Building Scalable Stateful Services

StrangeLoop 2015

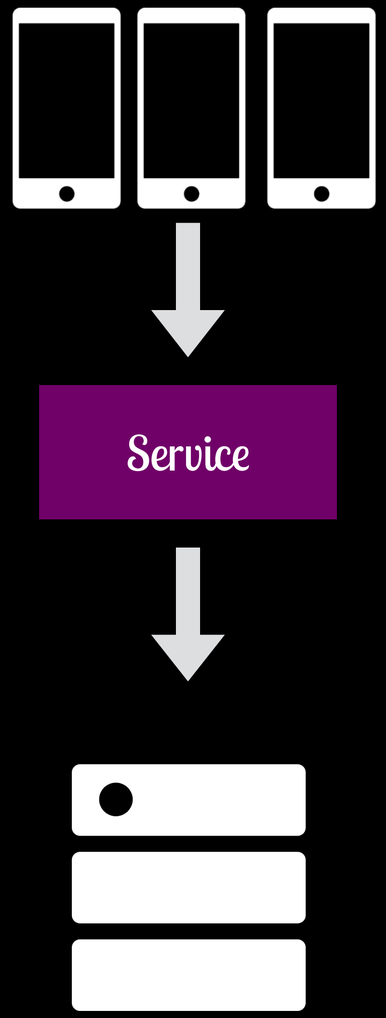
Caitie McCaffrey

Distributed Systems Engineer
Tech Lead Observability @ Twitter

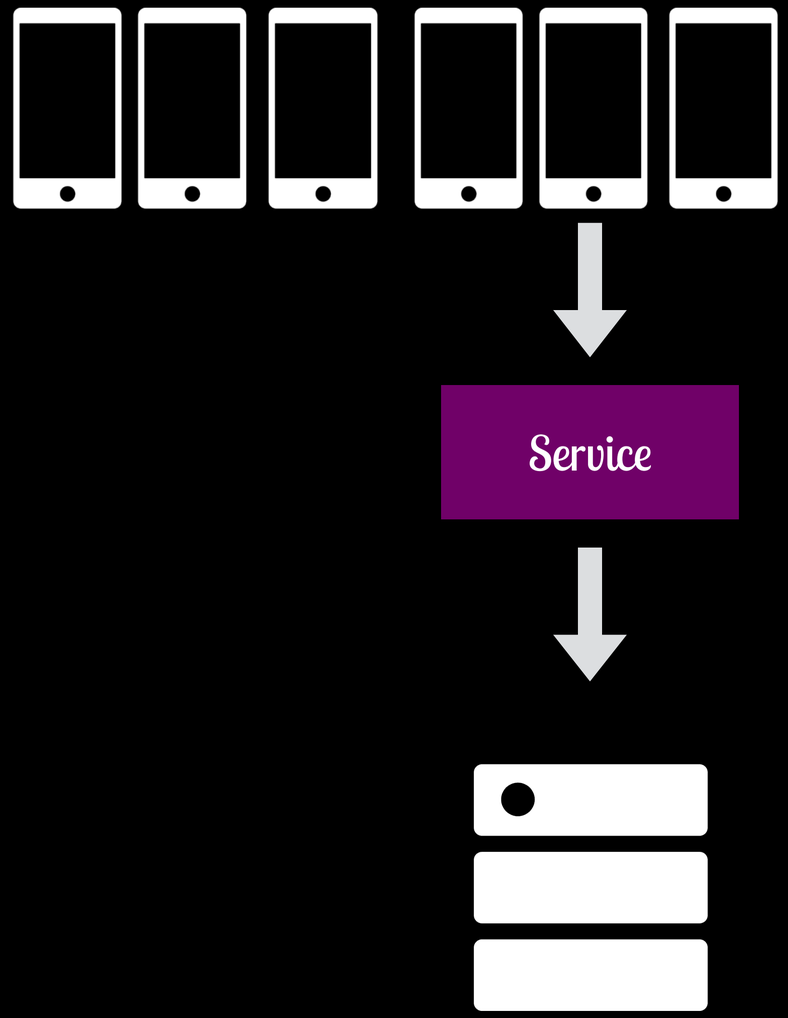
@caitie

caitiem.com

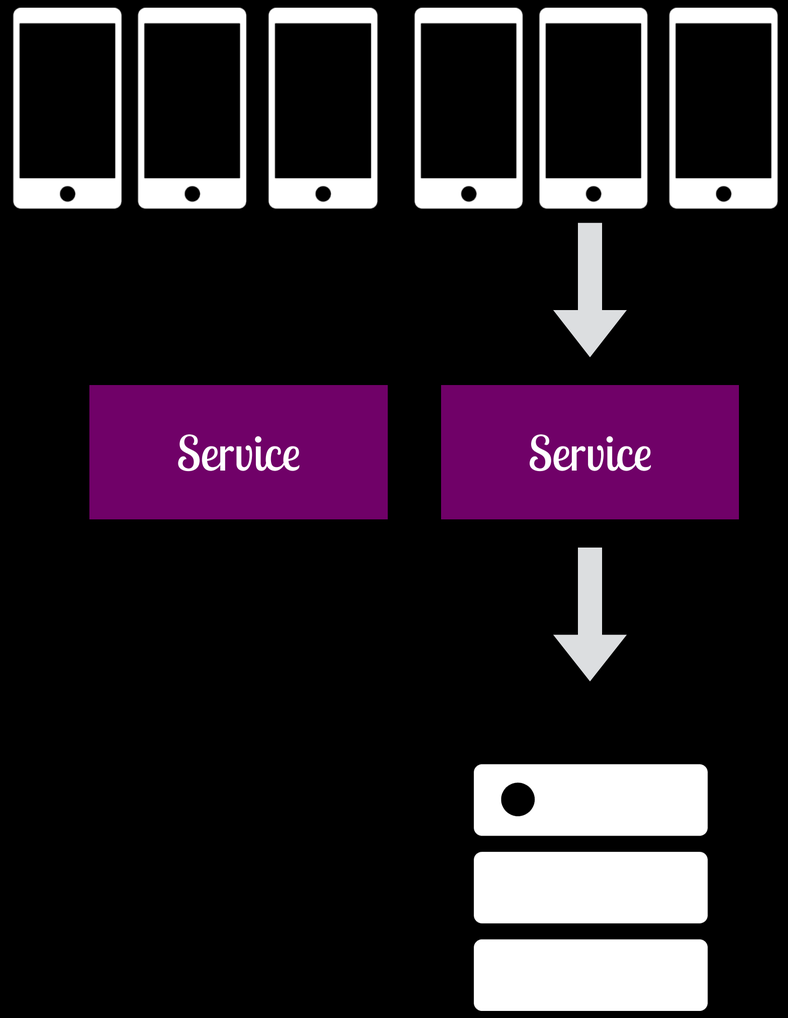
Stateless Services



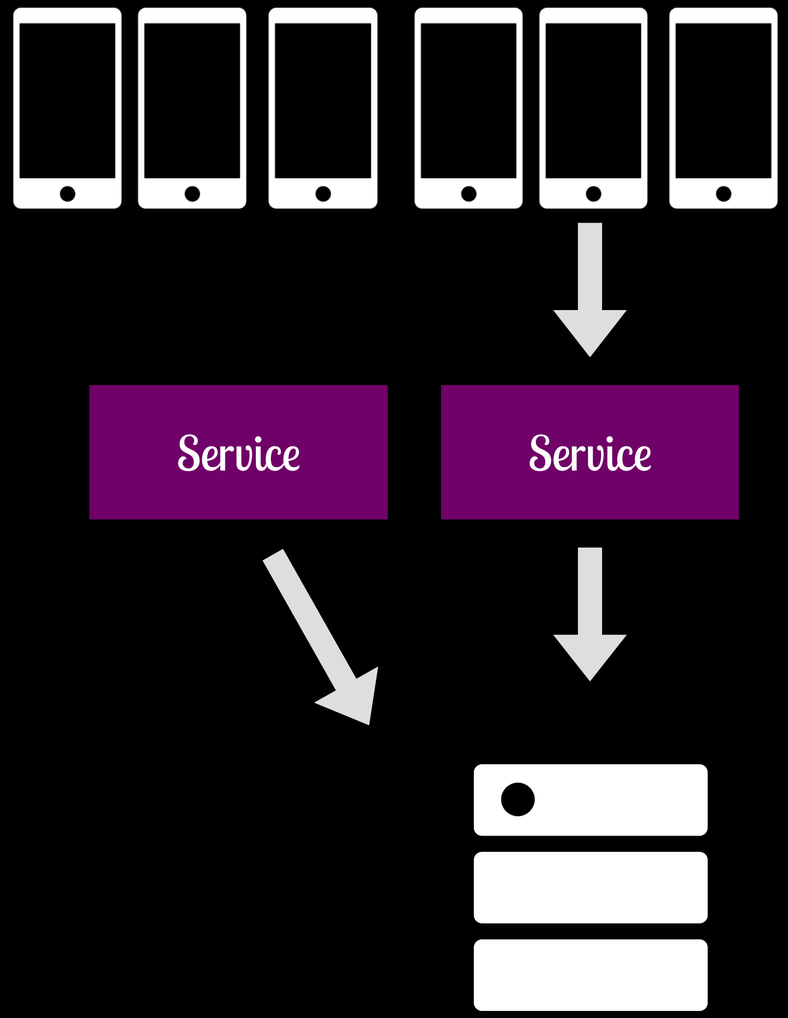
Stateless Services



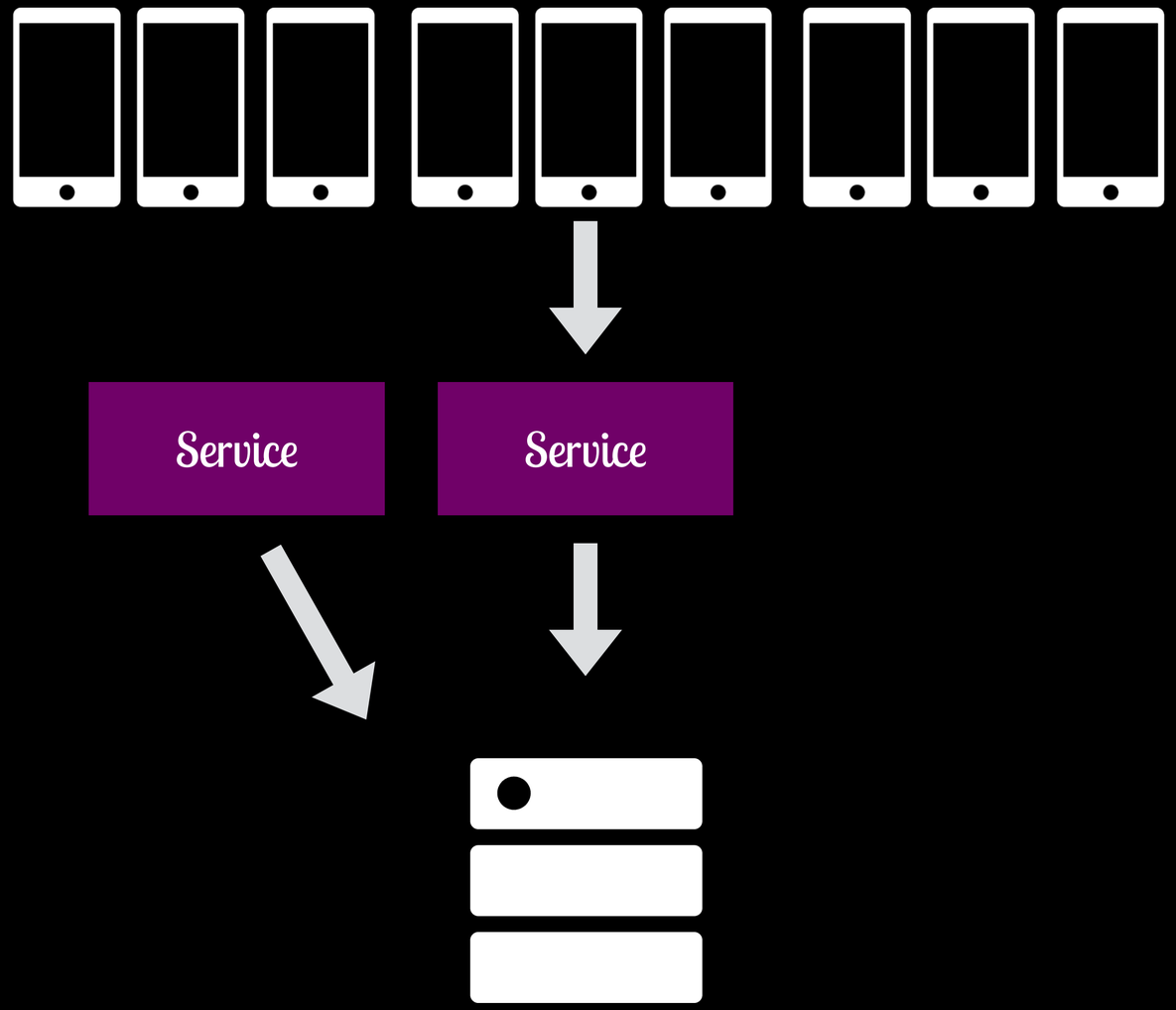
Stateless Services



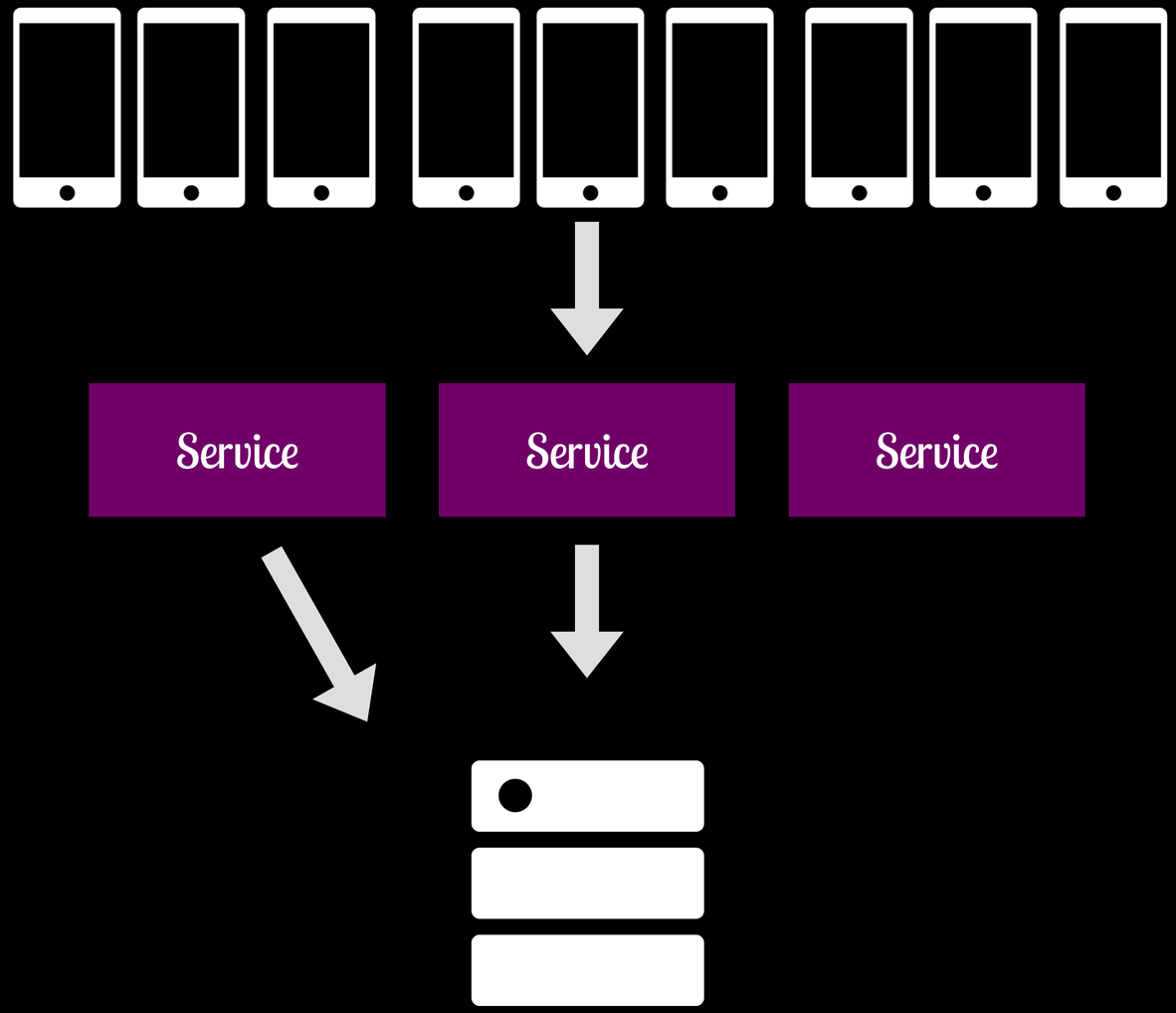
Stateless Services



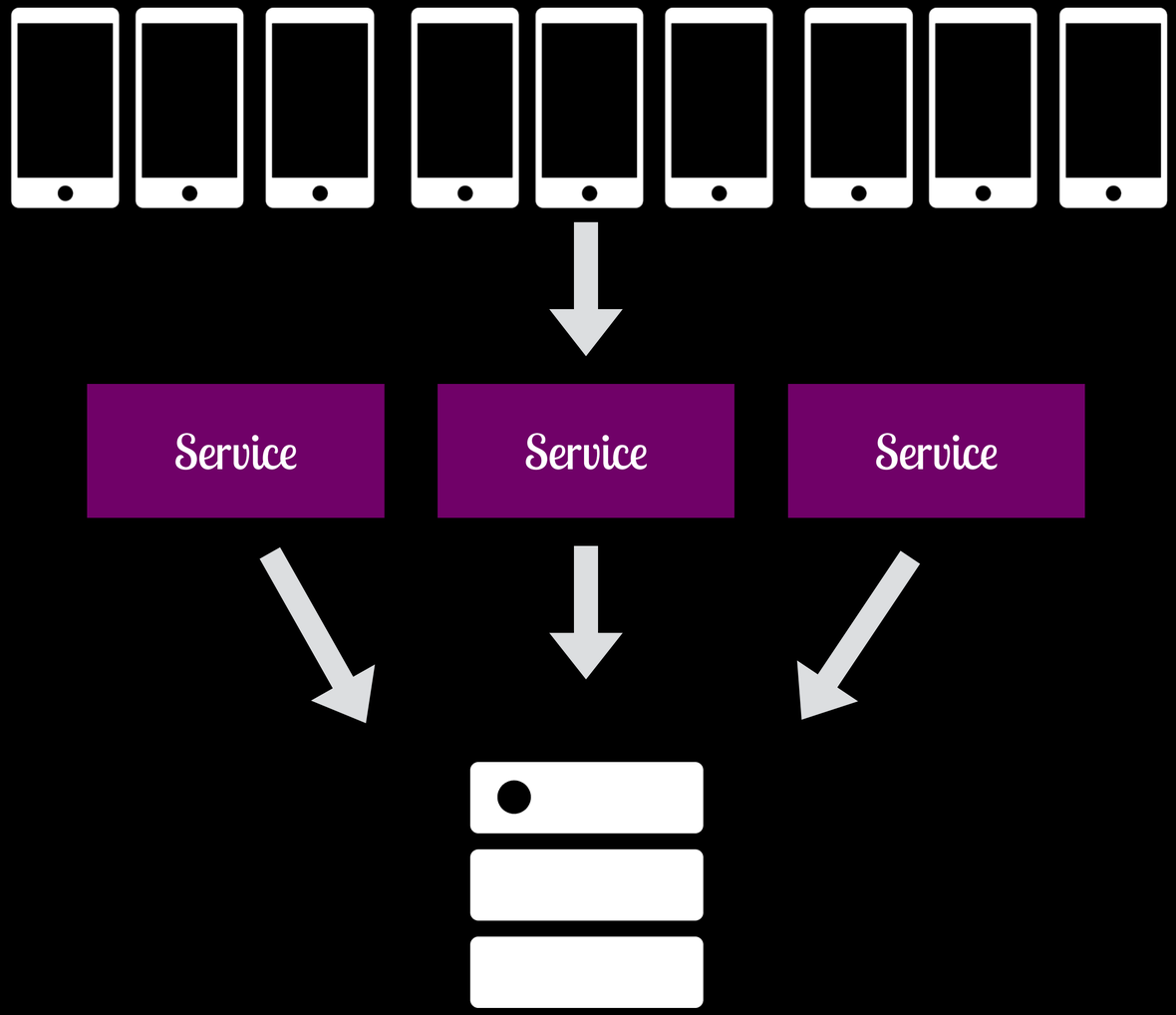
Stateless Services



