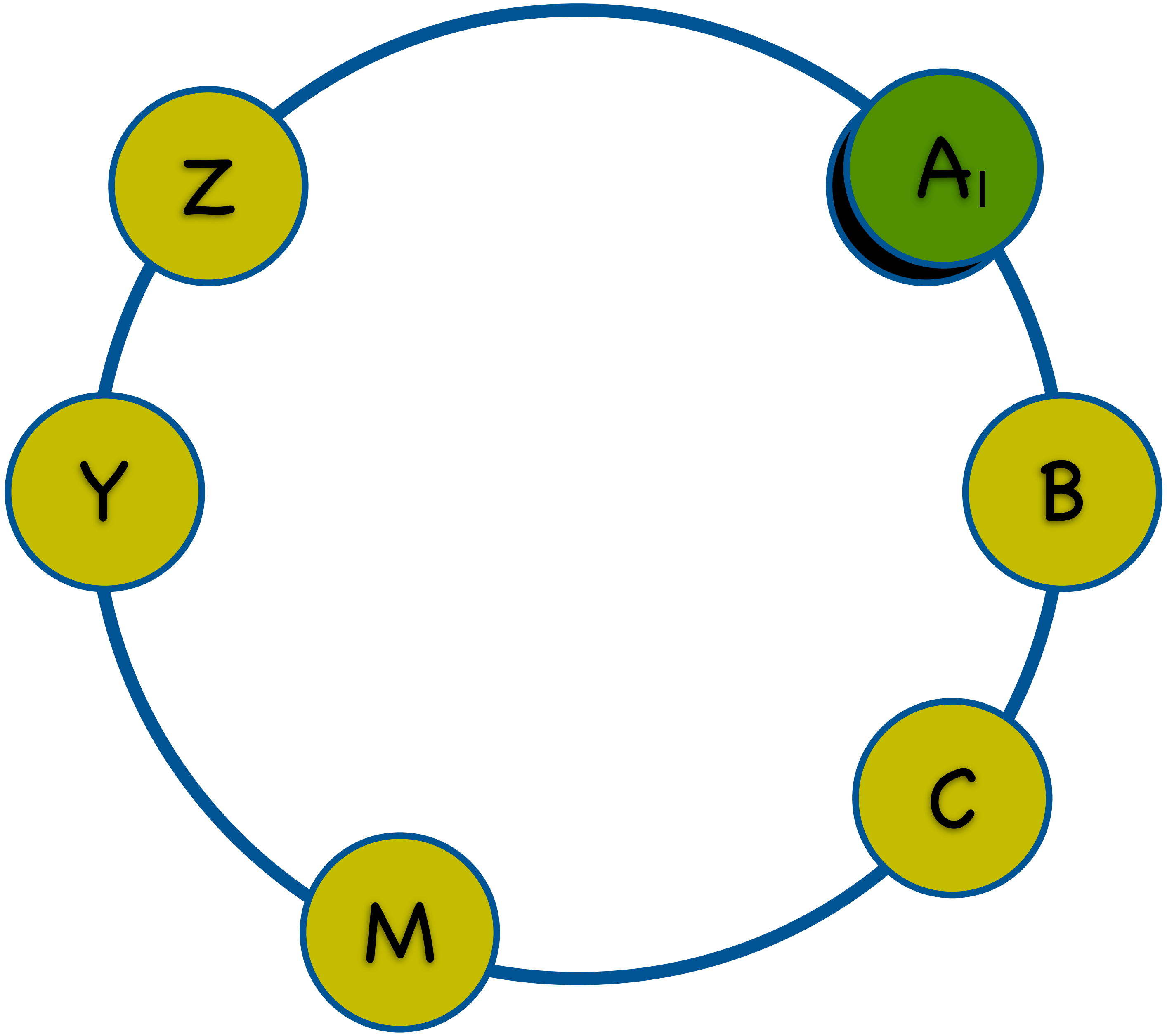
Acunu

Distributing Load



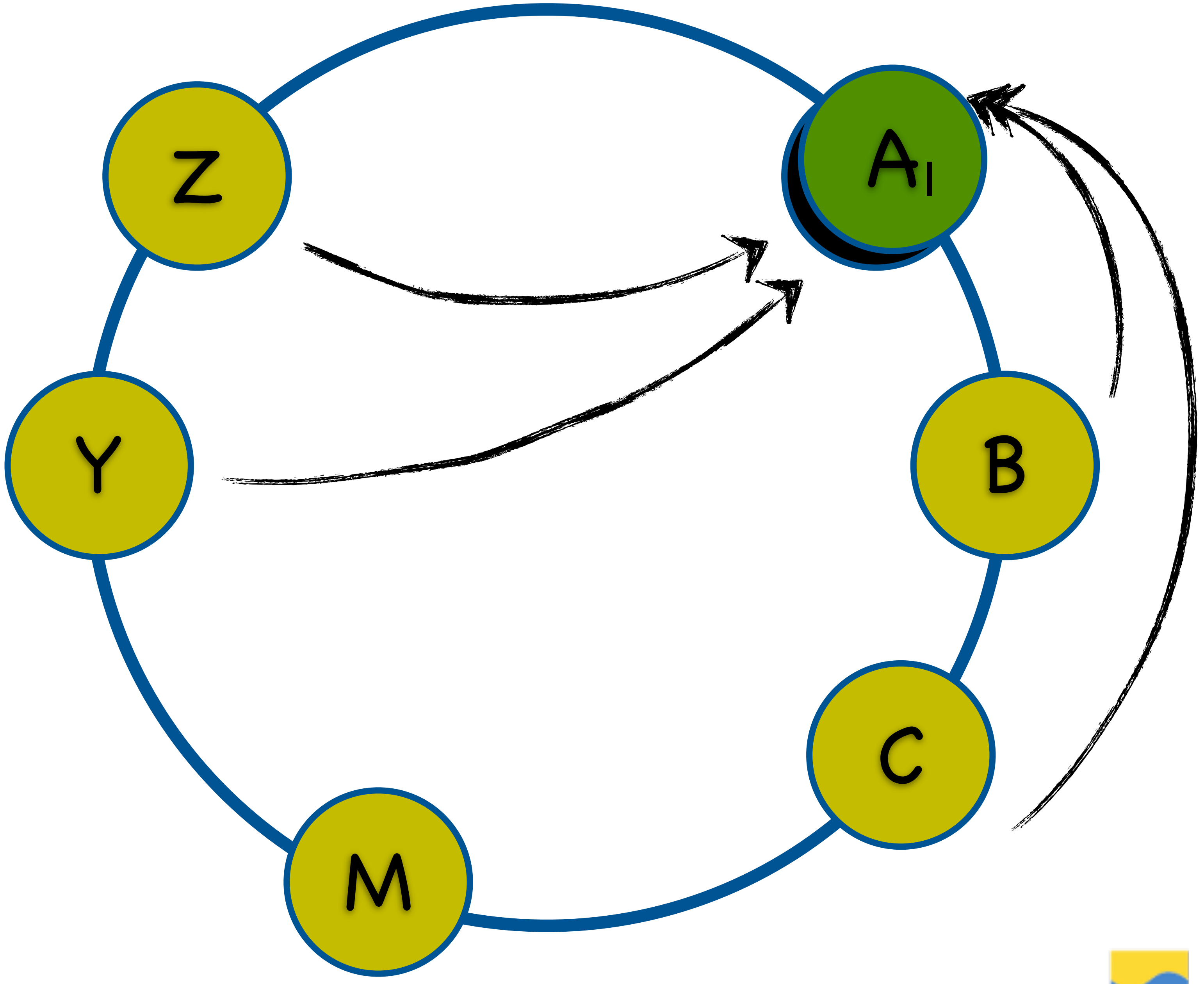
Acunu

Distributing Load