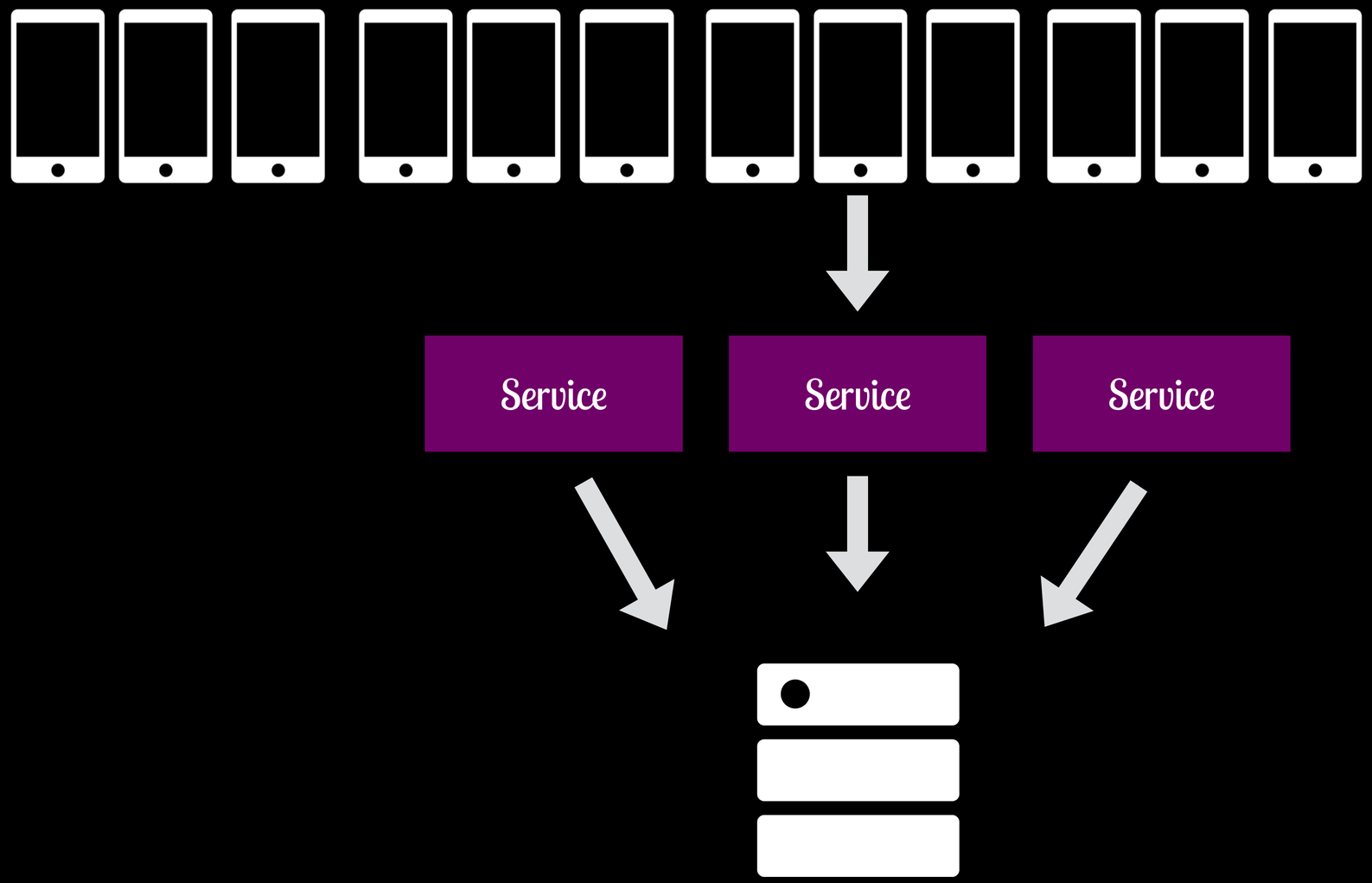
Stateless Services



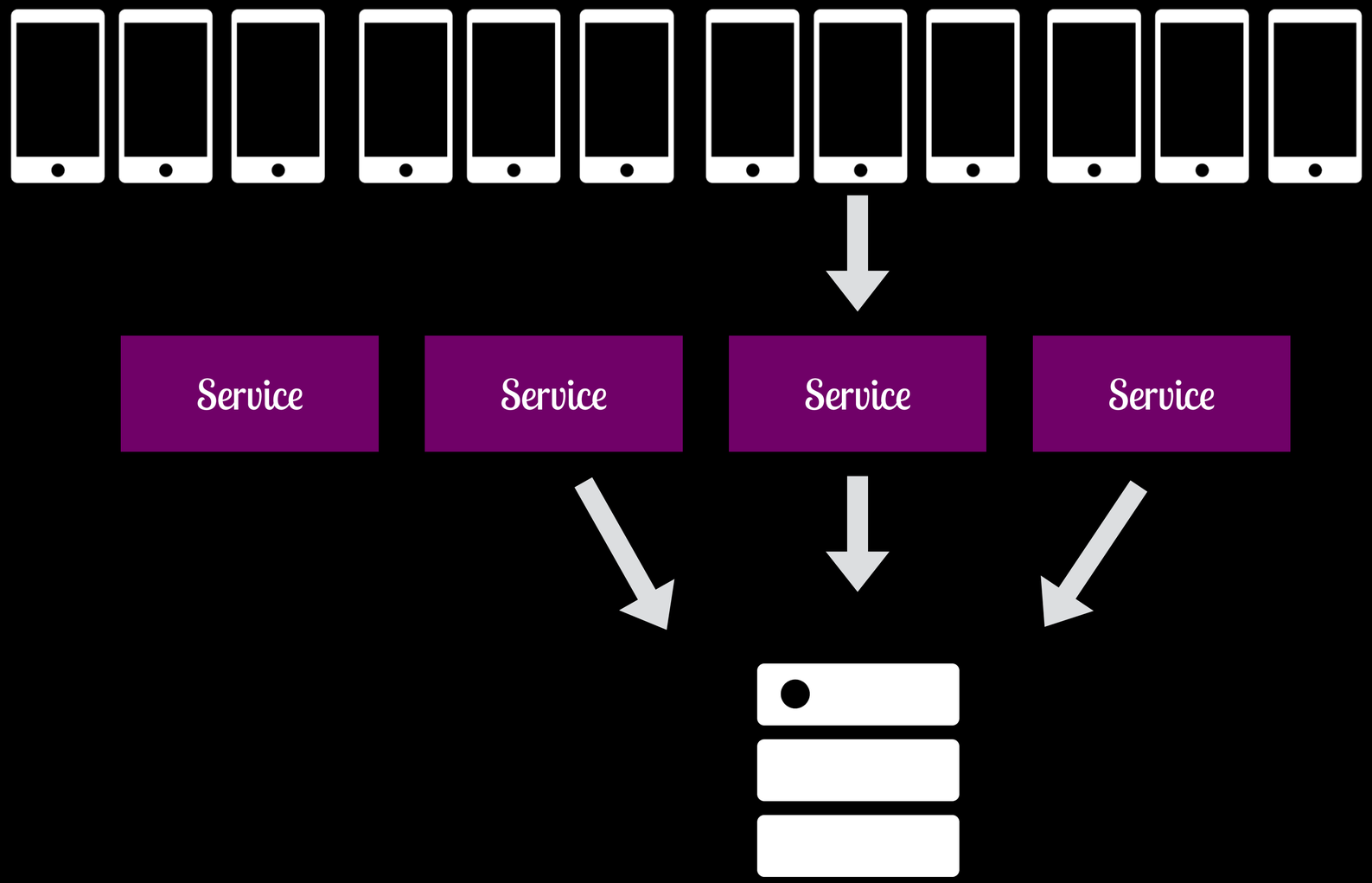
Stateless Services



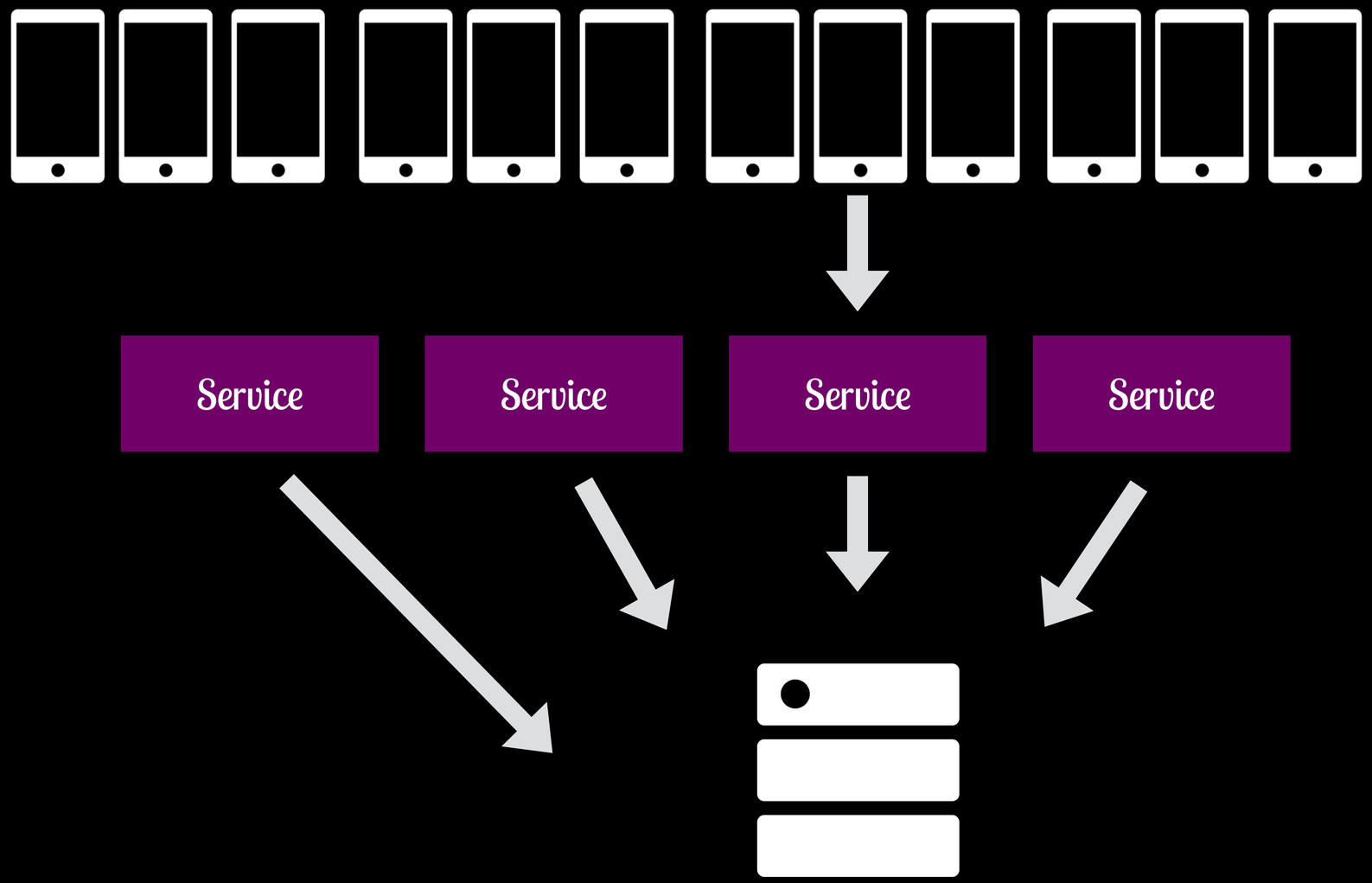
Stateless Services



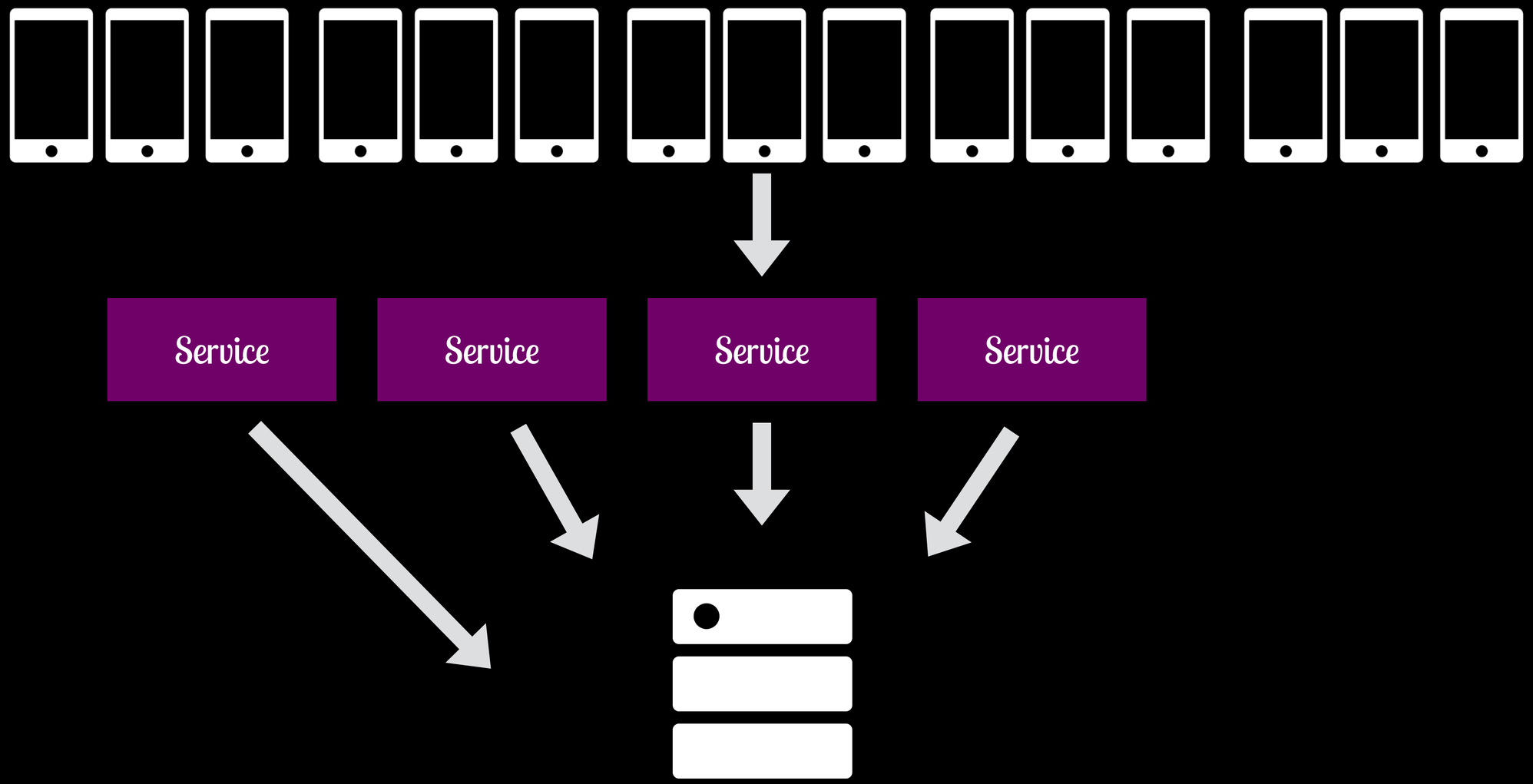
Stateless Services



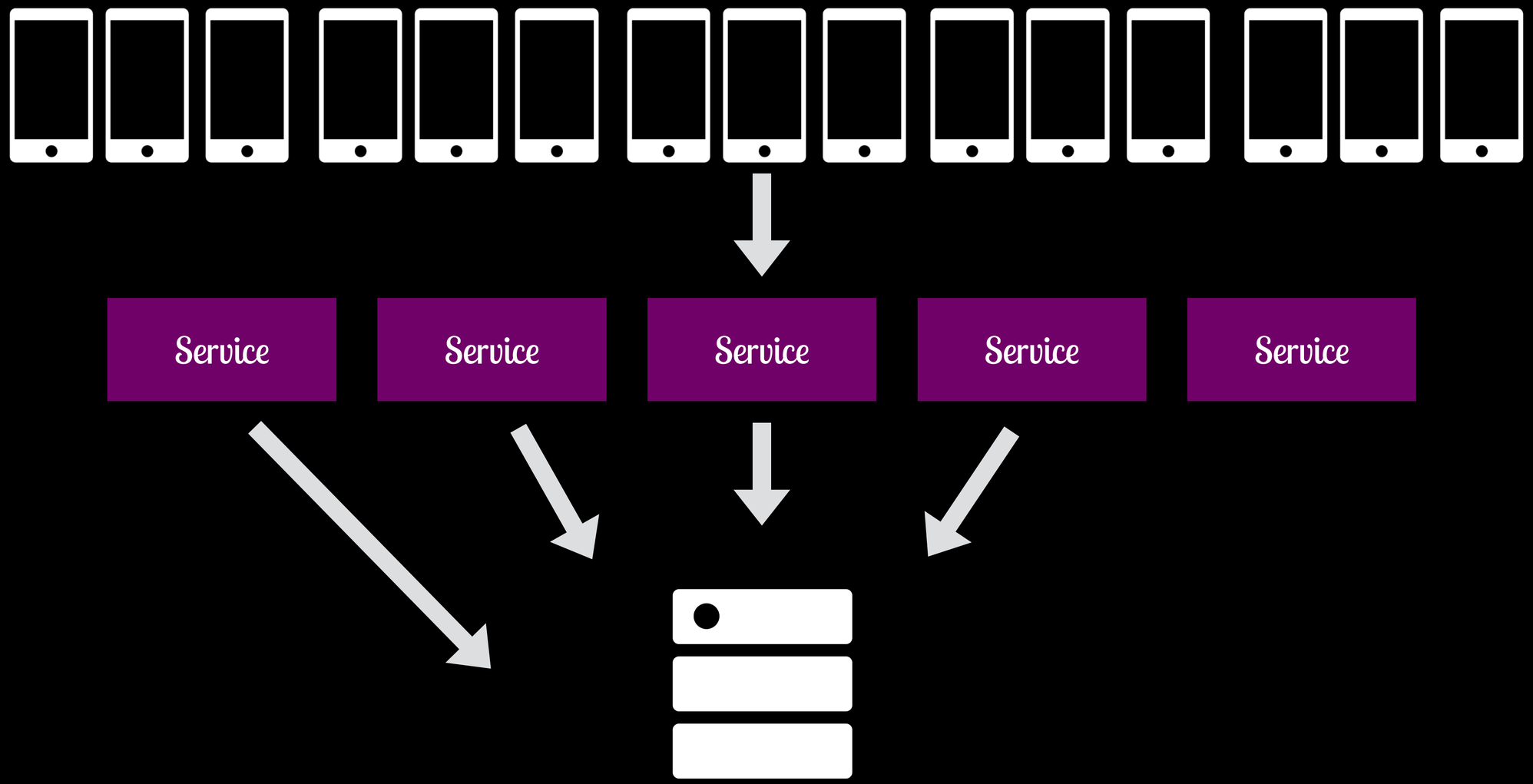
Stateless Services



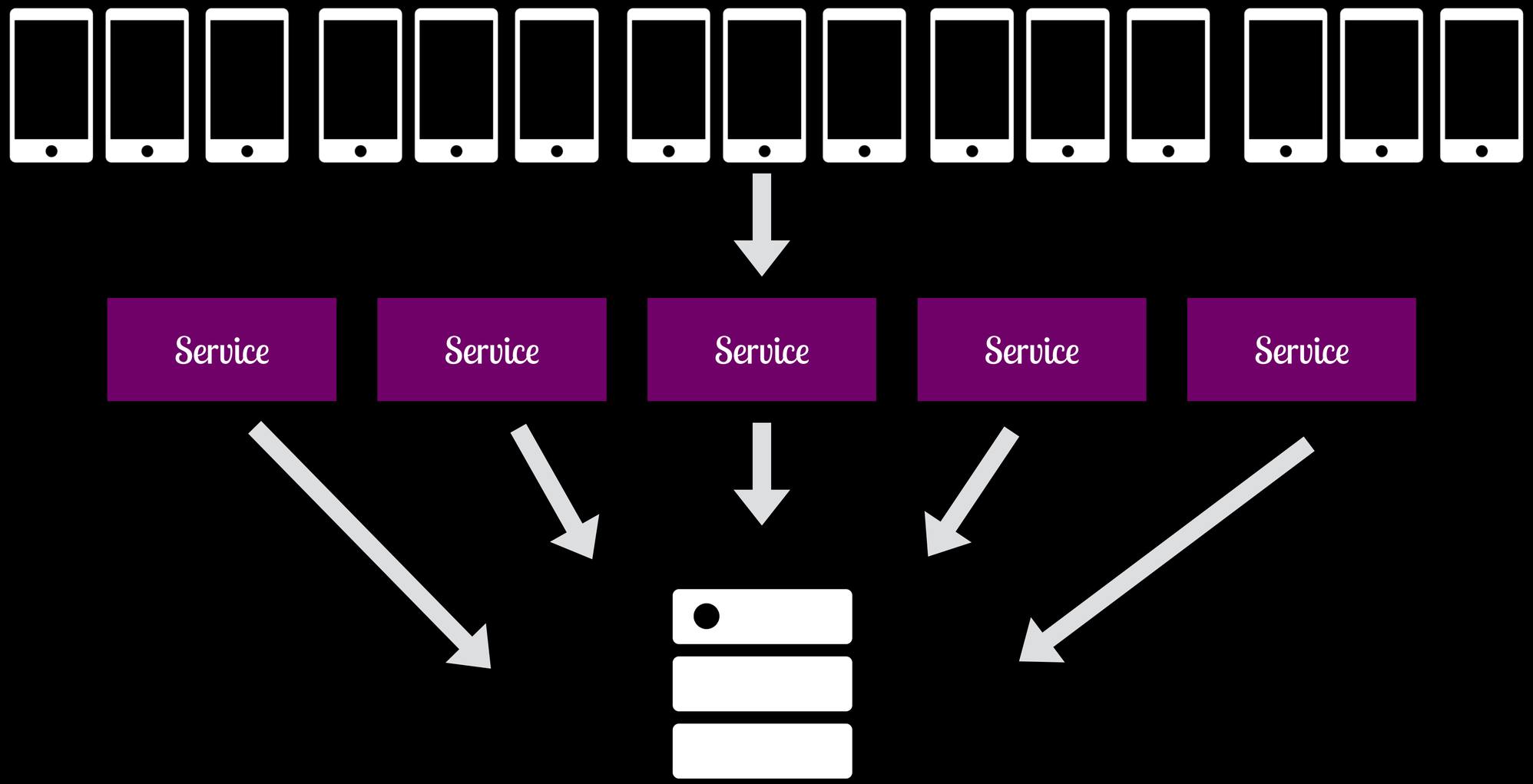
Stateless Services



Stateless Services



Stateless Services

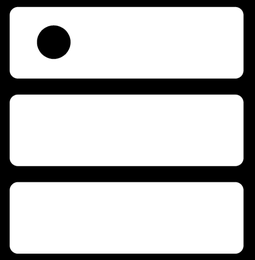


Data Shipping Paradigm

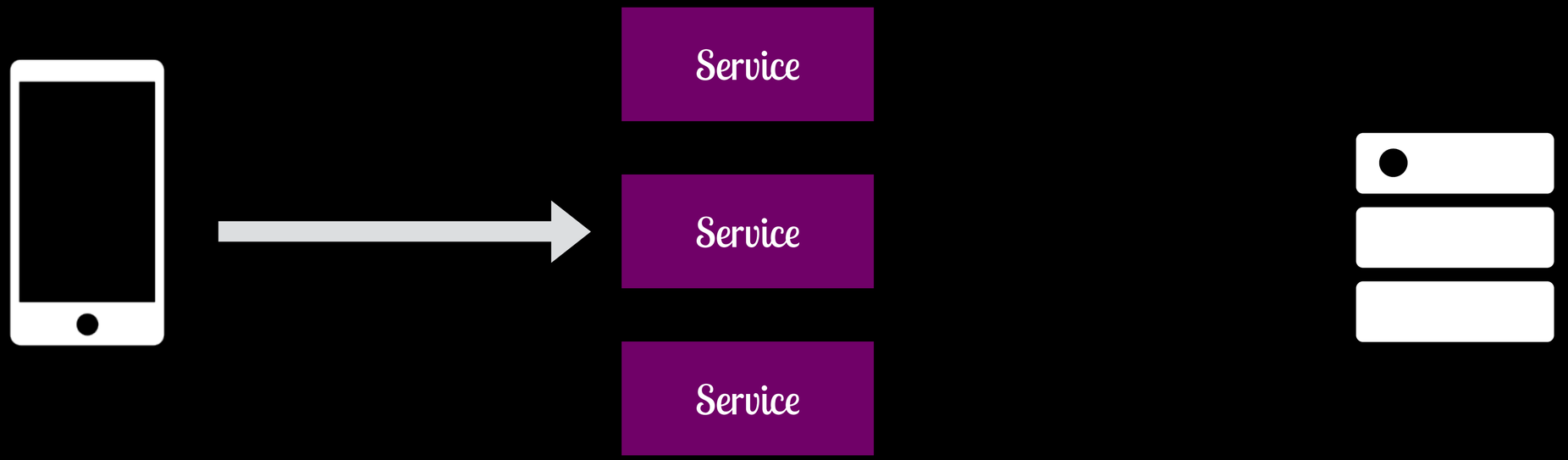
Service

Service

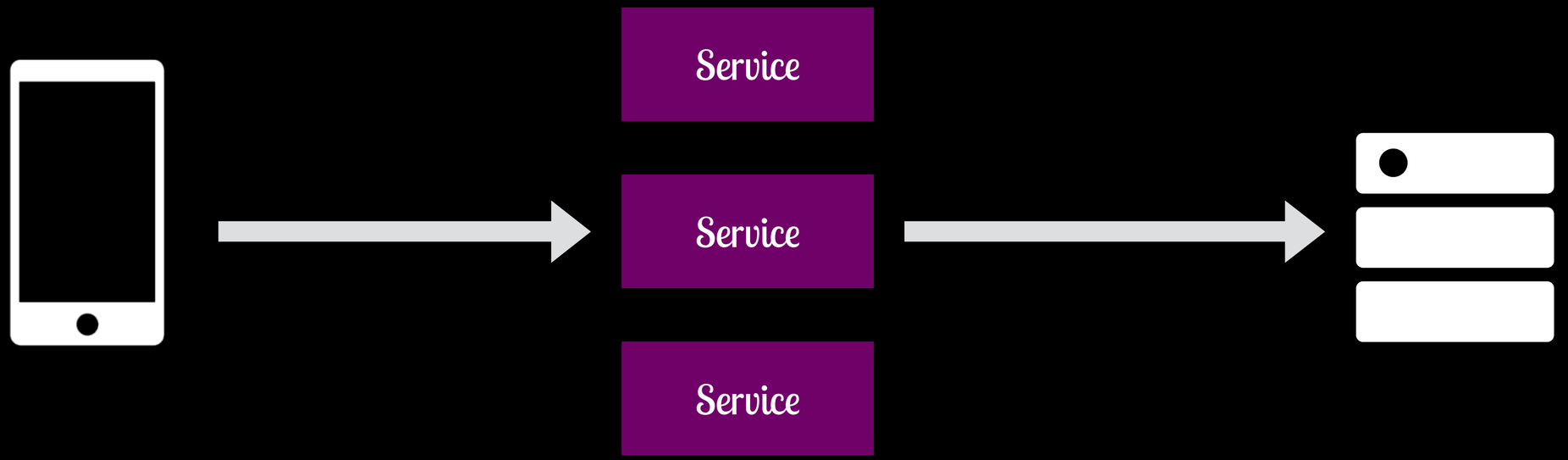
Service



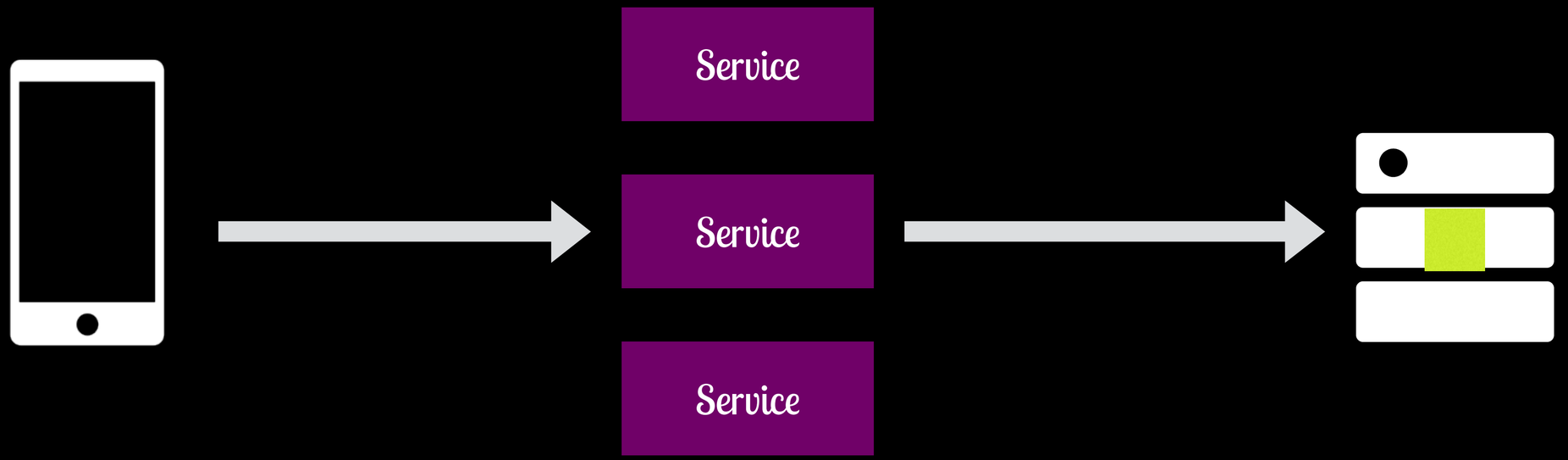
Data Shipping Paradigm



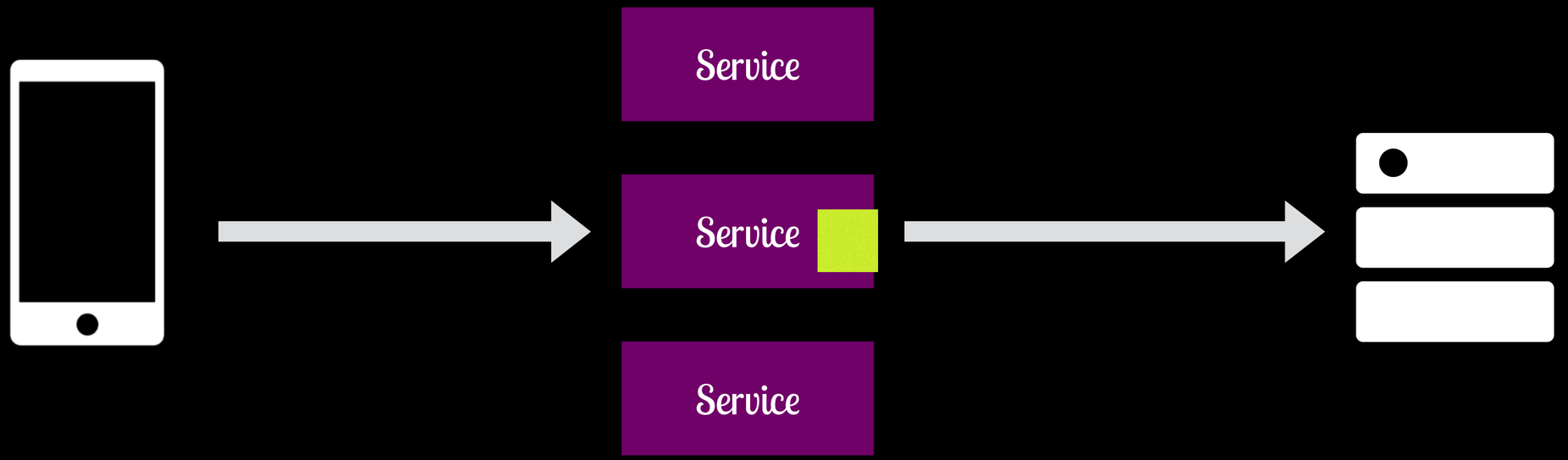
Data Shipping Paradigm



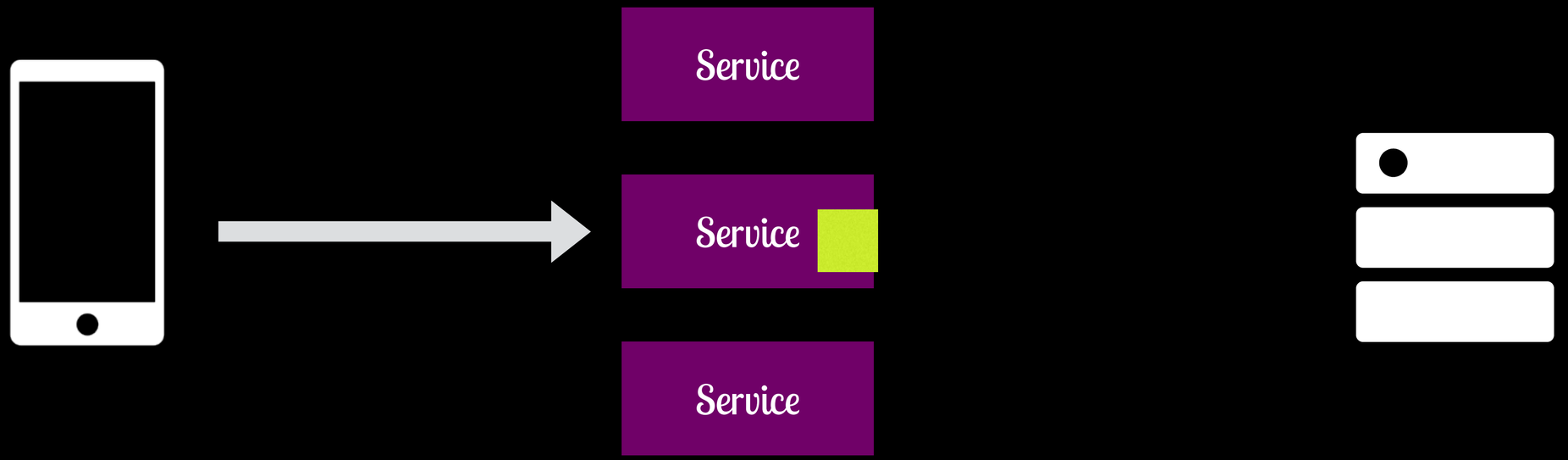
Data Shipping Paradigm



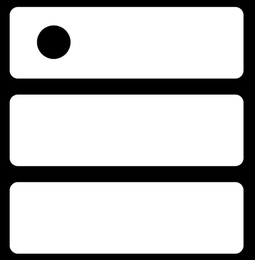
Data Shipping Paradigm



Data Shipping Paradigm



Data Shipping Paradigm

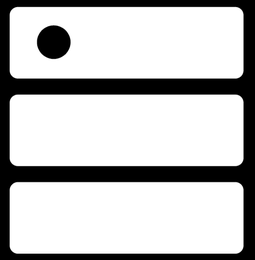


Data Shipping Paradigm

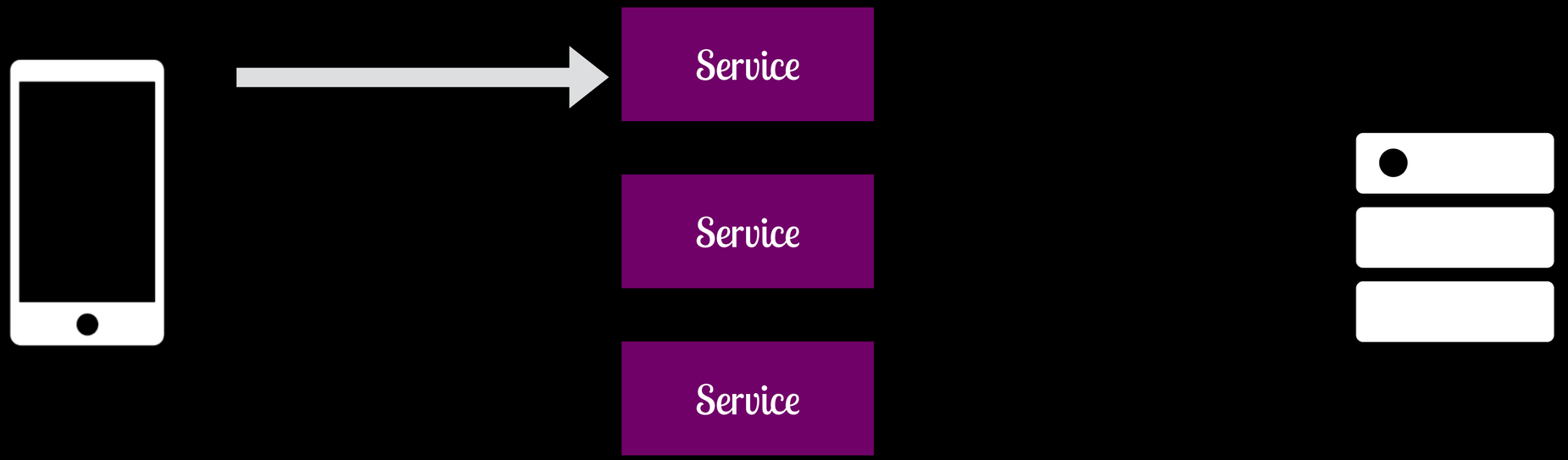
Service

Service

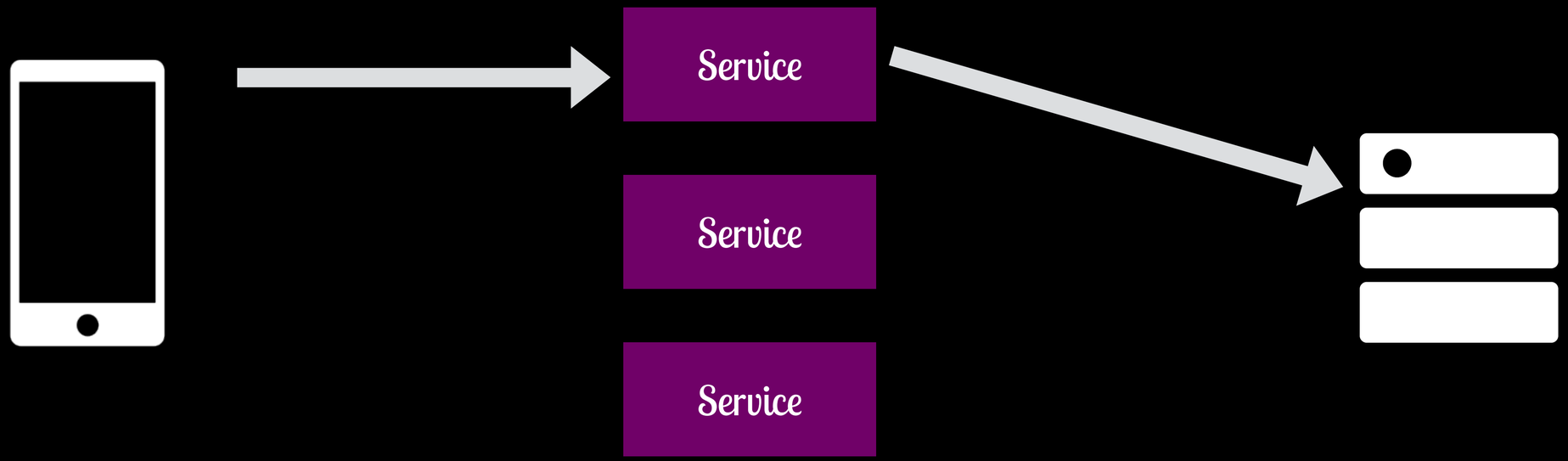
Service