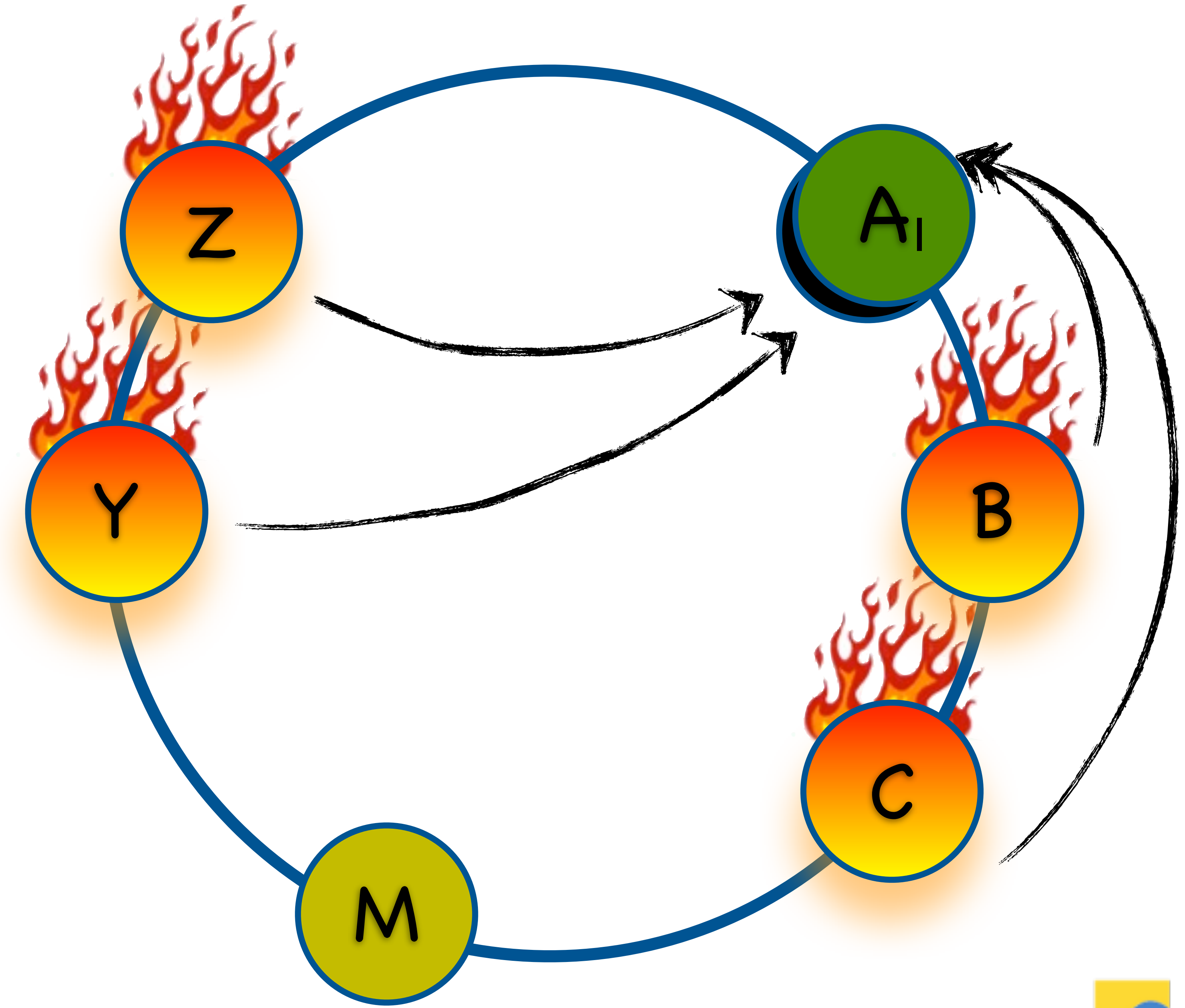
Acunu

Distributing Load



Acunu

Distributing Load

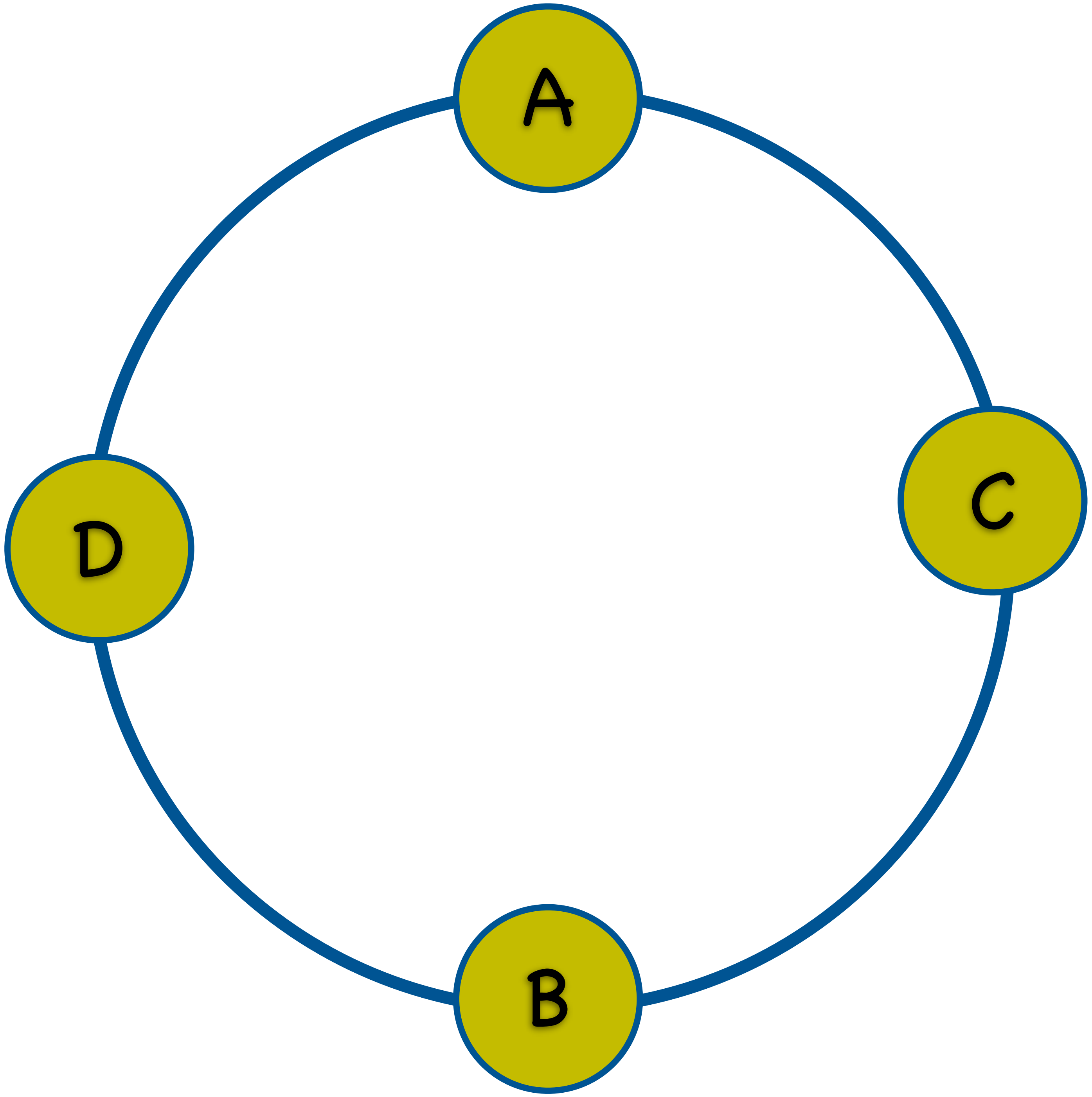


Acunu

Problem:

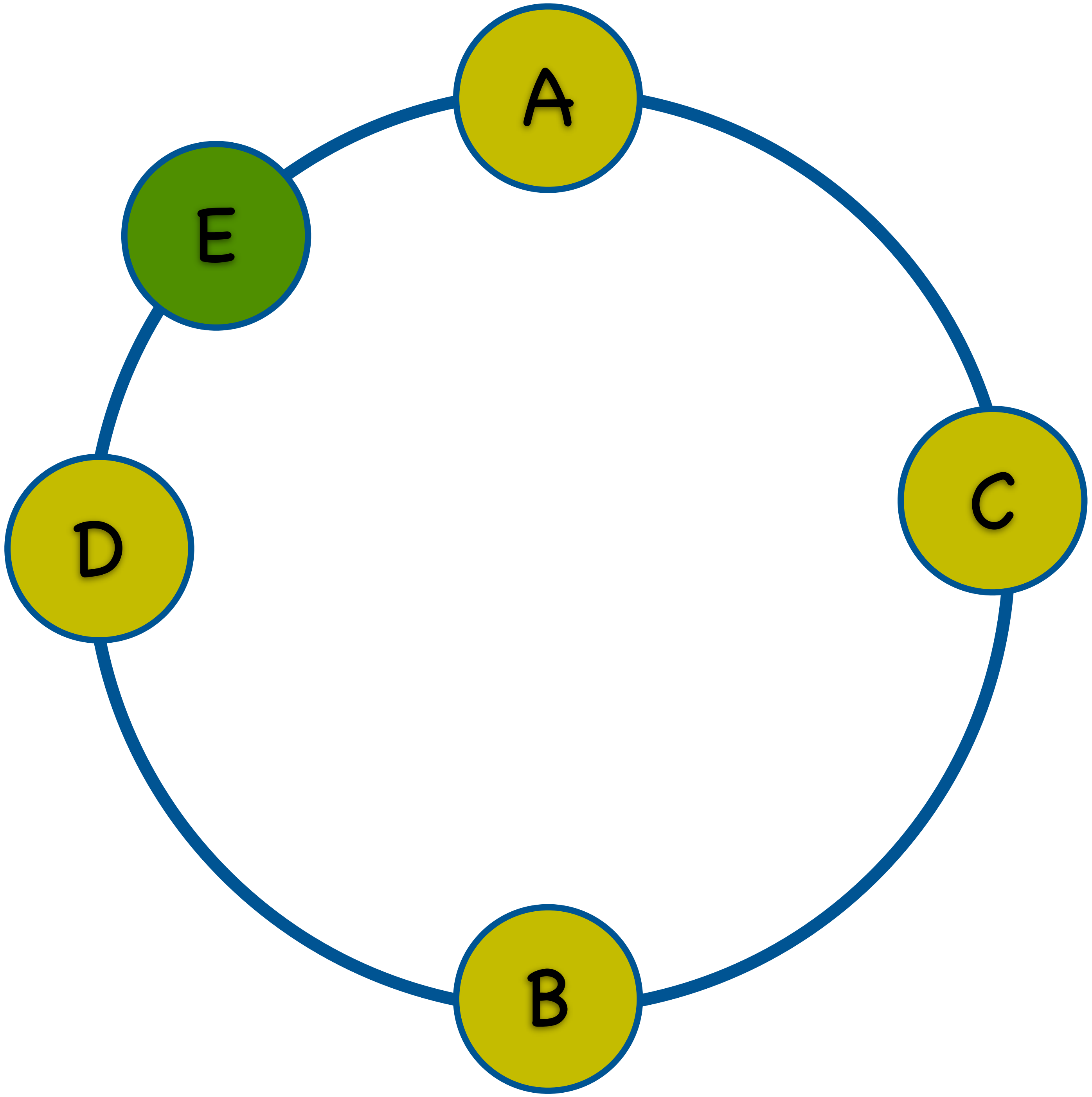
Poor data distribution

Distributing Data



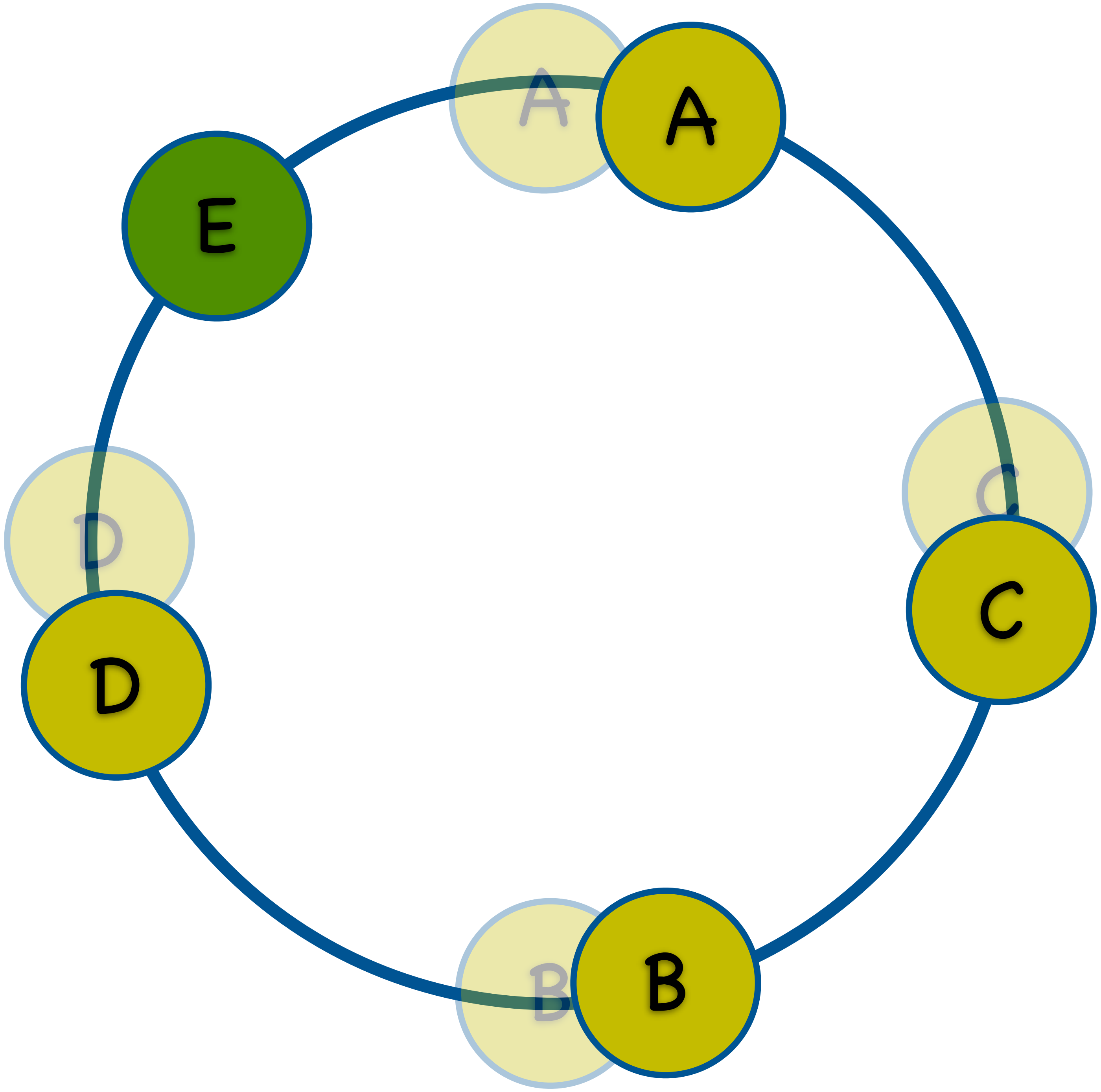
Acunu

Distributing Data

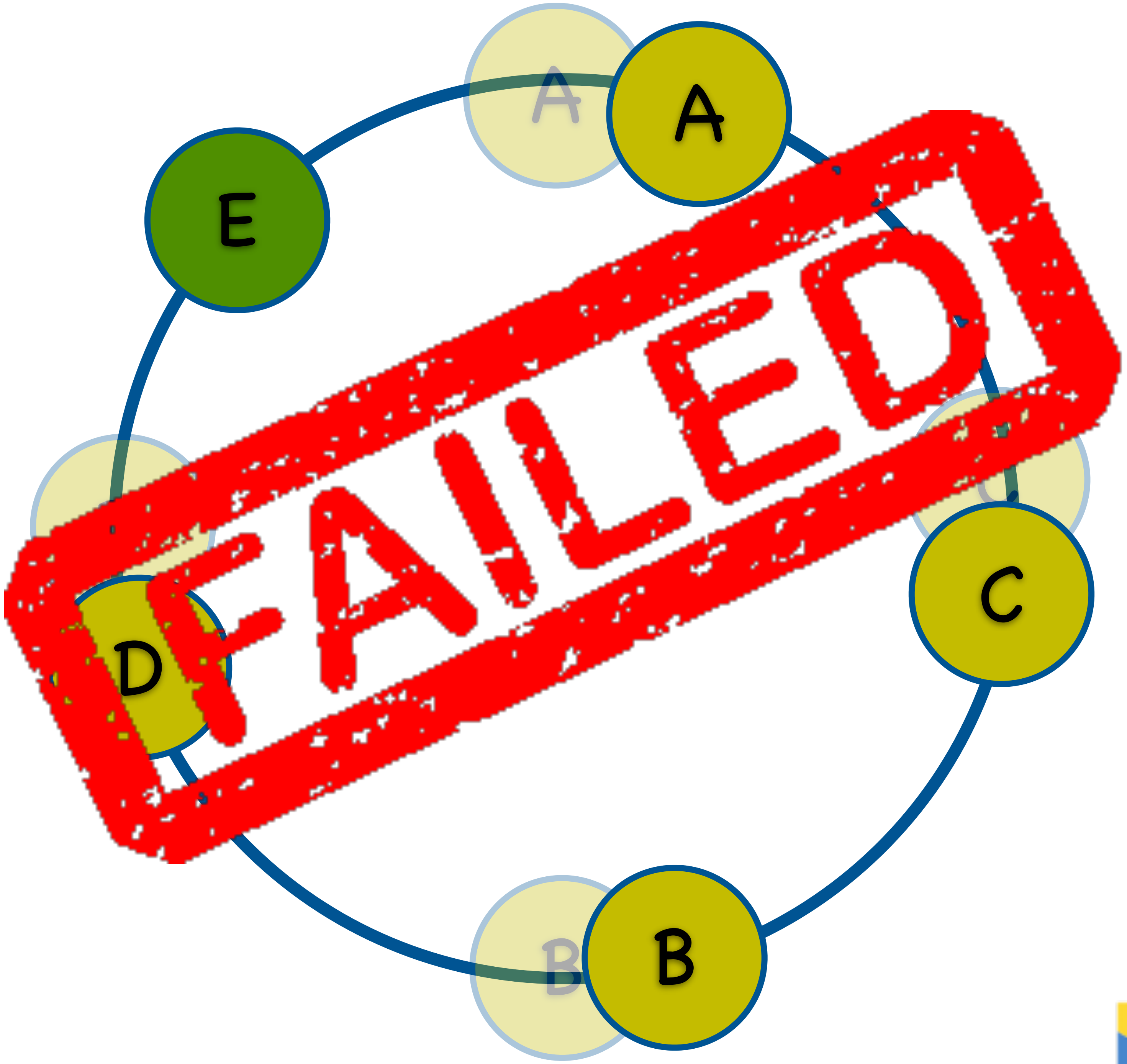


Acunu

Distributing Data

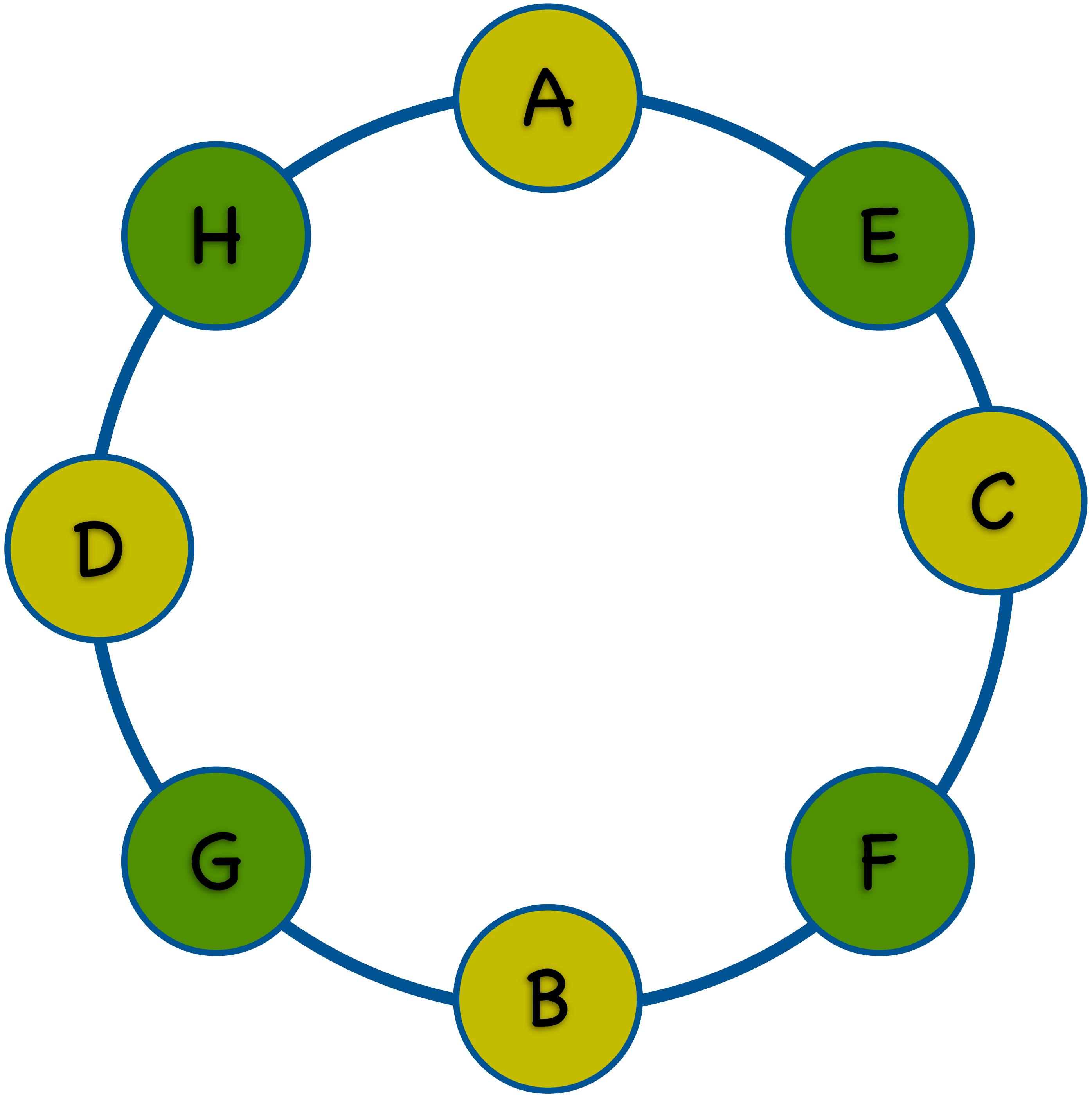


Distributing Data



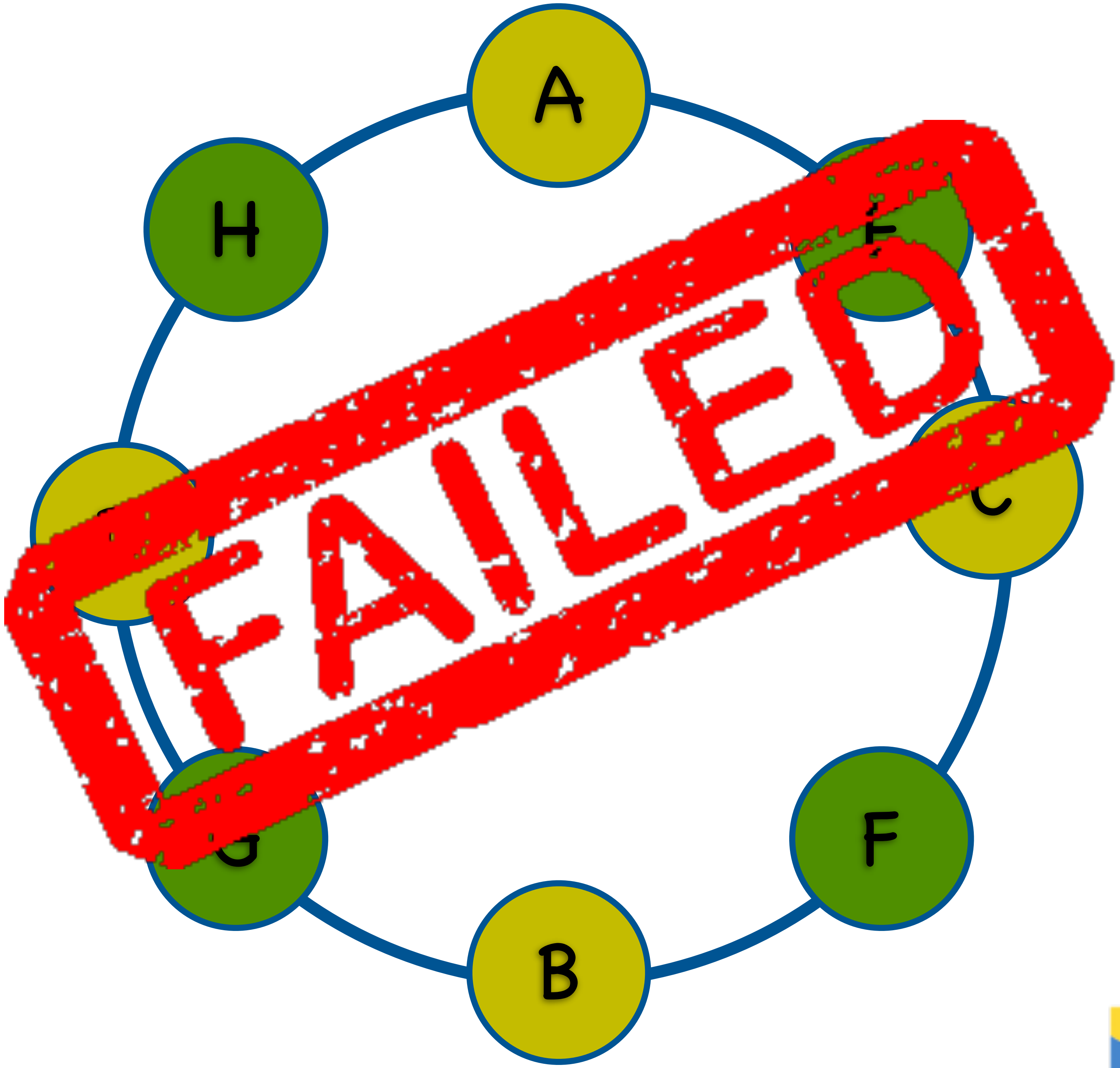
Acunu

Distributing Data



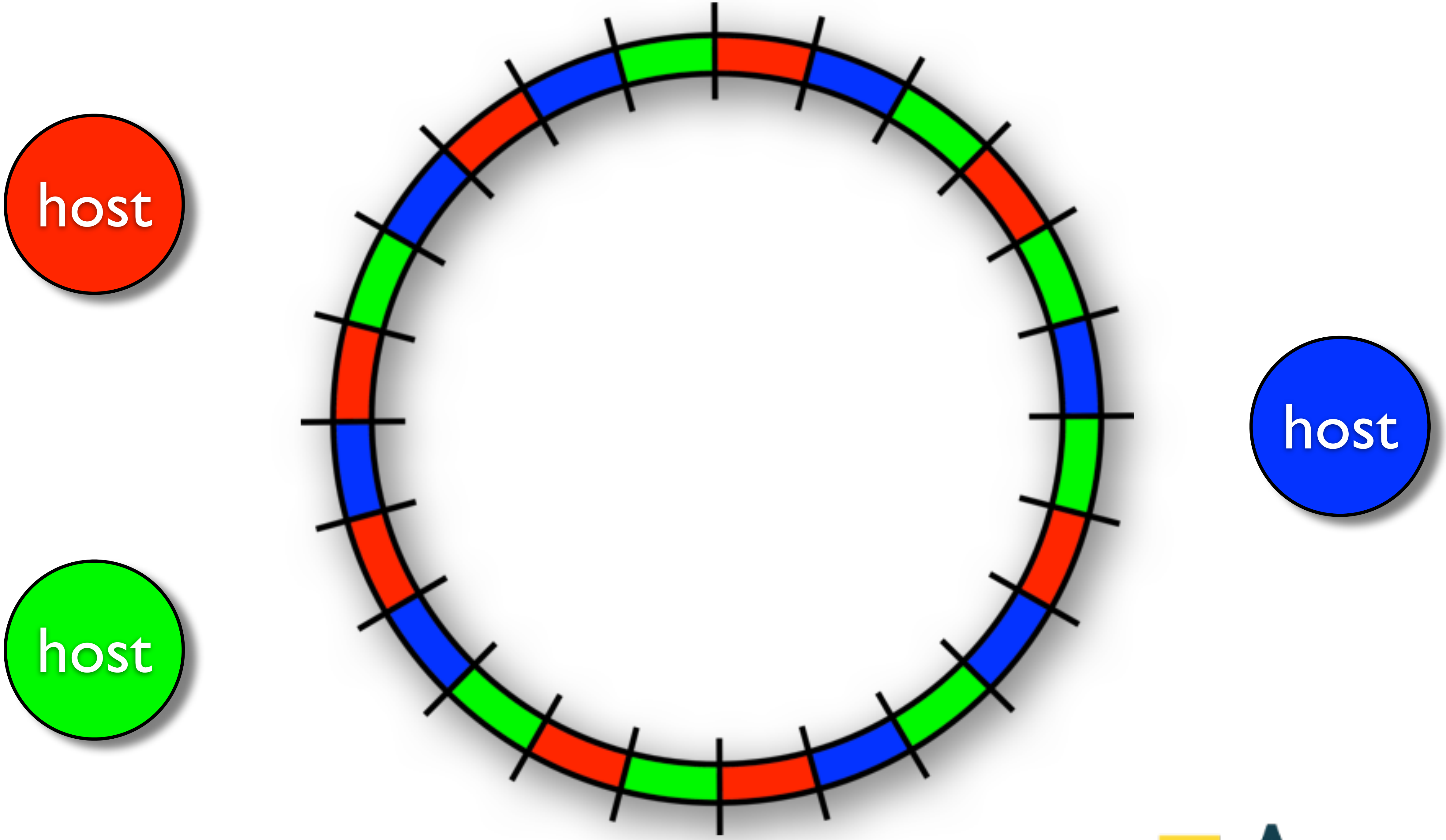
Acunu

Distributing Data



Virtual Nodes

In a nutshell...



Acunu

Benefits

- Operationally simpler (no token management)
- Better distribution of load
- Concurrent streaming involving all hosts
- Smaller partitions mean greater reliability
- Supports heterogenous hardware

Acunu

Strategies

- Automatic sharding
- Fixed partition assignment
- Random token assignment

Strategy

Automatic Sharding

- Partitions are split when data exceeds a threshold
- Newly created partitions are relocated to a host with lower data load