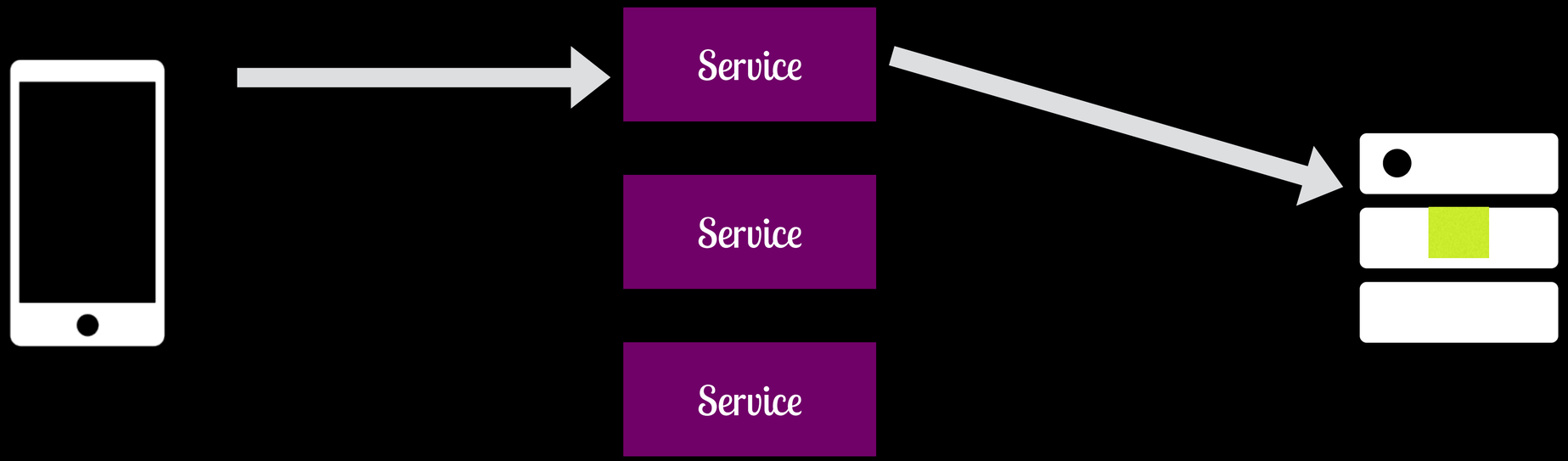
Data Shipping Paradigm



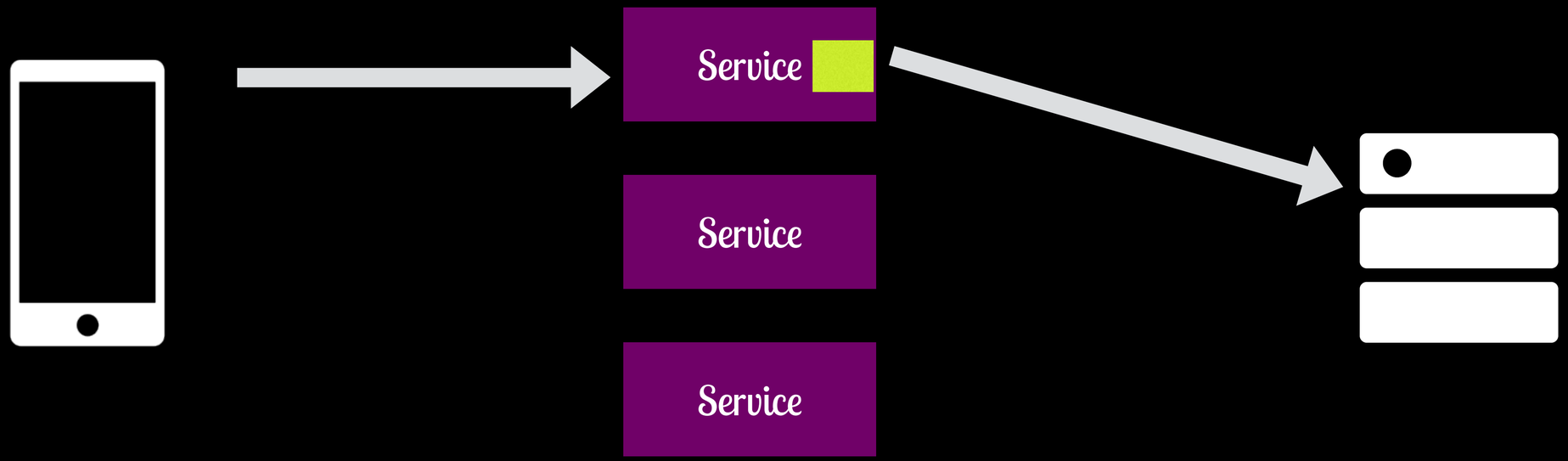
Data Shipping Paradigm



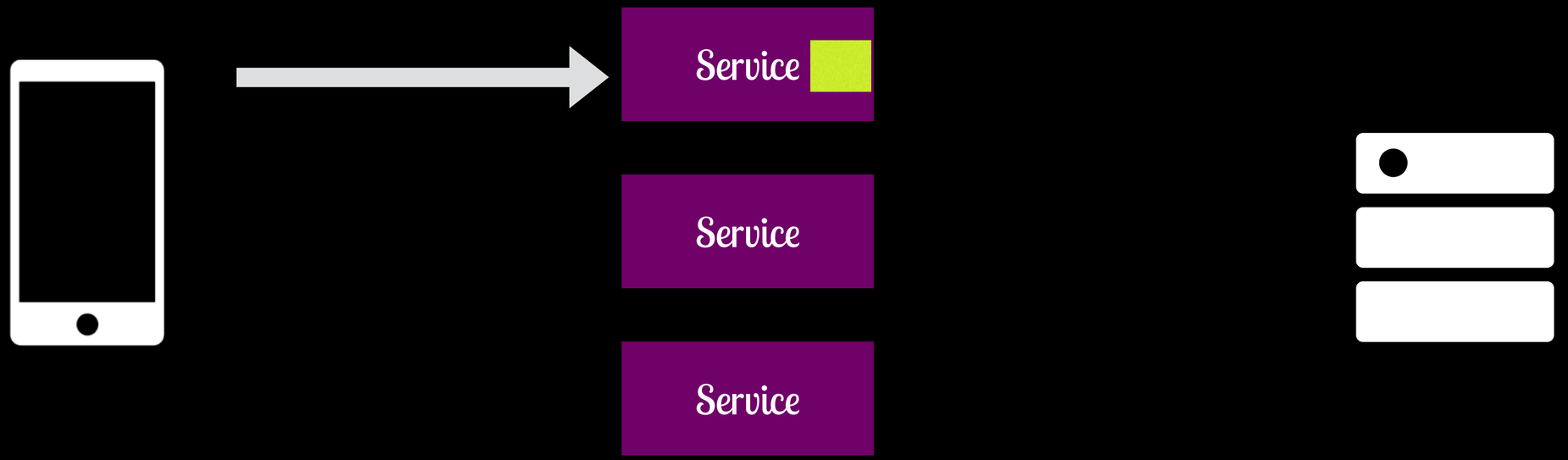
Data Shipping Paradigm



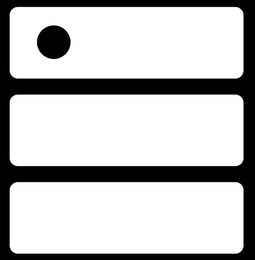
Data Shipping Paradigm



Data Shipping Paradigm



Data Shipping Paradigm

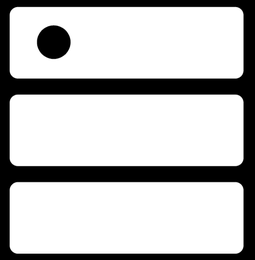


Data Shipping Paradigm

Service

Service

Service



Overview

Benefits

Building

Real World

Caution

Data Locality

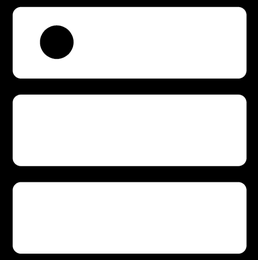
For Low Latency & Data Intensive Services