- Similar to sharding performed by Bigtable, or Mongo auto-sharding

Acunu

Strategy

Fixed Partition Assignment

- Namespace divided into Q evenly-sized partitions
- Q/N partitions assigned per host (where N is the number of hosts)
- Joining hosts "steal" partitions evenly from existing hosts.
- Used by Dynamo and Voldemort (described in Dynamo paper as "strategy 3")

Strategy

Random Token Assignment

- Each host assigned T random tokens
- T random tokens generated for joining hosts; New tokens divide existing ranges
- Similar to libketama; Identical to Classic Cassandra when T=1

Acunu

Considerations

1. Number of partitions
2. Partition size
3. How 1 changes with more nodes and data
4. How 2 changes with more nodes and data

Acunu

Evaluating

Strategy	No. Partitions	Partition size
Random	O(N)	O(B/N)
Fixed	O(1)	O(B)
Auto-sharding	O(B)	O(1)

B ~ total data size, **N** ~ number of hosts

Acunu

Evaluating

- Automatic sharding
  - partition size constant (great)
  - number of partitions scales linearly with data size (bad)
- Fixed partition assignment
- Random token assignment

Evaluating

- Automatic sharding
- Fixed partition assignment
    - Number of partitions is constant (good)
    - Partition size scales linearly with data size (bad)
    - Higher operational complexity (bad)
- Random token assignment

Evaluating

- Automatic sharding
- Fixed partition assignment
- Random token assignment
  - Number of partitions scales linearly with number of hosts (~~good~~ ok)
  - Partition size increases with more data; decreases with more hosts (good)

Evaluating

- Automatic sharding
- Fixed partition assignment
- Random token assignment ✓

Cassandra

Configuration

conf/cassandra.yaml

```
# Comma separated list of tokens,
# (new installs only).
initial_token:<token>,<token>,<token>

or

# Number of tokens to generate.
num_tokens: 256
```