Function Shipping Paradigm

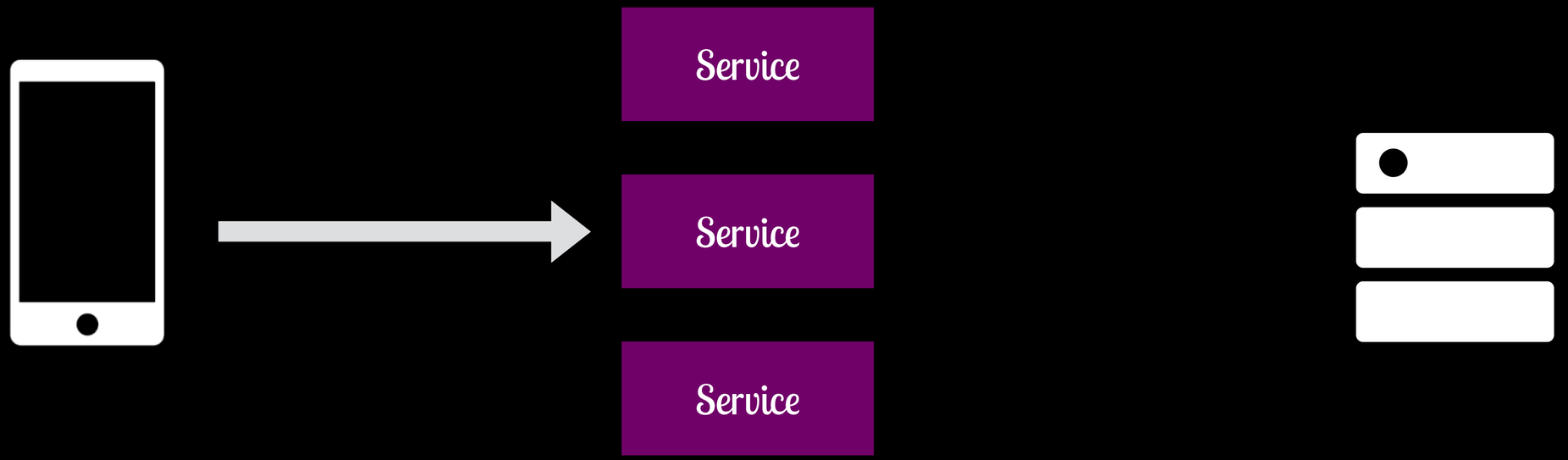
Service

Service

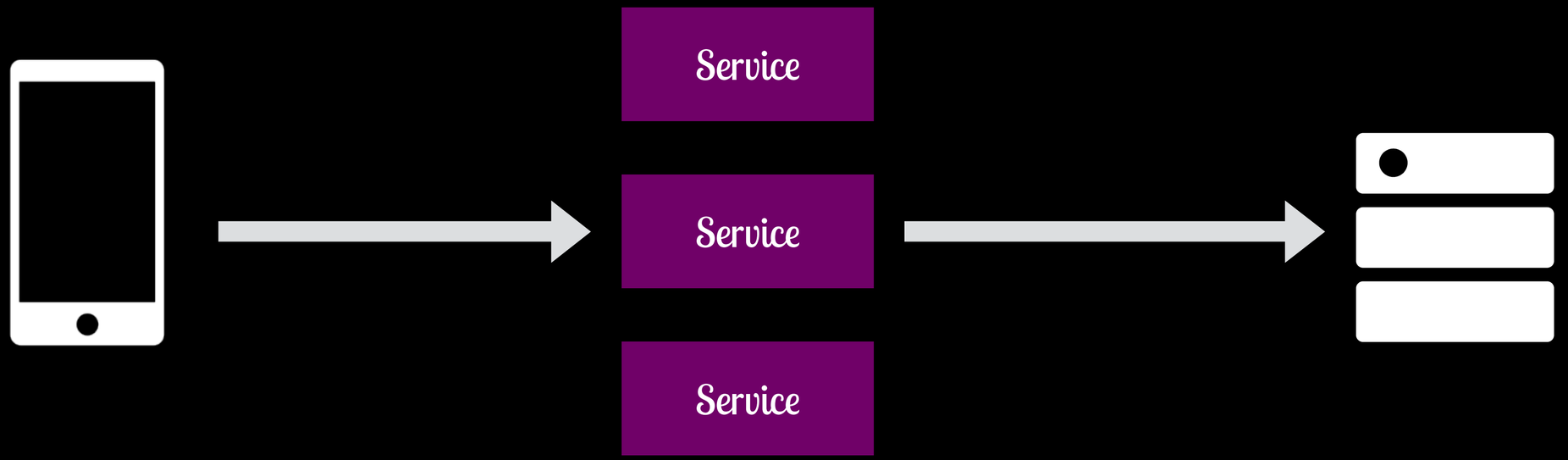
Service



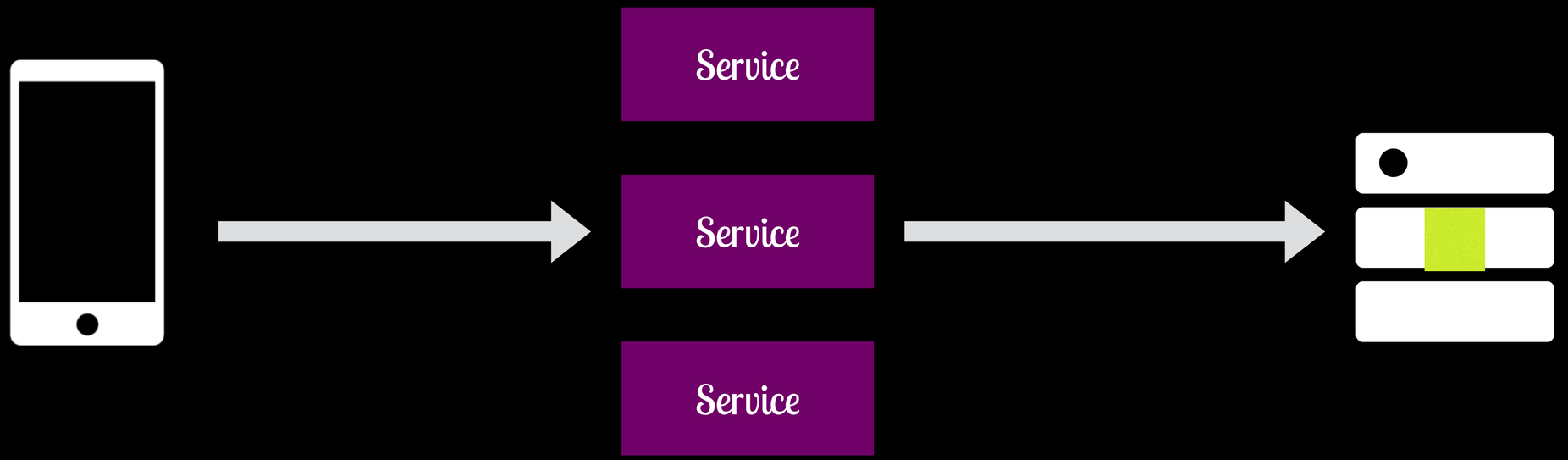
Function Shipping Paradigm



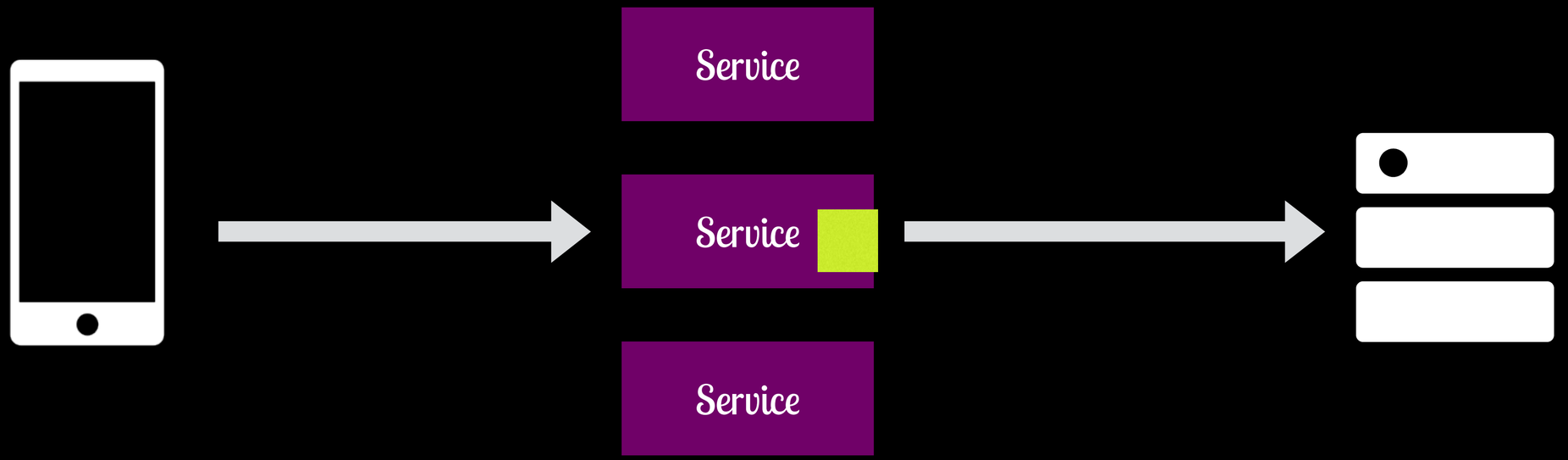
Function Shipping Paradigm



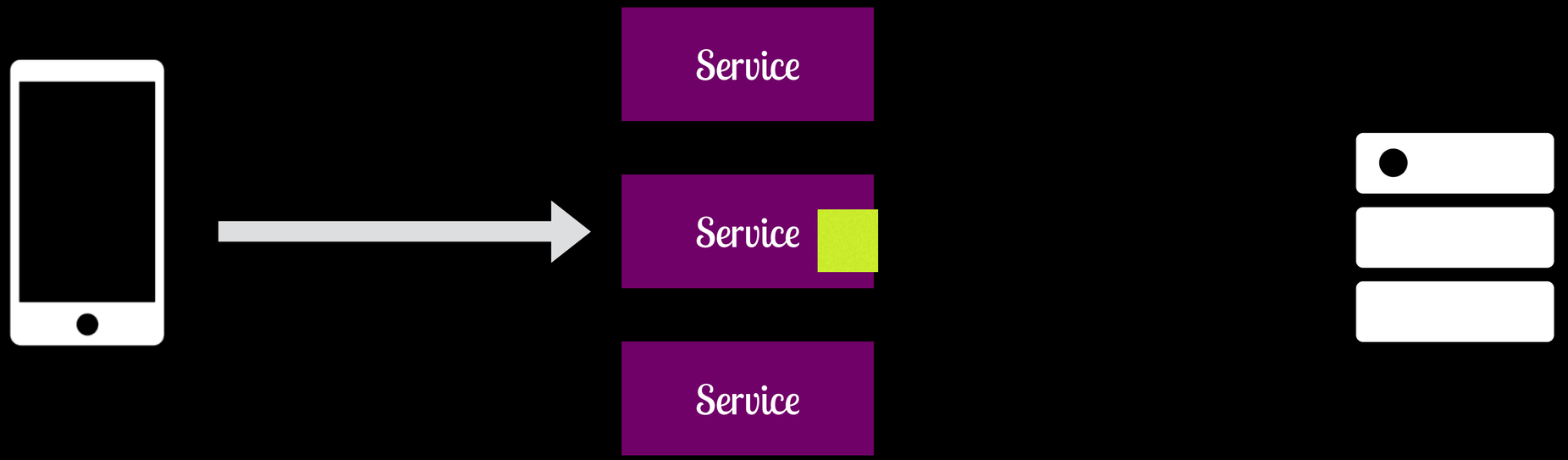
Function Shipping Paradigm



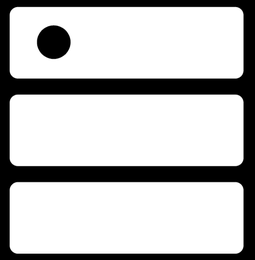
Function Shipping Paradigm



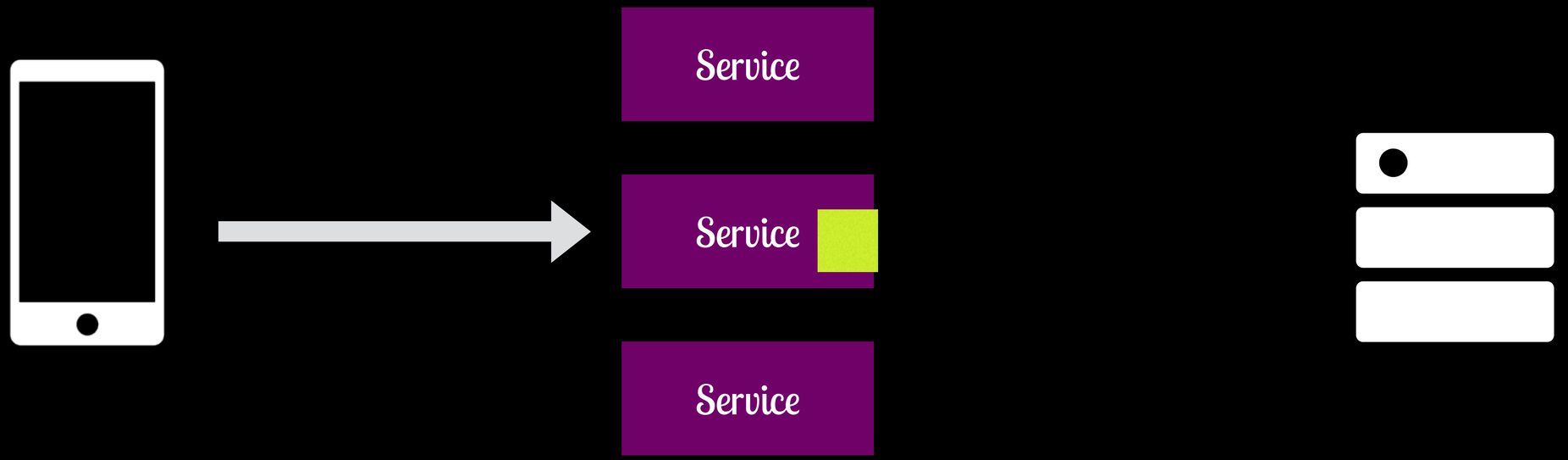
Function Shipping Paradigm



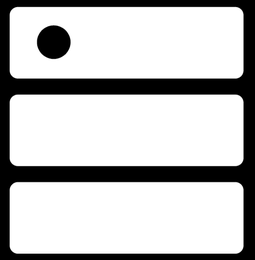
Function Shipping Paradigm



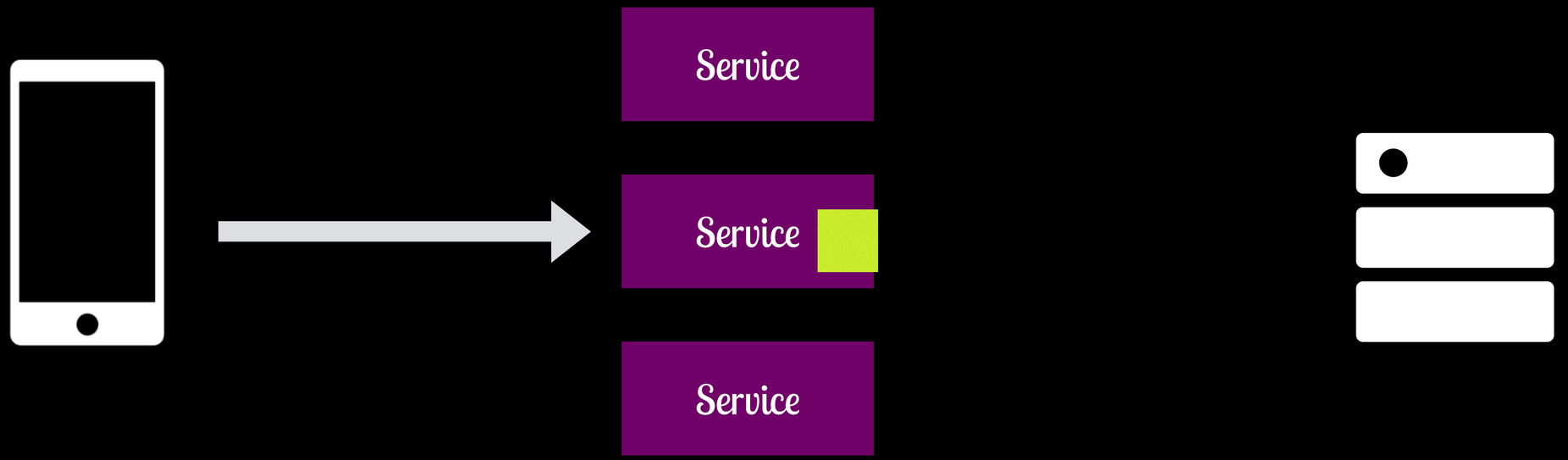
Function Shipping Paradigm



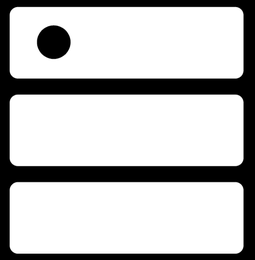
Function Shipping Paradigm



Function Shipping Paradigm



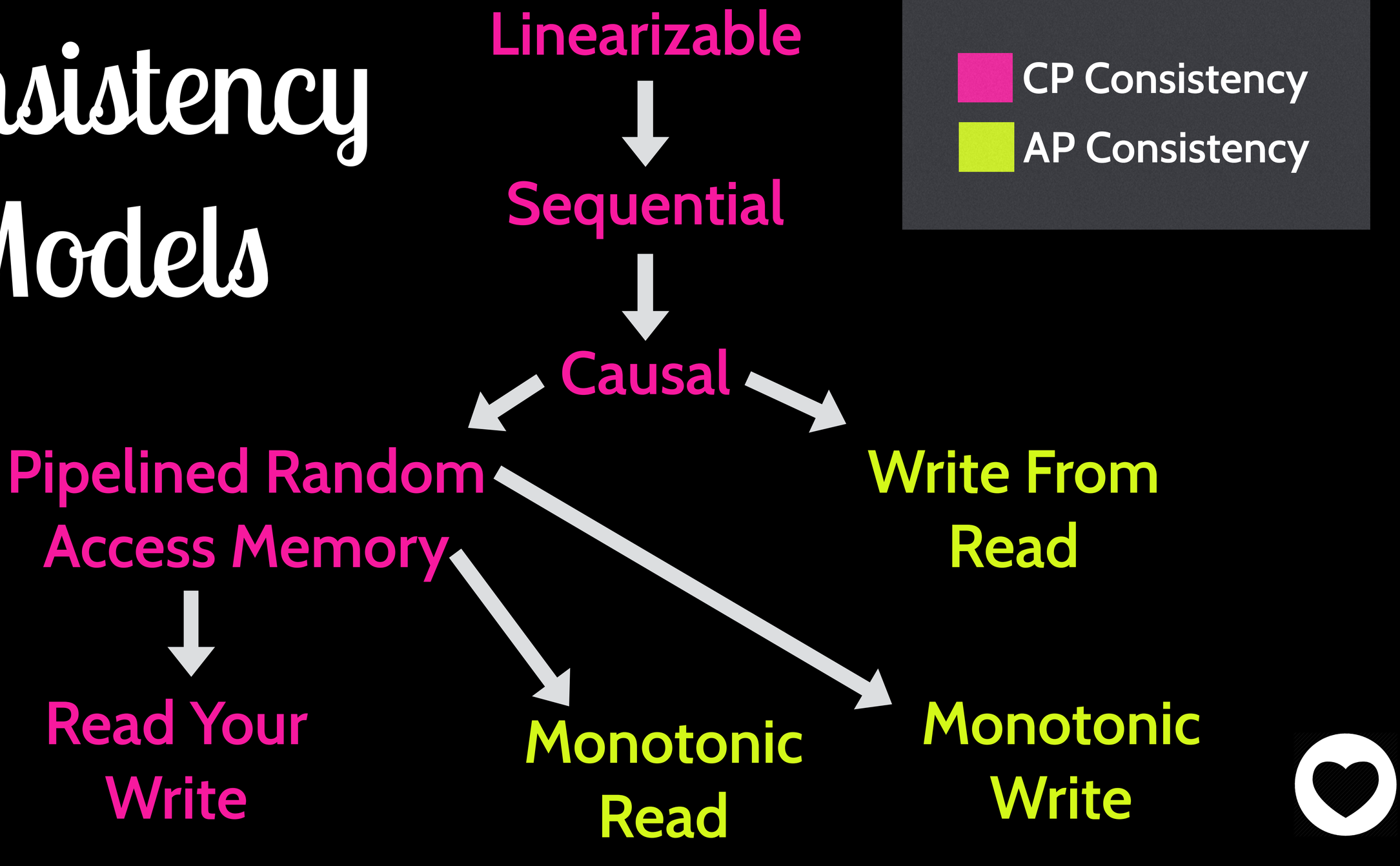
Function Shipping Paradigm



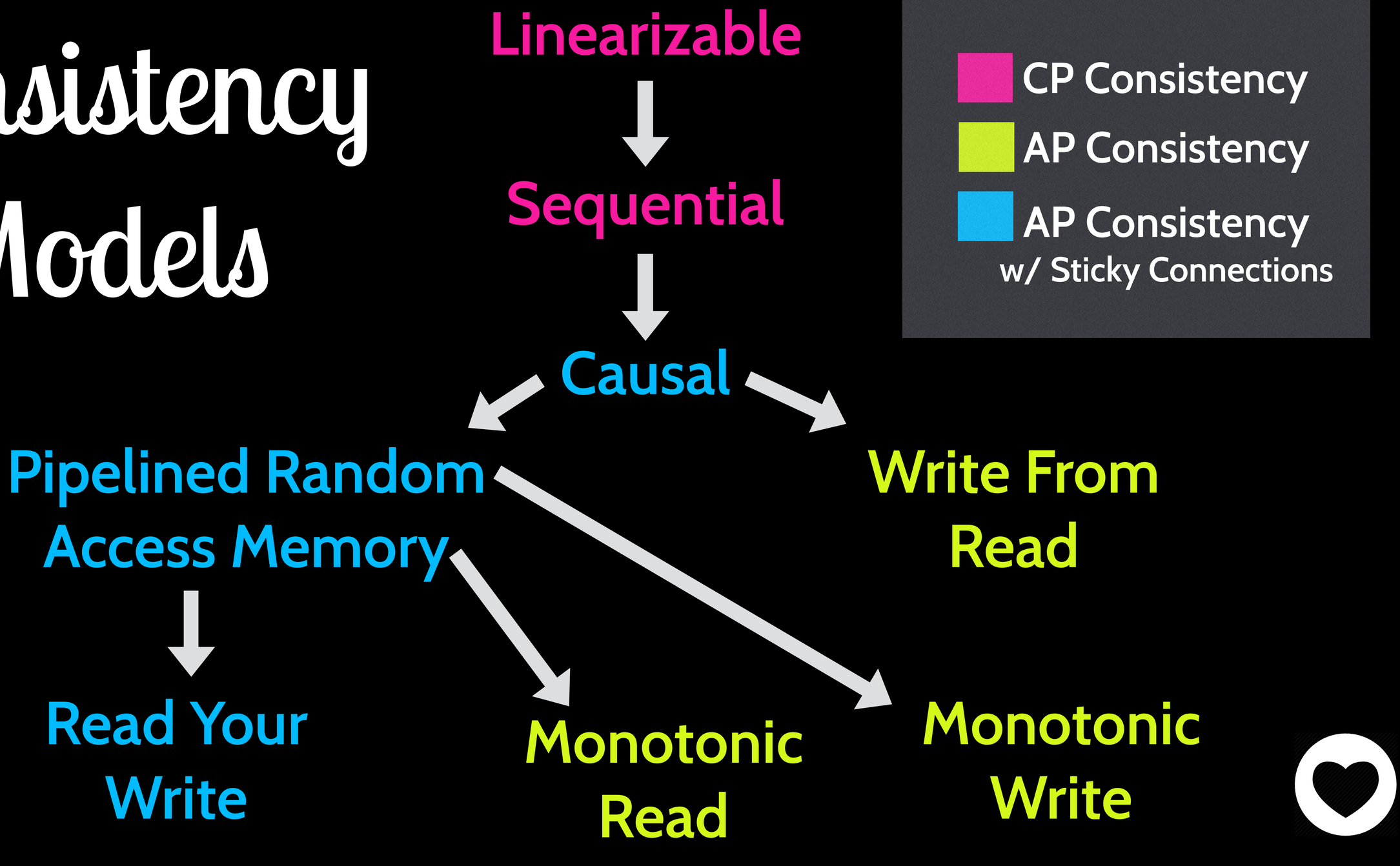
Sticky Connections & Consistency

Additional Available Consistency Models

Consistency Models



Consistency Models



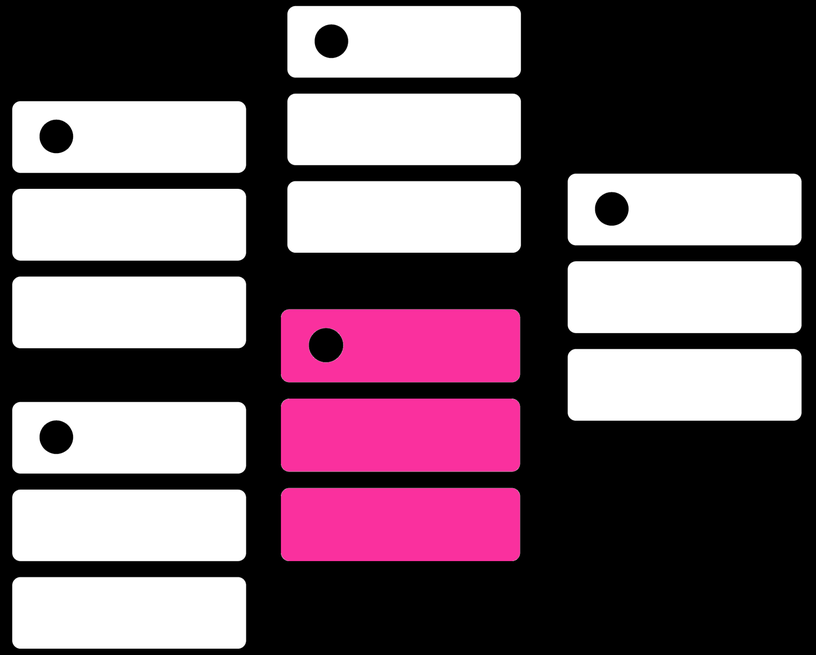
“Whether or not read-your-write, session and monotonic consistency can be achieved depends in general on the "stickiness" of clients to the server that executes the distributed protocol for them... Using sessions, which are sticky, makes this explicit and provides an exposure level that clients can reason about.”

- Werner Vogel 2007

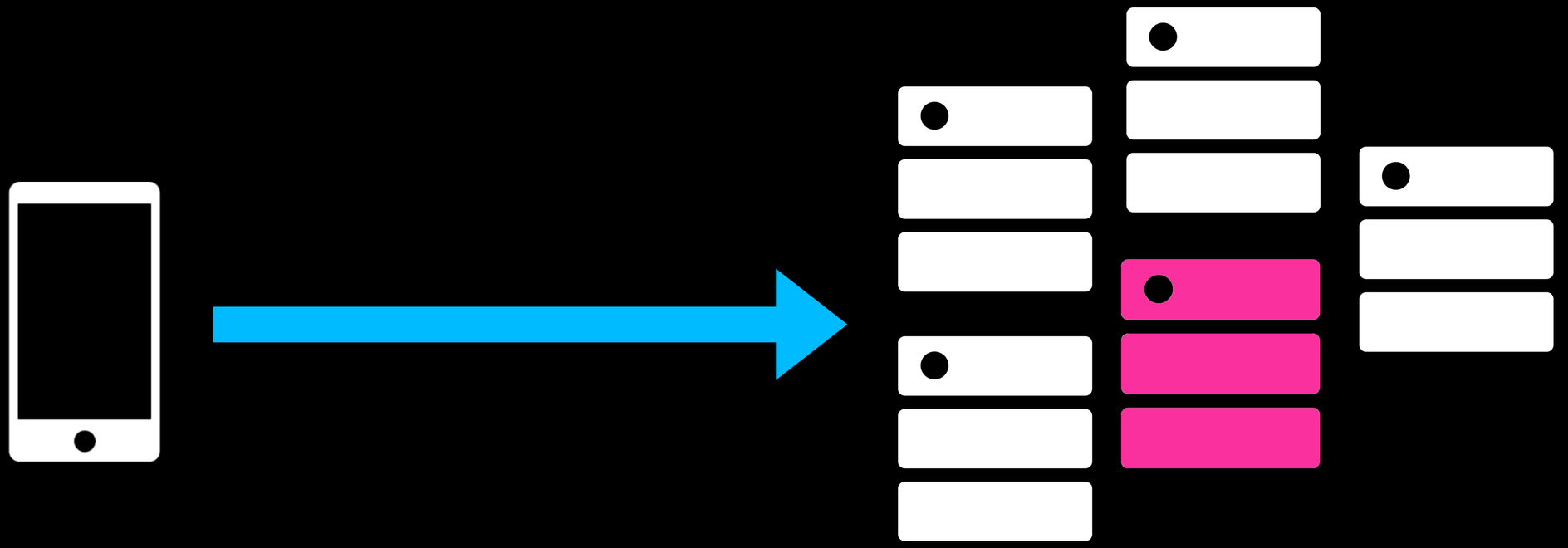
Building Sticky Connections

For Low Latency & Data Intensive Services

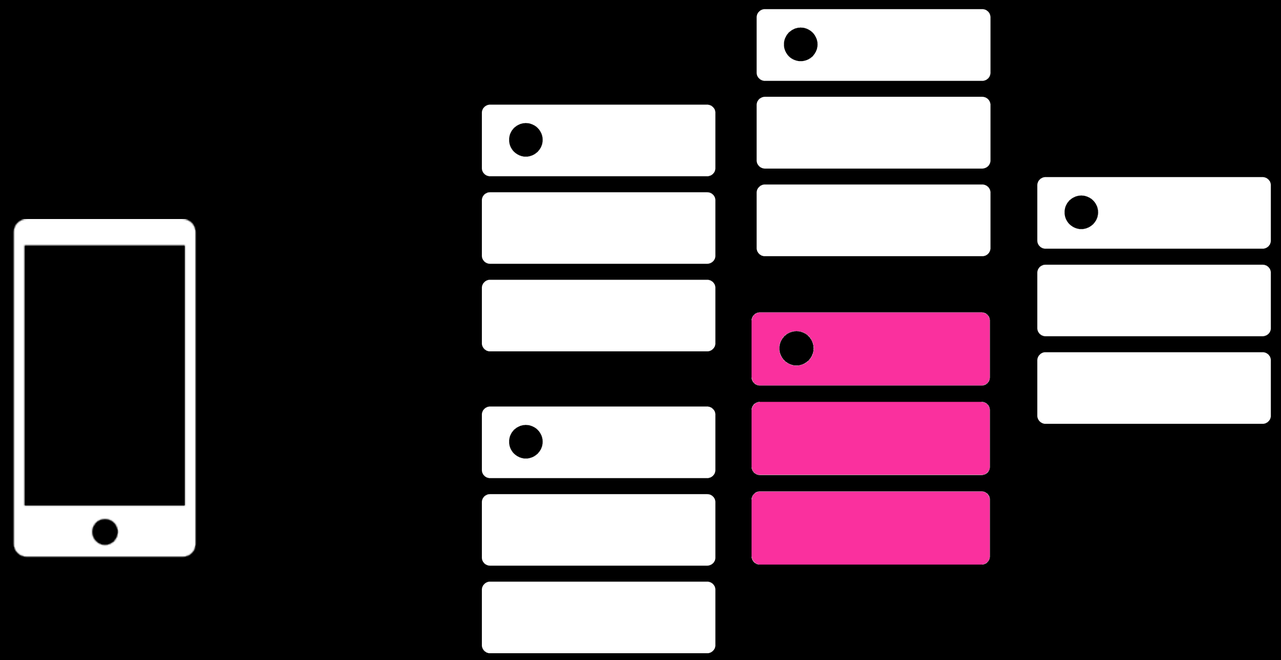
Building Sticky Connections



Building Sticky Connections



Persistent Connections

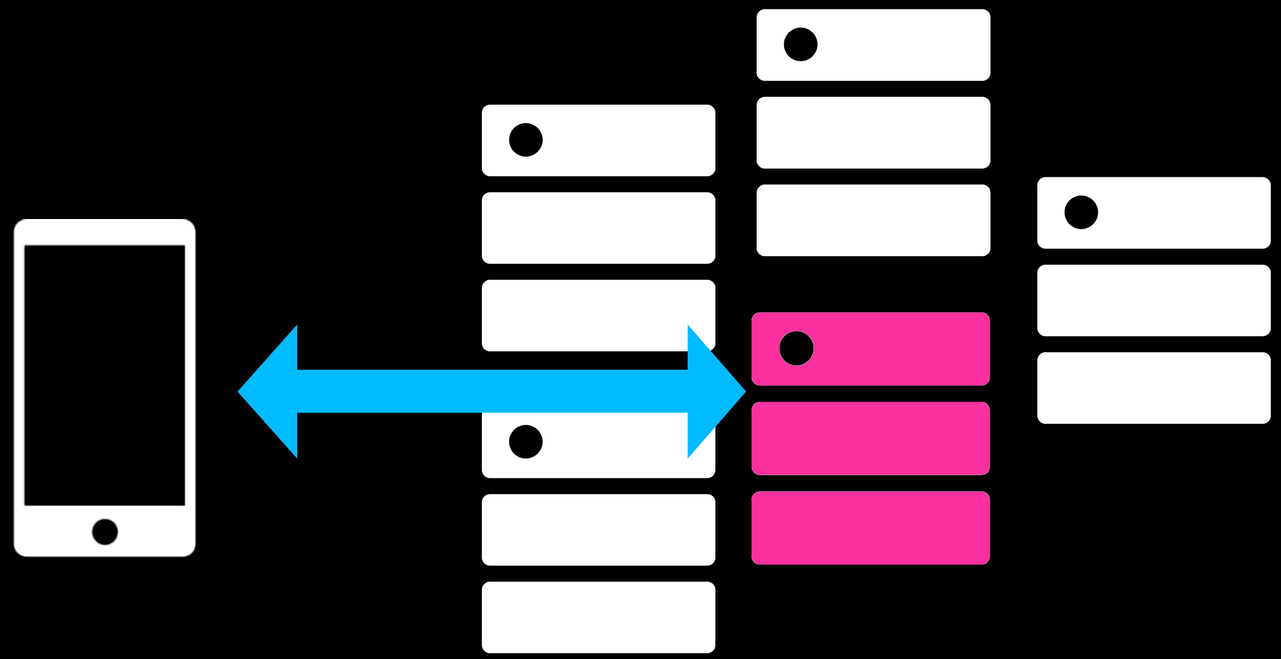


Problems

Load Balancing Problems

No Stickiness Once Connection Breaks

Persistent Connections

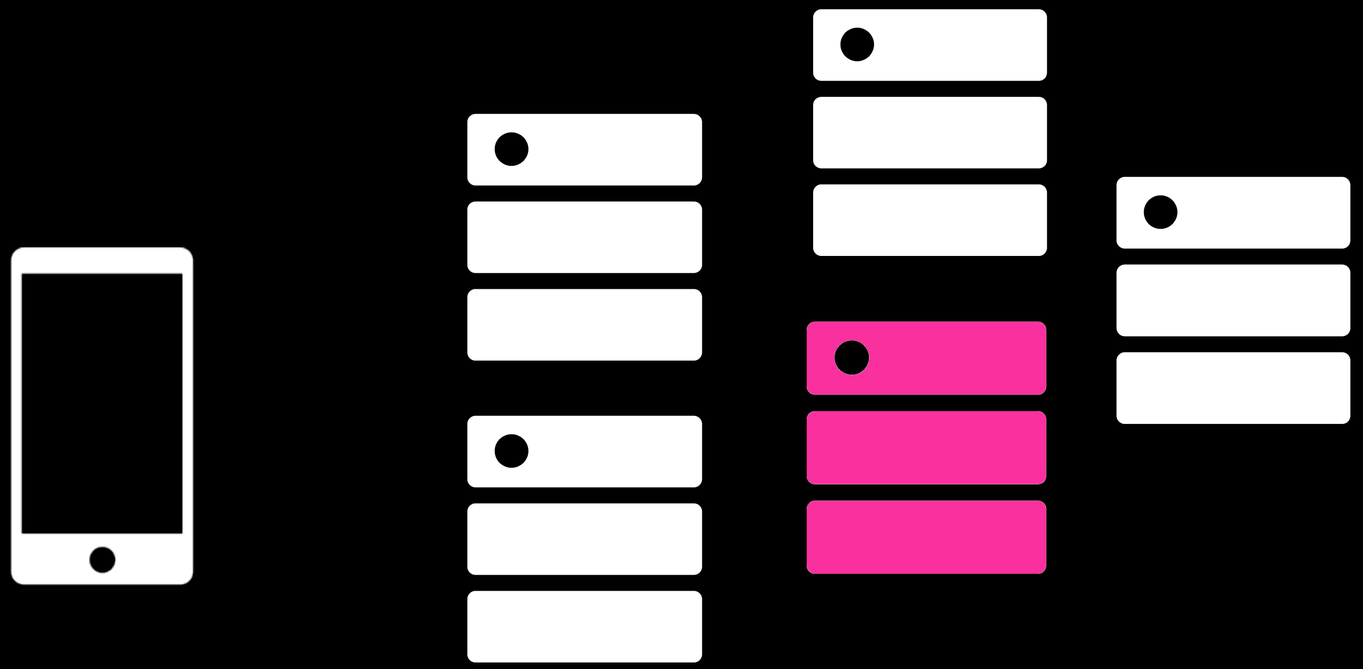


Problems

Load Balancing Problems

No Stickiness Once Connection Breaks

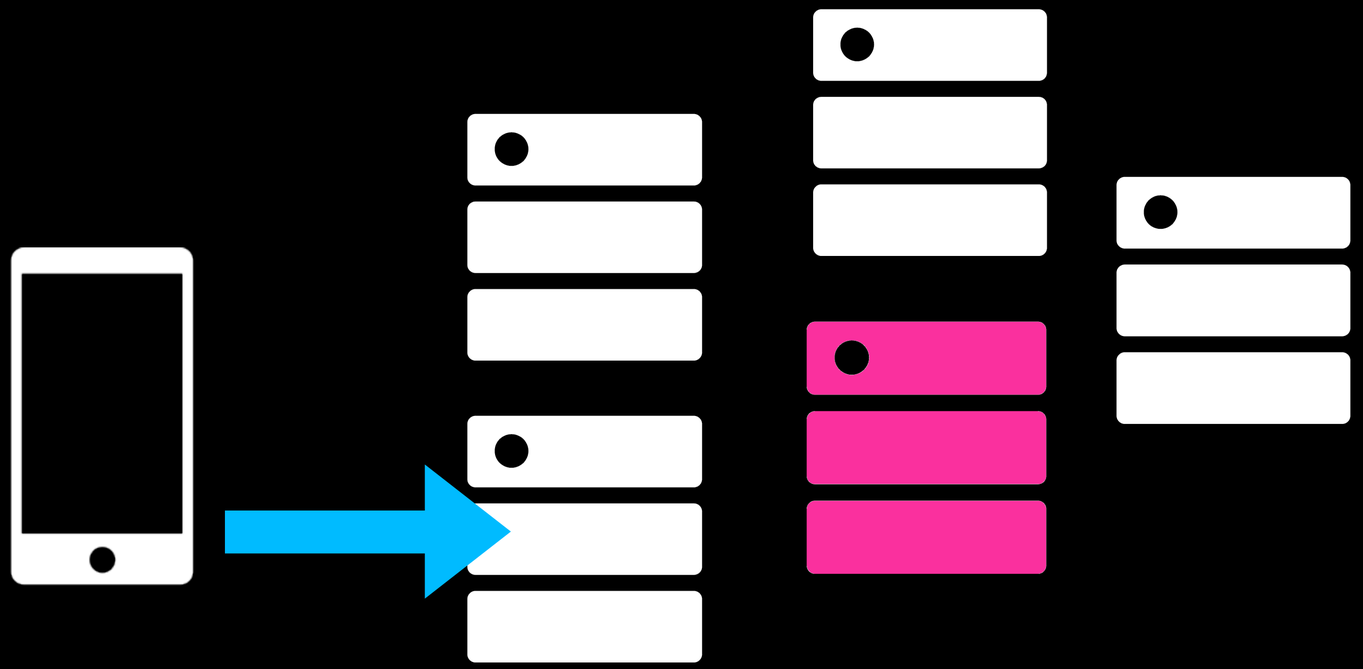
Routing Logic



Problems to Solve

- Cluster Membership
- Work Distribution

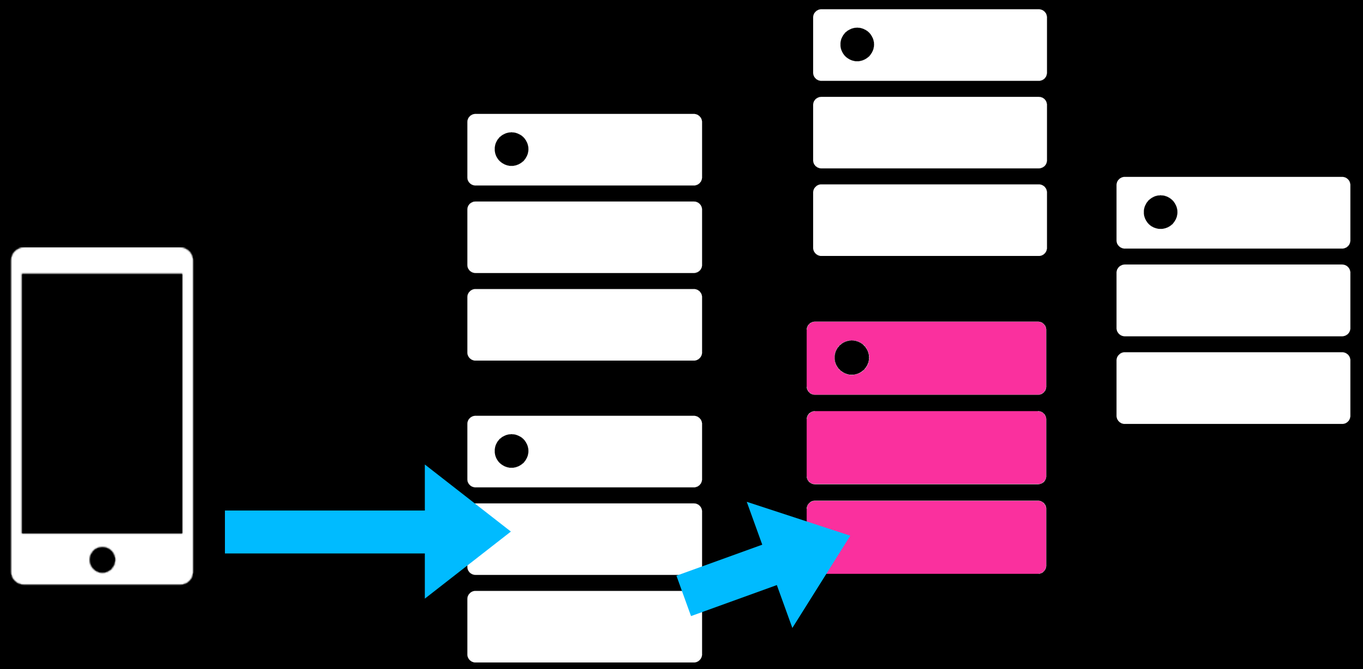
Routing Logic



Problems to Solve

- Cluster Membership
- Work Distribution

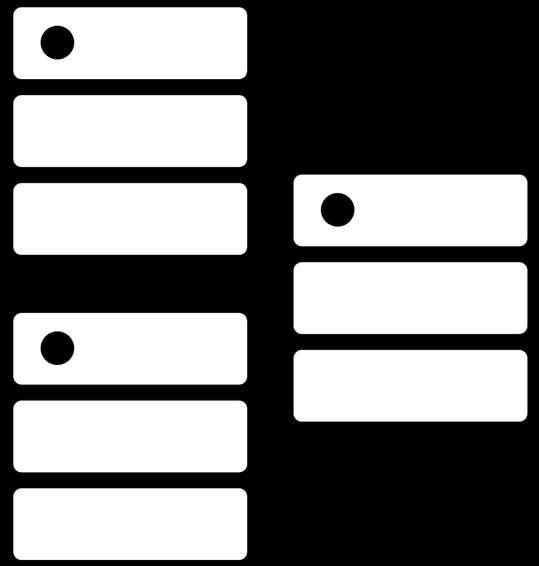
Routing Logic



Problems to Solve

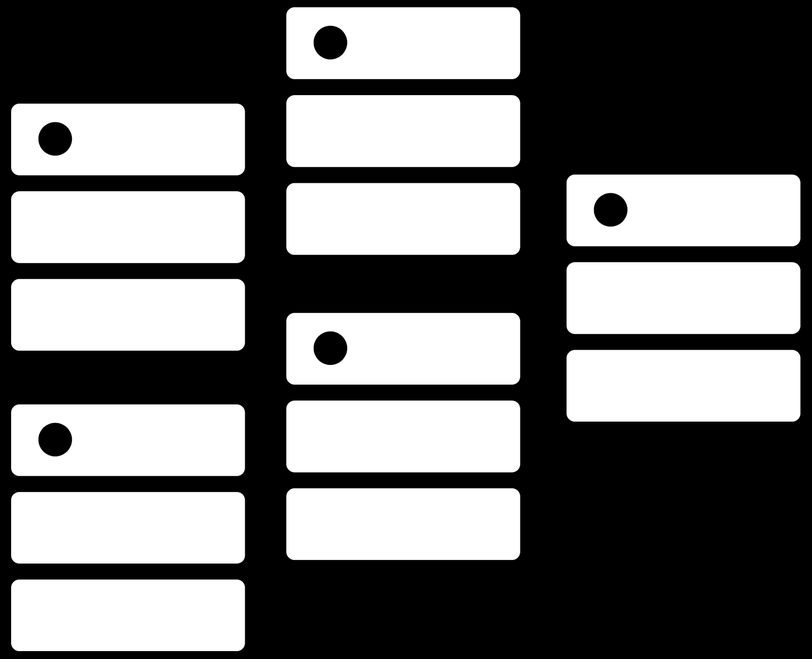
- Cluster Membership
- Work Distribution