Configuration

nodetool info

```
Token            : (invoke with -T/--tokens to see all 256 tokens)
ID               : 64090651-6034-41d5-bfc6-ddd24957f164
Gossip active    : true
Thrift active    : true
Load             : 92.69 KB
Generation No    : 1351030018
Uptime (seconds): 45
Heap Memory (MB): 95.16 / 1956.00
Data Center      : datacenter1
Rack             : rack1
Exceptions       : 0
Key Cache        : size 240 (bytes), capacity 101711872 (bytes ...
Row Cache        : size 0 (bytes), capacity 0 (bytes), 0 hits, ...
```

Acunu

Configuration

nodetool ring

```
Datacenter: datacenter1
==========
Replicas: 2

Address      Rack    Status State   Load        Owns      Token
                                                          9022770486425350384
127.0.0.1    rack1   Up     Normal  97.24 KB    66.03%    -9182469192098976078
127.0.0.1    rack1   Up     Normal  97.24 KB    66.03%    -9054823614314102214
127.0.0.1    rack1   Up     Normal  97.24 KB    66.03%    -8970752544645156769
127.0.0.1    rack1   Up     Normal  97.24 KB    66.03%    -8927190060345427739
127.0.0.1    rack1   Up     Normal  97.24 KB    66.03%    -8880475677109843259
127.0.0.1    rack1   Up     Normal  97.24 KB    66.03%    -8817876497520861779
127.0.0.1    rack1   Up     Normal  97.24 KB    66.03%    -8810512134942064901
127.0.0.1    rack1   Up     Normal  97.24 KB    66.03%    -8661764562509480261
127.0.0.1    rack1   Up     Normal  97.24 KB    66.03%    -8641550925069186492
127.0.0.1    rack1   Up     Normal  97.24 KB    66.03%    -8636224350654790732
...
...
```

Configuration

nodetool status

```
Datacenter: datacenter1
=======================
Status=Up/Down
|/ State=Normal/Leaving/Joining/Moving
-- Address   Load     Tokens  Owns   Host ID                               Rack
UN 10.0.0.1  97.2 KB  256     66.0%  64090651-6034-41d5-bfc6-ddd24957f164  rack1
UN 10.0.0.2  92.7 KB  256     66.2%  b3c3b03c-9202-4e7b-811a-9de89656ec4c  rack1
UN 10.0.0.3  92.6 KB  256     67.7%  e4eef159-cb77-4627-84c4-14efbc868082  rack1
```

Configuration

nodetool status

```
Datacenter: datacenter1
=======================
Status=Up/Down
|/ State=Normal/Leaving/Joining/Moving
-- Address   Load     Tokens Owns   Host ID                               Rack
UN 10.0.0.1  97.2 KB  256    66.0%  64090651-6034-41d5-bfc6-ddd24957f164  rack1
UN 10.0.0.2  92.7 KB  256    66.2%  b3c3b03c-9202-4e7b-811a-9de89656ec4c  rack1
UN 10.0.0.3  92.6 KB  256    67.7%  e4eef159-cb77-4627-84c4-14efbc868082  rack1
```

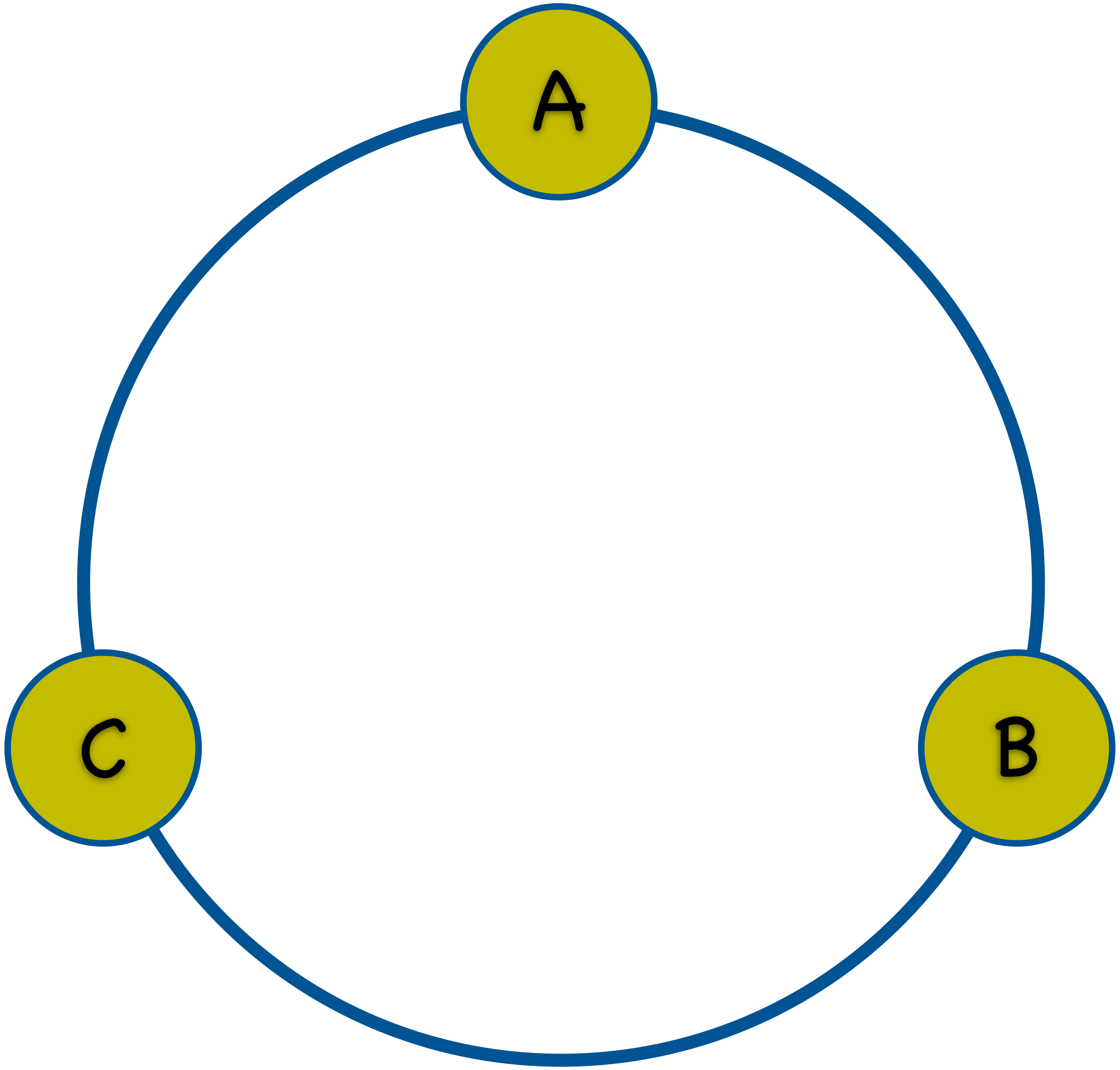
Acunu

Configuration

nodetool status

```
Datacenter: datacenter1
=======================
Status=Up/Down
|/ State=Normal/Leaving/Joining/Moving
-- Address    Load     Tokens  Owns   Host ID                               Rack
UN 10.0.0.1   97.2 KB  256     66.0%  64090651-6034-41d5-bfc6-ddd24957f164  rack1
UN 10.0.0.2   92.7 KB  256     66.2%  b3c3b03c-9202-4e7b-811a-9de89656ec4c  rack1
UN 10.0.0.3   92.6 KB  256     67.7%  e4eef159-cb77-4627-84c4-14efbc868082  rack1
```