Static Cluster Membership

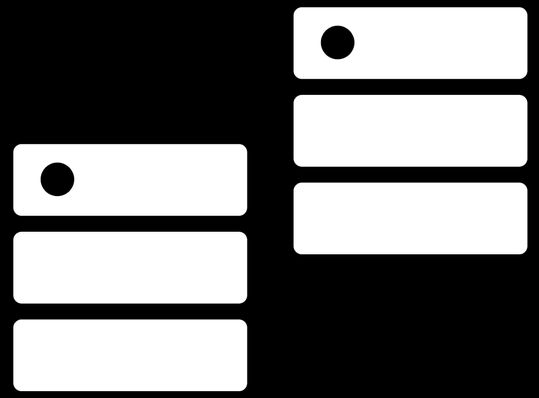


Static Cluster Membership

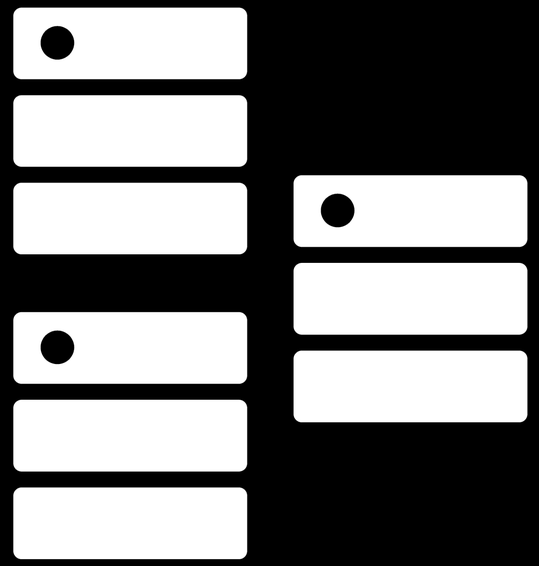
Static Cluster Membership



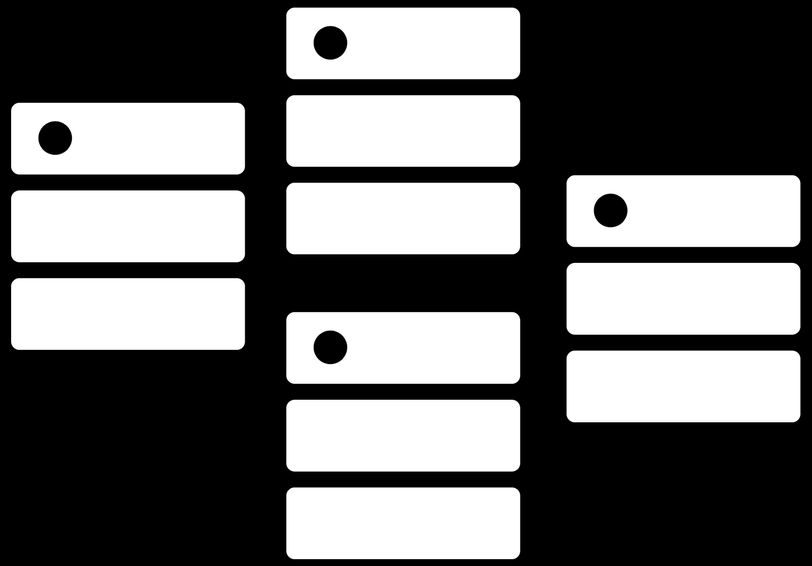
Dynamic Cluster Membership



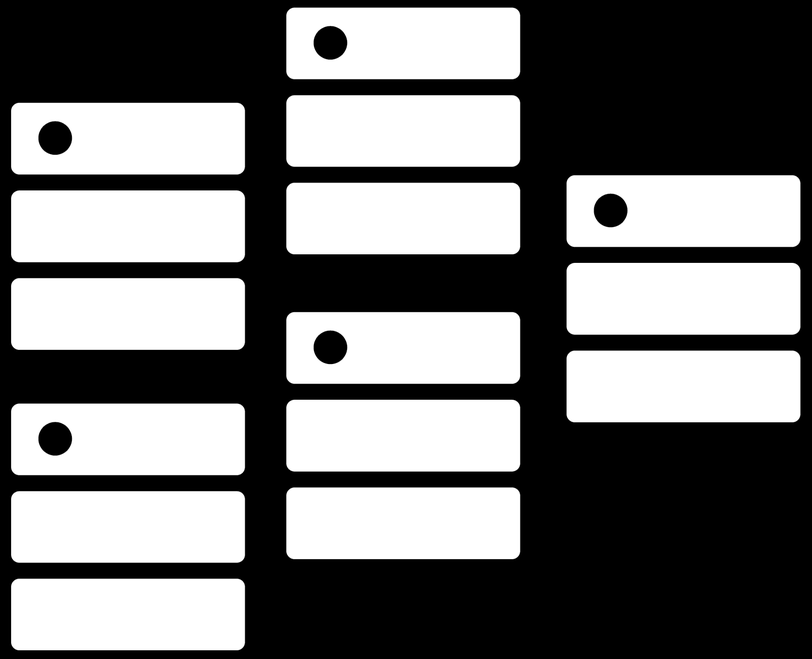
Dynamic Cluster Membership



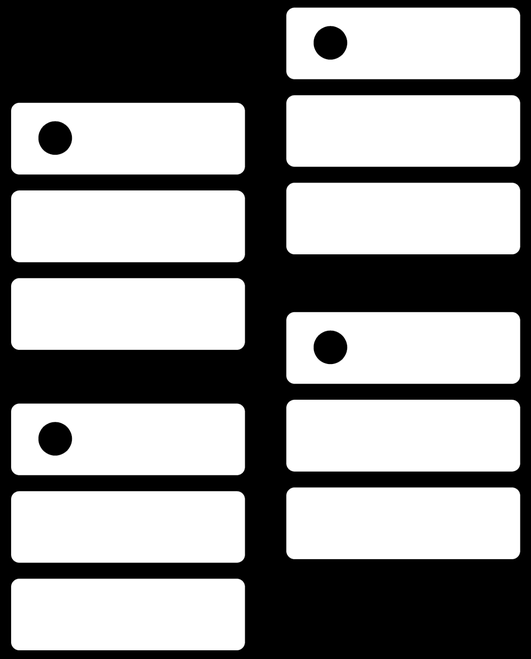
Dynamic Cluster Membership



Dynamic Cluster Membership



Dynamic Cluster Membership



Dynamic Cluster Membership

Gossip Protocols

Consensus Systems

Availability vs Consistency

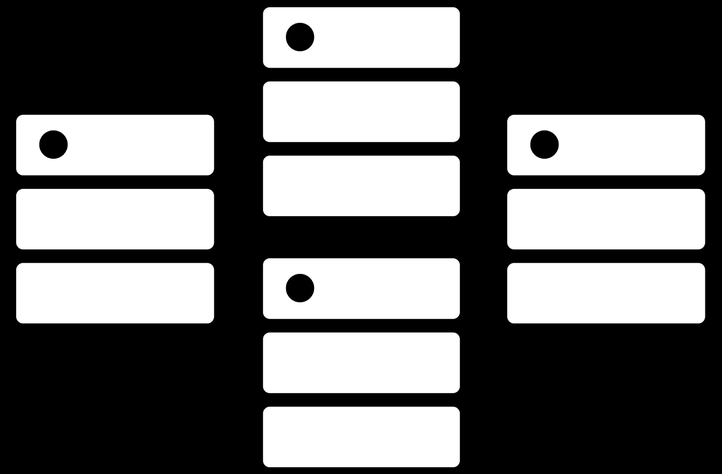
Work Distribution

Random	Consistent	Distributed
Placement	Hashing	Hash Tables

Random Placement

Write Anywhere

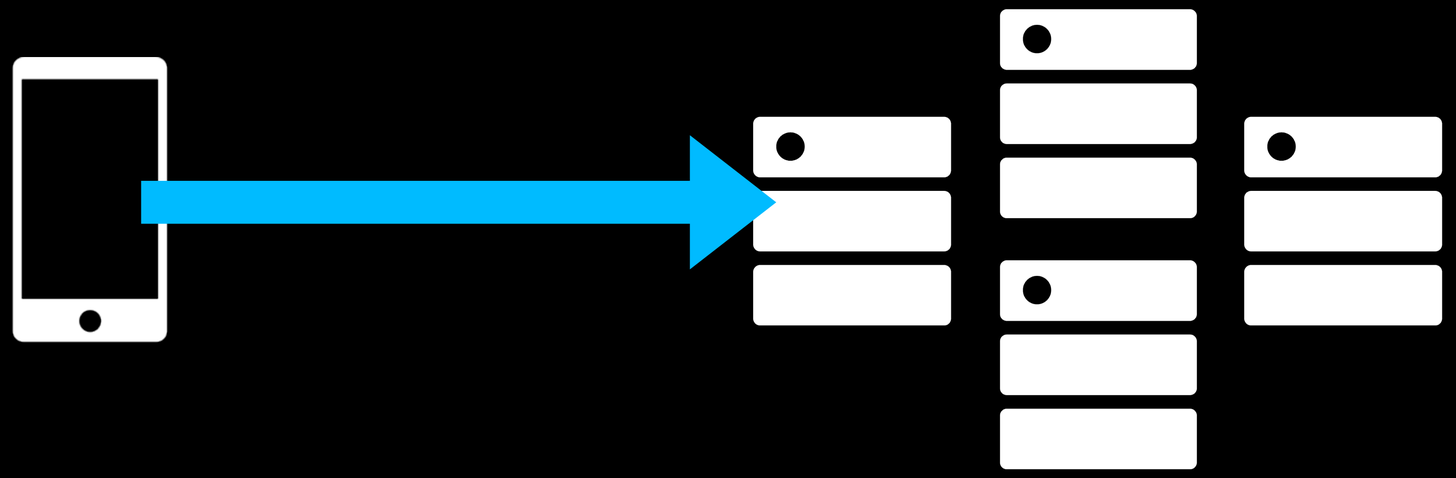
Read from Everywhere



Random Placement

Write
Anywhere

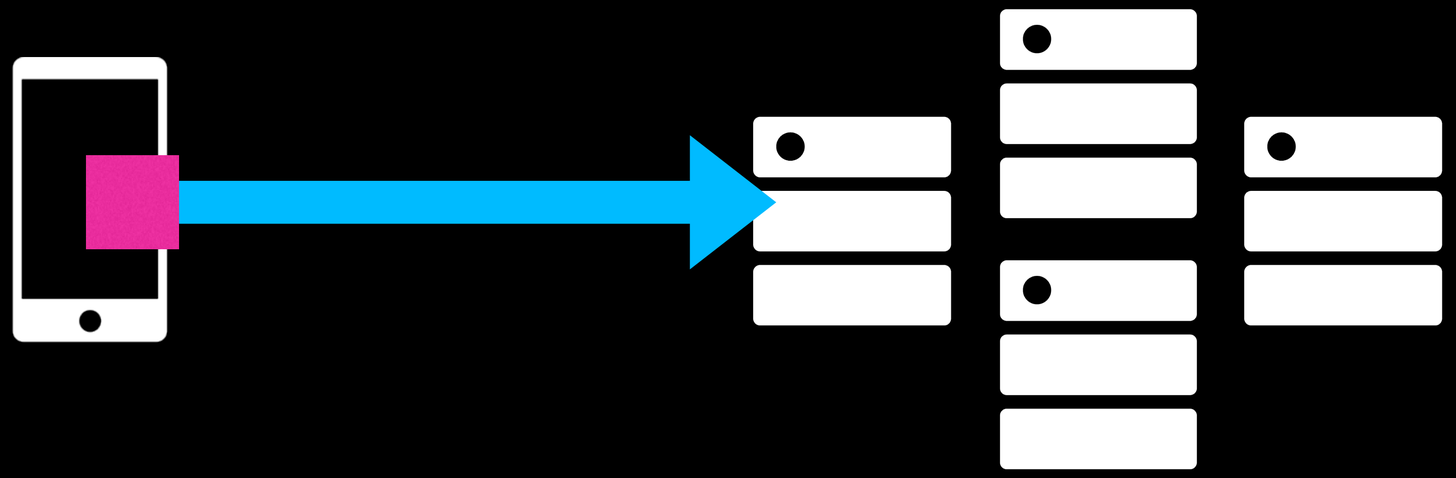
Read from
Everywhere



Random Placement

Write
Anywhere

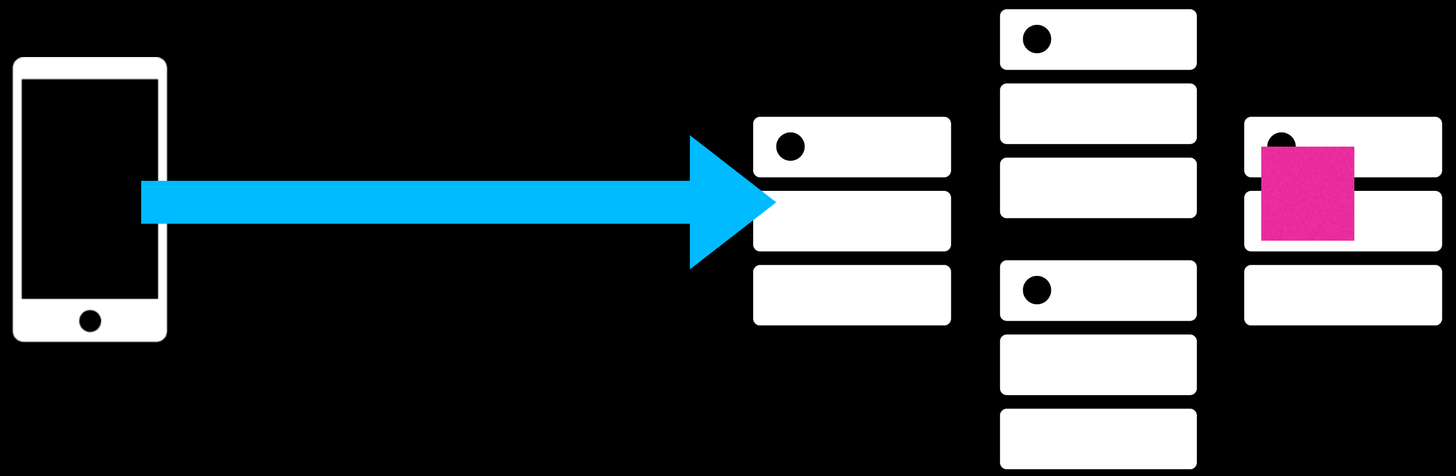
Read from
Everywhere



Random Placement

Write Anywhere

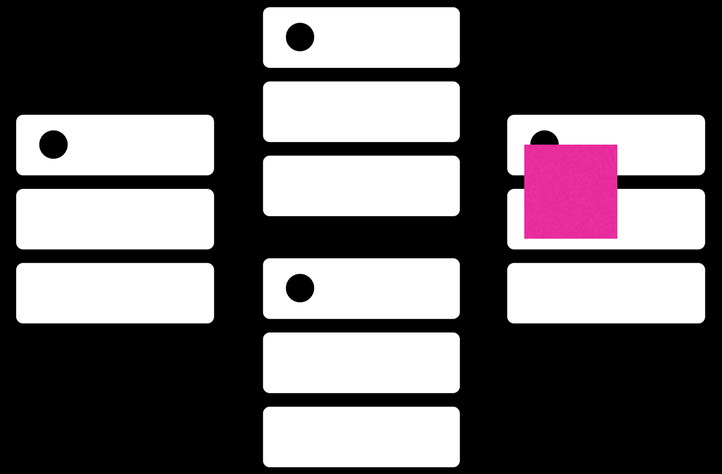
Read from Everywhere



Random Placement

Write Anywhere

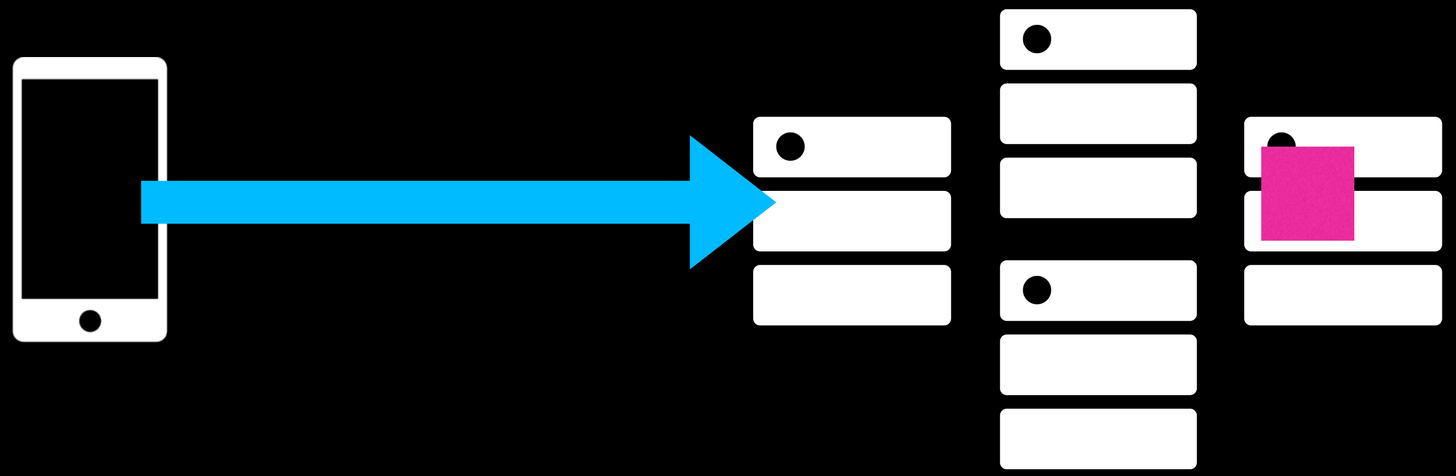
Read from Everywhere



Random Placement

Write Anywhere

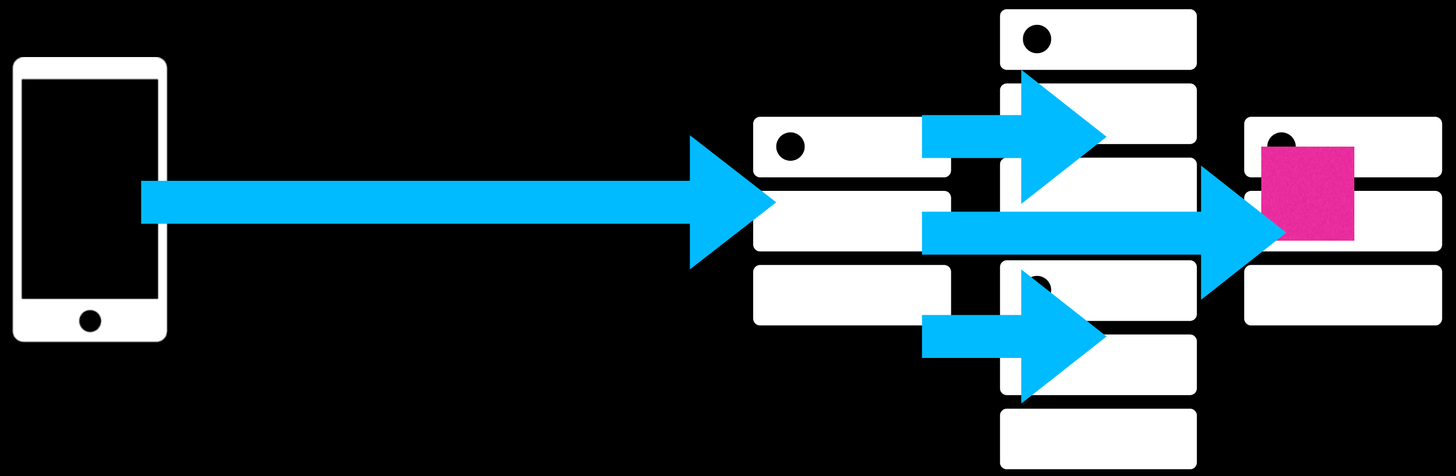
Read from Everywhere



Random Placement

Write Anywhere

Read from Everywhere

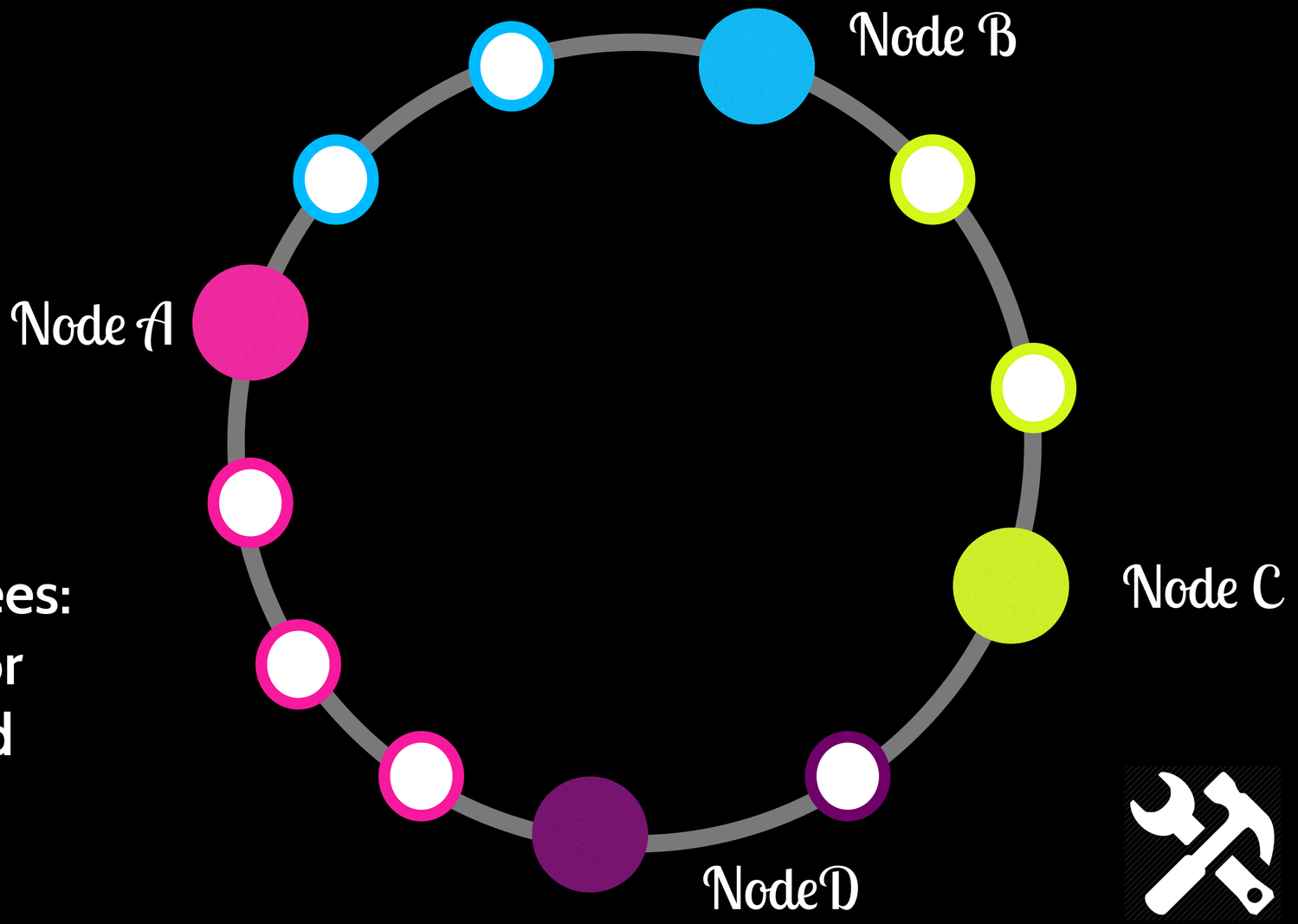


Consistent Hashing

Deterministic Placement

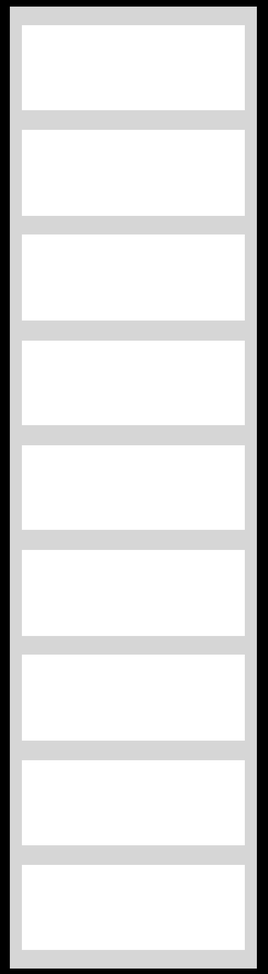
- - - -

Consistent Hashing & Random Trees: Distributed caching protocols for relieving hot spots on the World Wide Web