Migration



Acunu

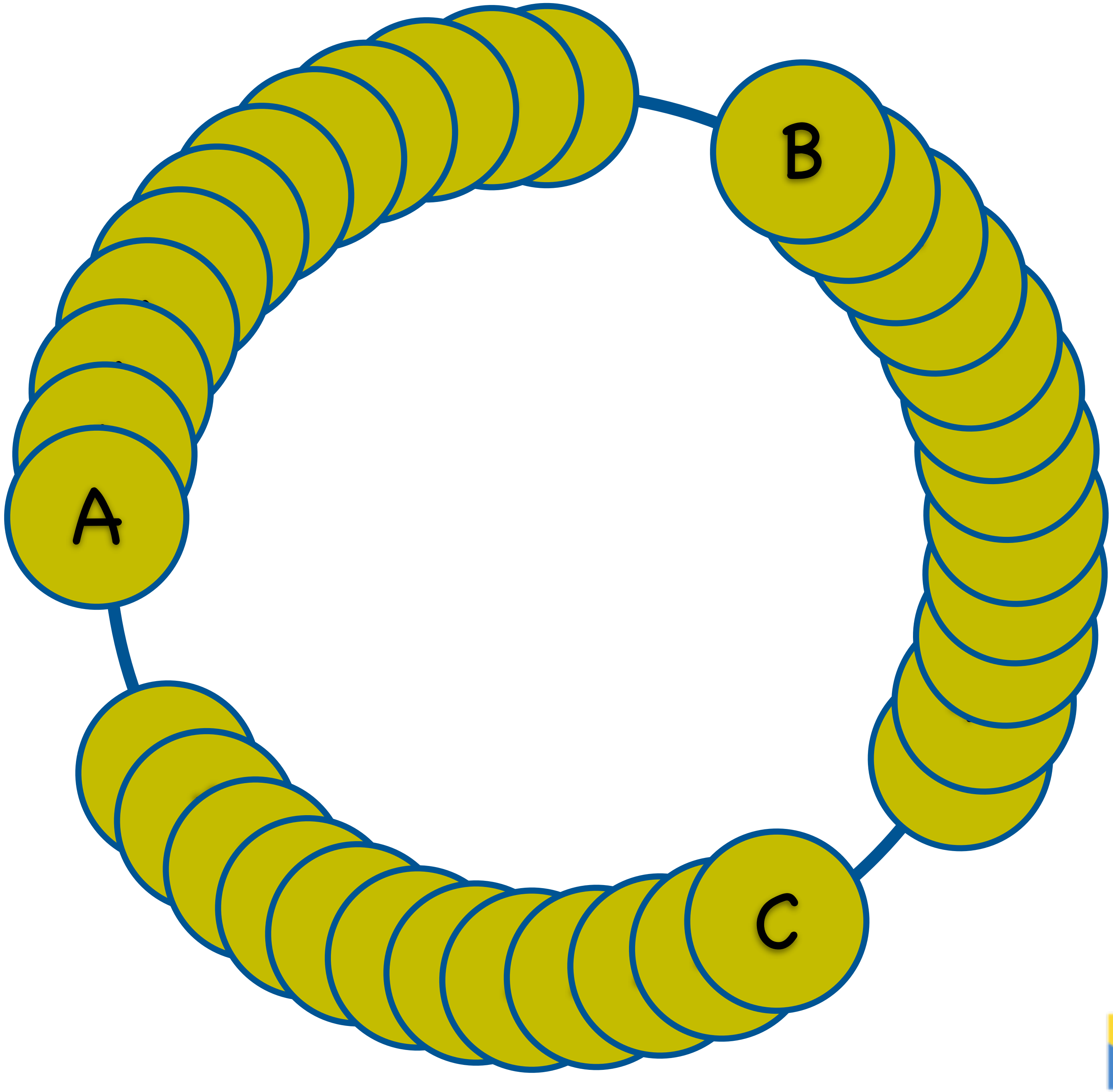
Migration

edit conf/cassandra.yaml and restart

```
# Number of tokens to generate.
num_tokens: 256
```

Migration

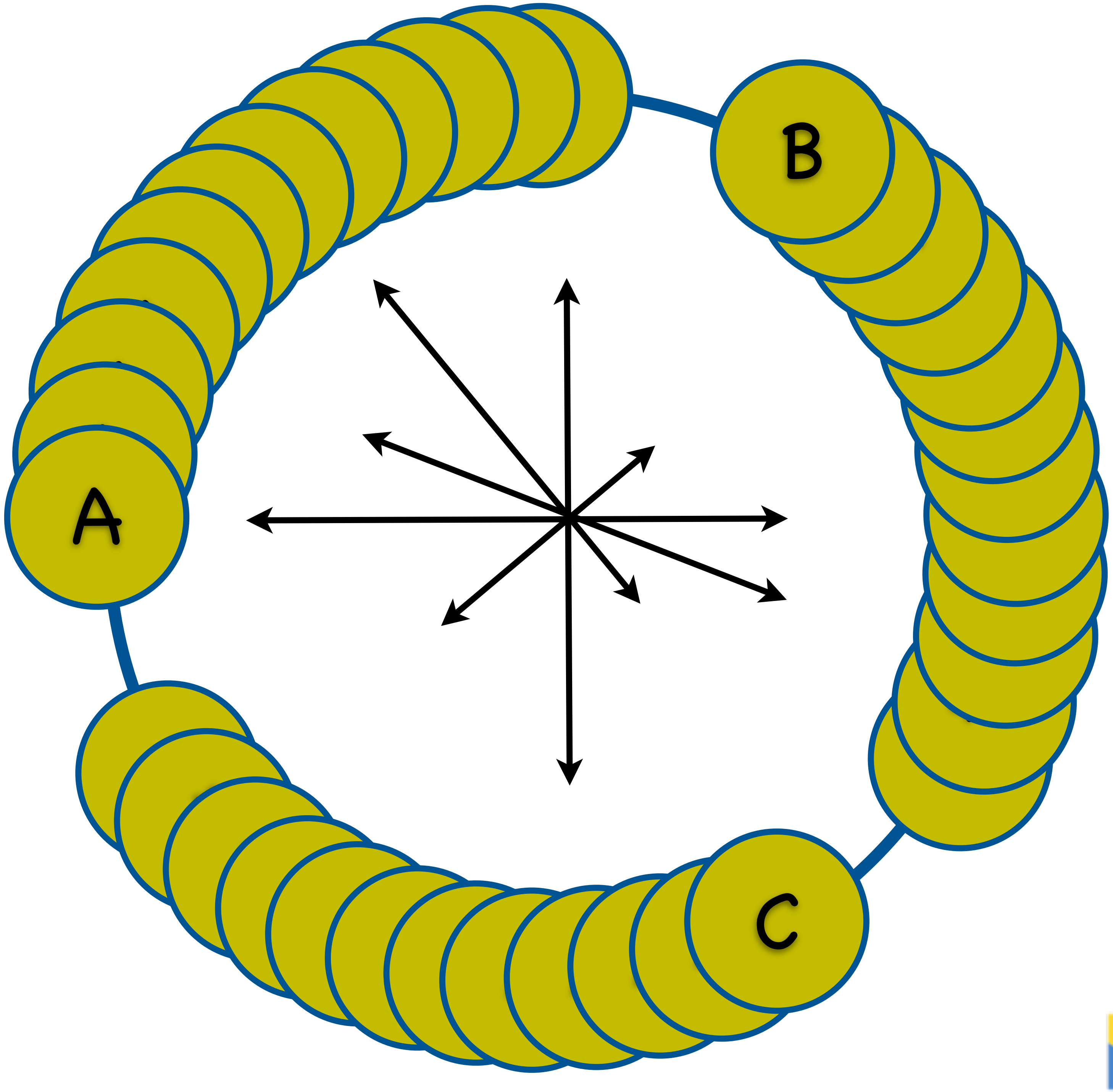
convert to T contiguous tokens in existing ranges



Acunu

Migration

shuffle



Acunu

Shuffle

- Range transfers are queued on each host
- Hosts initiate transfer of ranges to self
- Pay attention to the logs!

Shuffle

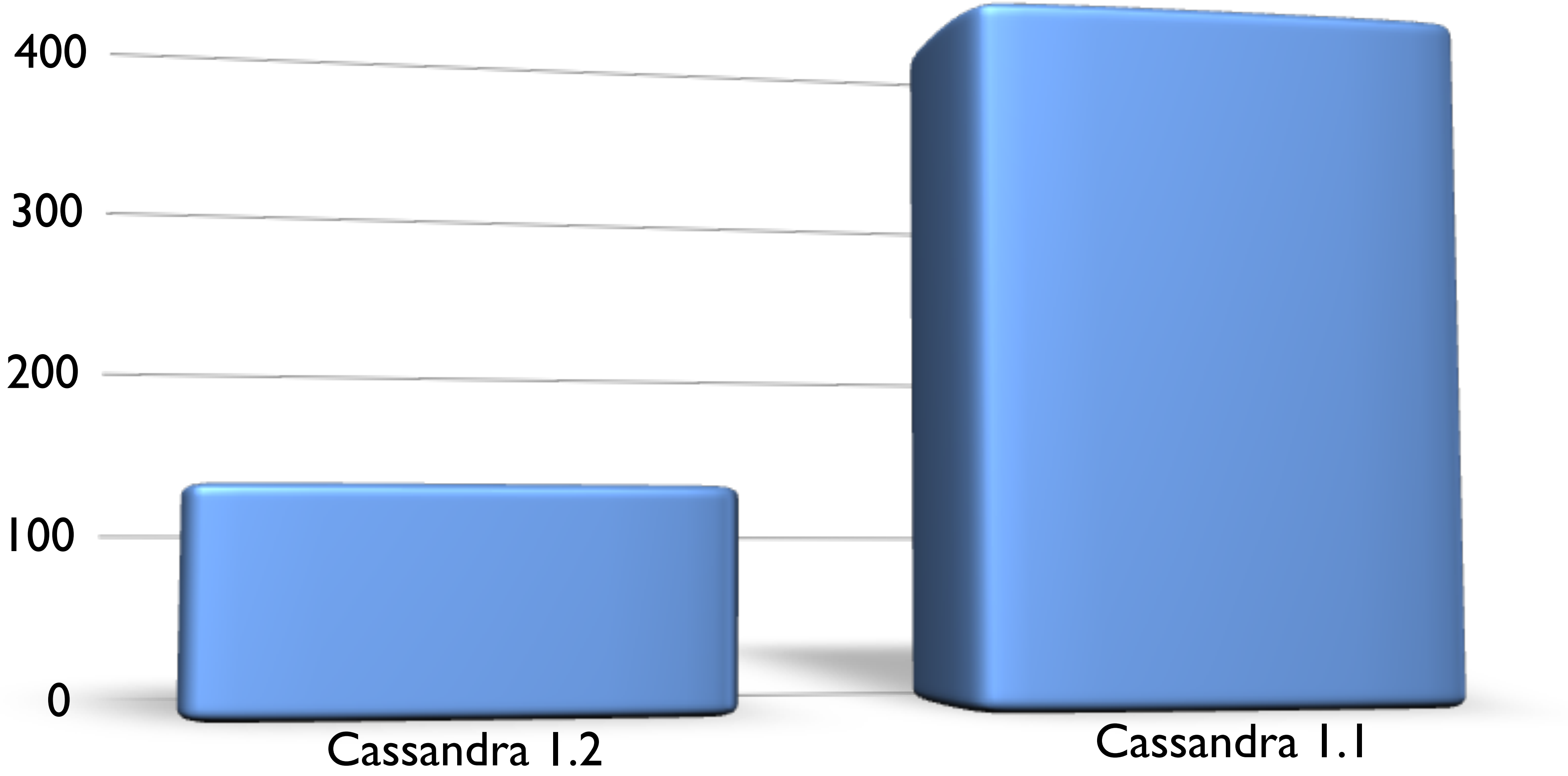
bin/shuffle

```
Usage: shuffle [options] <sub-command>

Sub-commands:
 create           Initialize a new shuffle operation
 ls               List pending relocations
 clear            Clear pending relocations
 en[able]         Enable shuffling
 dis[able]        Disable shuffling

Options:
 -dc,  --only-dc          Apply only to named DC (create only)
 -tp,  --thrift-port      Thrift port number (Default: 9160)
 -p,   --port             JMX port number (Default: 7199)
 -tf,  --thrift-framed    Enable framed transport for Thrift (Default: false)
 -en,  --and-enable       Immediately enable shuffling (create only)
 -H,   --help             Print help information
 -h,   --host             JMX hostname or IP address (Default: localhost)
 -th,  --thrift-host      Thrift hostname or IP address (Default: JMX host)
```

Acunu

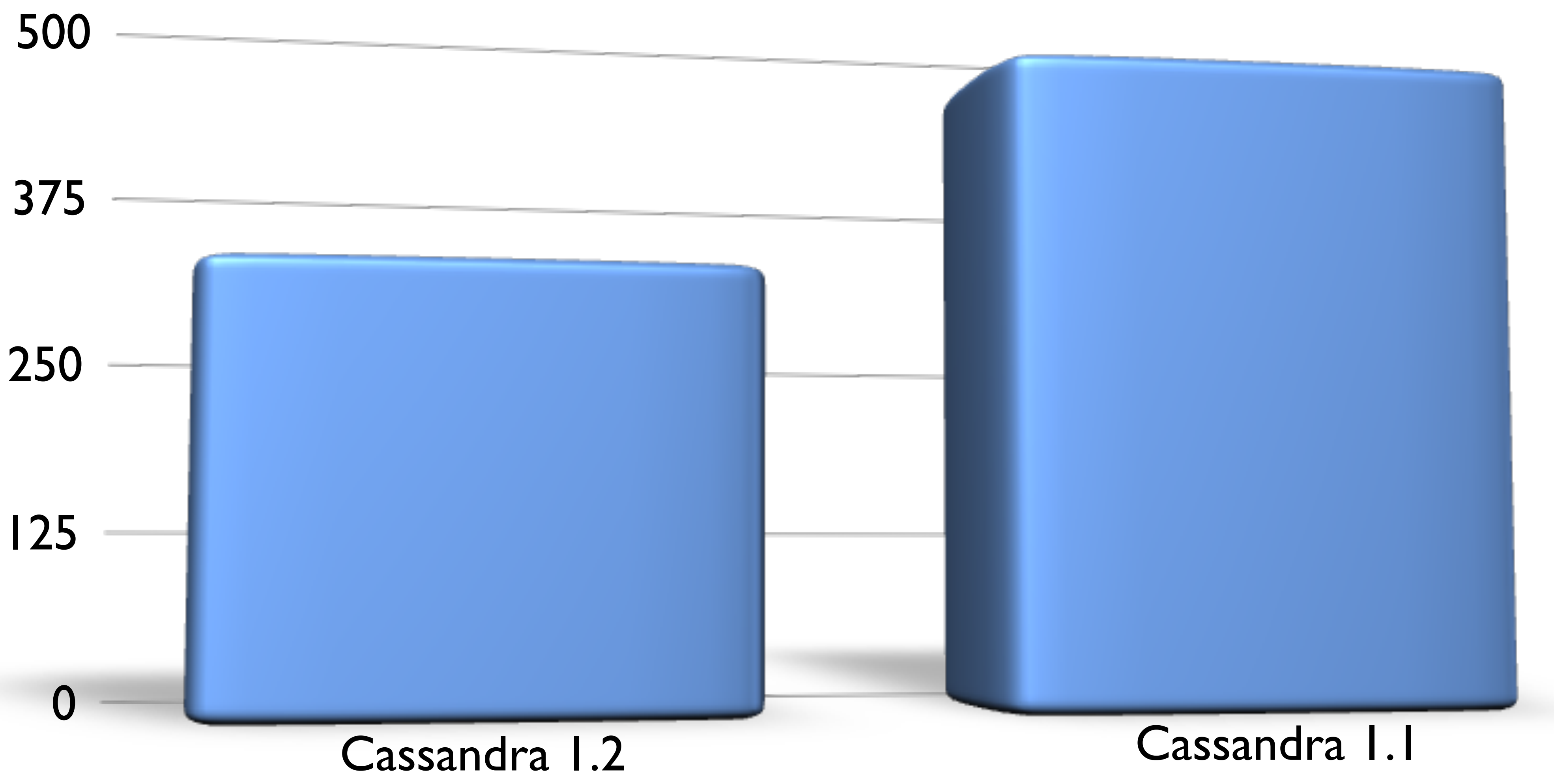
Performance

removenode



Acunu

bootstrap



The End

- Giuseppe DeCandia, Deniz Hastorun, Madan Jampani, Gunavardhan Kakulapati, Avinash Lakshman, Alex Pilchin, Swaminathan Sivasubramanian, Peter Vosshall and Werner Vogels "Dynamo: Amazon's Highly Available Key-value Store" Web.

- Low, Richard. "Improving Cassandra's uptime with virtual nodes" Web.

- Overton, Sam. "Virtual Nodes Strategies." Web.

- Overton, Sam. "Virtual Nodes: Performance Results." Web.

- Jones, Richard. "libketama - a consistent hashing algo for memcache clients" Web.