Distributed Hash Table

Non- Deterministic Placement



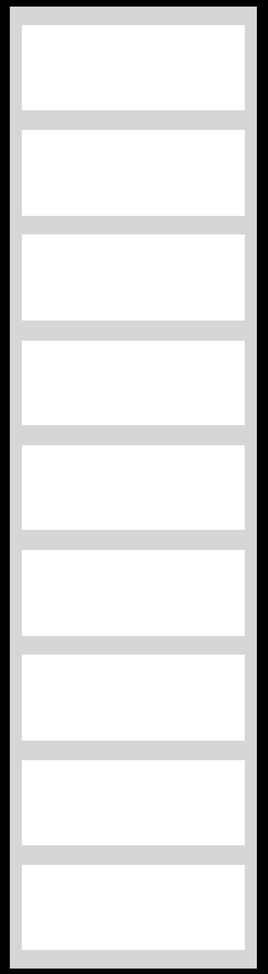
Node A

Node B

Node C

Distributed Hash Table

Non- Deterministic Placement



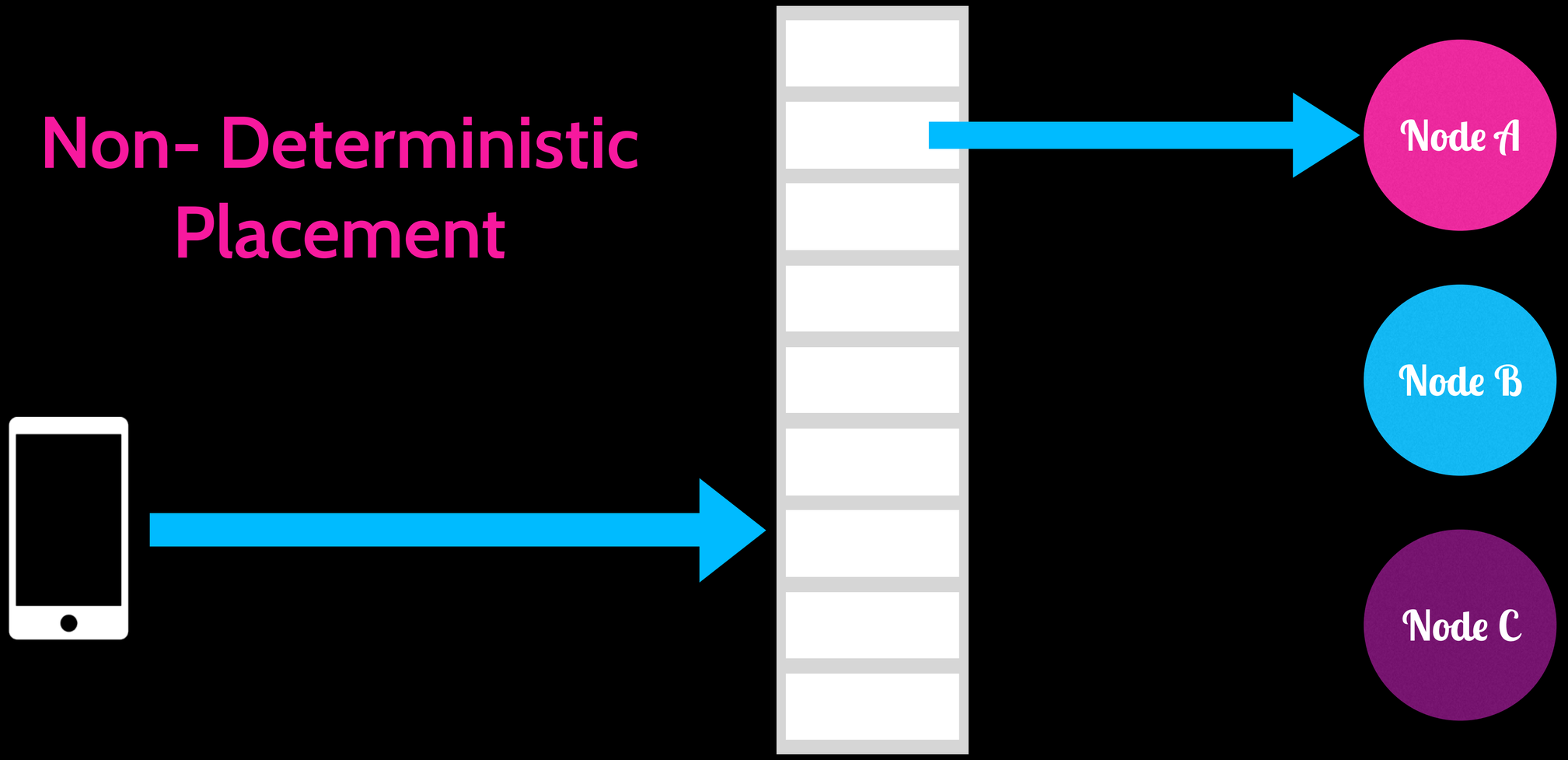
Node A

Node B

Node C

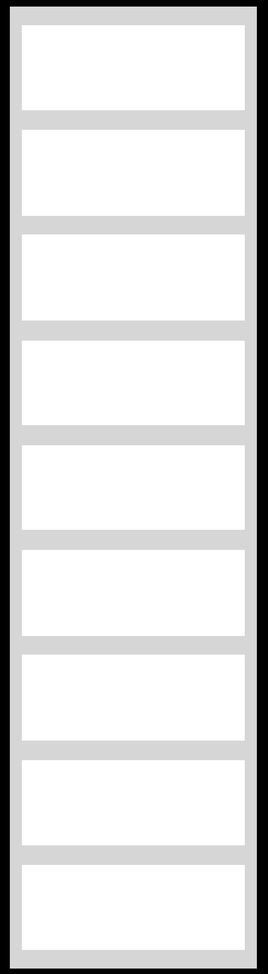
Distributed Hash Table

Non- Deterministic Placement



Distributed Hash Table

Non- Deterministic Placement



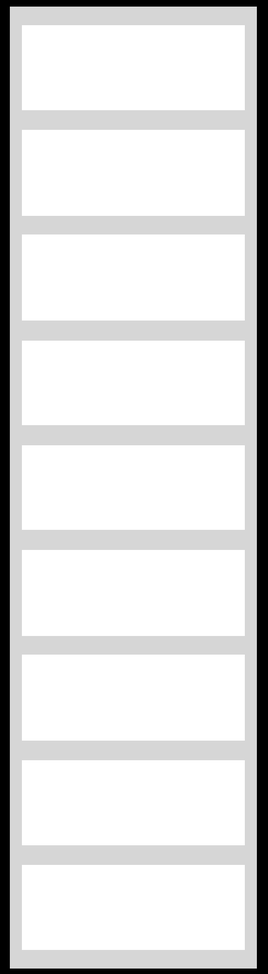
Node A

Node B

Node C

Distributed Hash Table

Non- Deterministic Placement



Node A

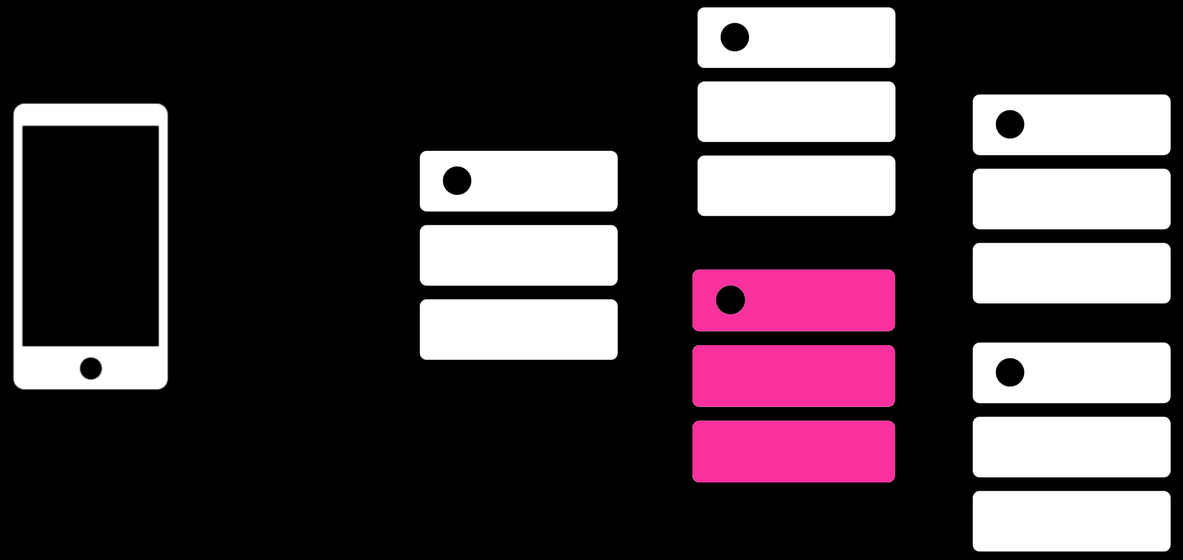
Node B

Node C

Stateful Services

In the Real World

Scuba is a fast, scalable, distributed, in-memory database built at Facebook. It is the workhorse behind code regression analysis & bug report, revenue, and performance debugging

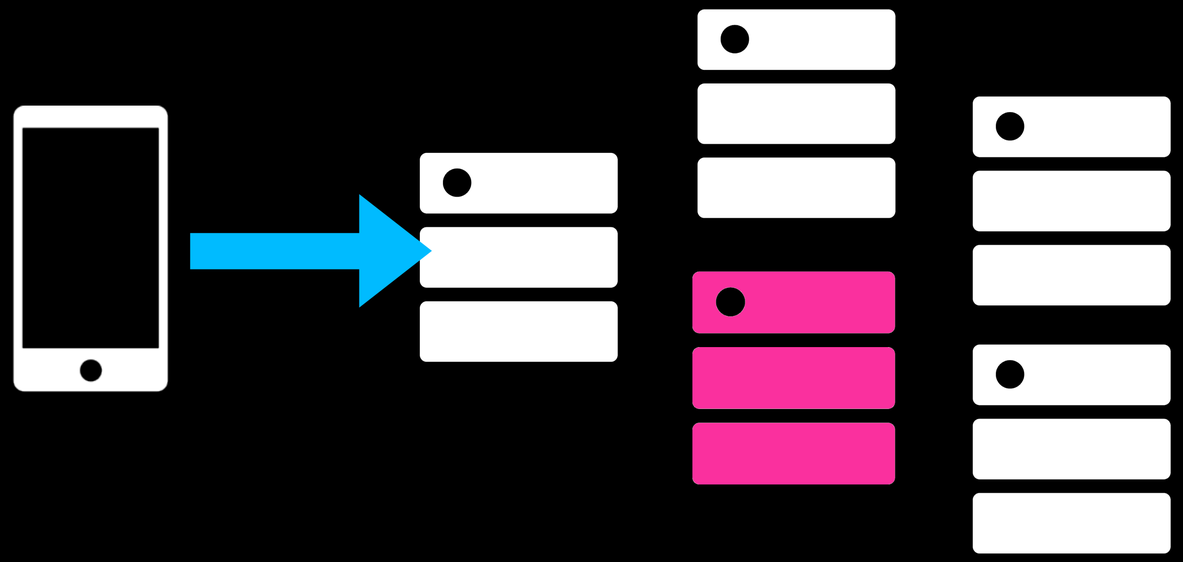


Fan-out request to all machines in the cluster

Compose Results

Return Results and Completeness

Scuba is a fast, scalable, distributed, in-memory database built at Facebook. It is the workhorse behind code regression analysis & bug report, revenue, and performance debugging

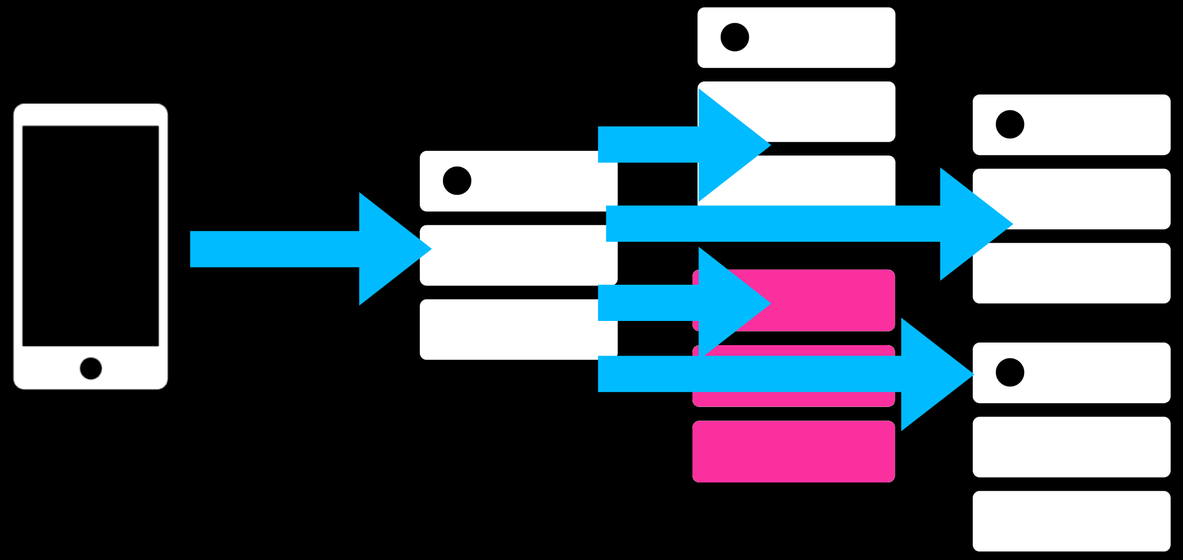


Fan-out request to all machines in the cluster

Compose Results

Return Results and Completeness

Scuba is a fast, scalable, distributed, in-memory database built at Facebook. It is the workhorse behind code regression analysis & bug report, revenue, and performance debugging

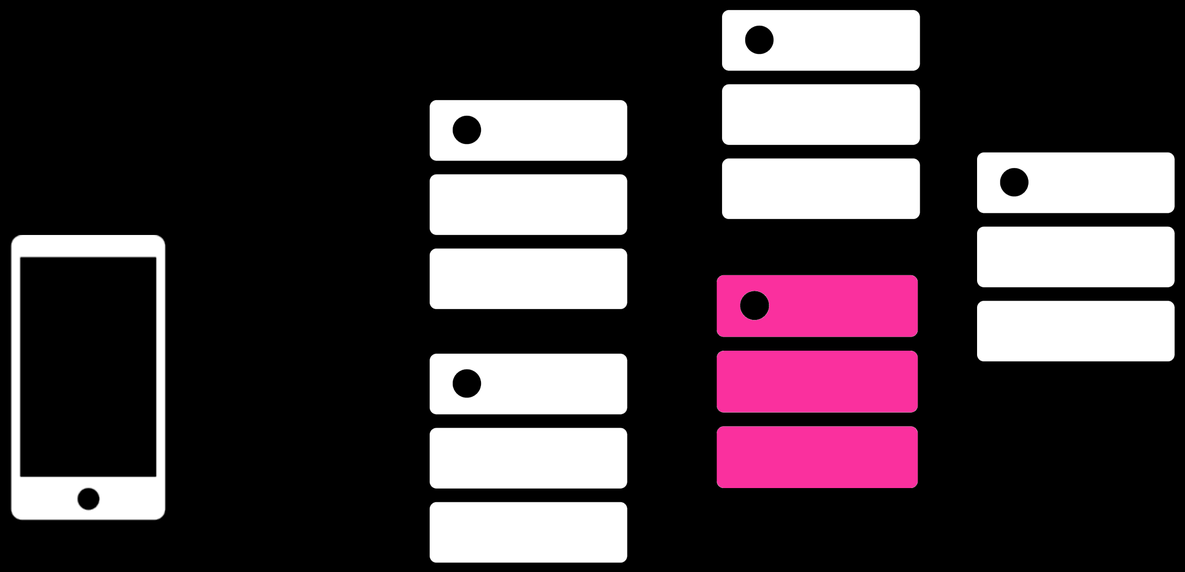


Fan-out request to all machines in the cluster

Compose Results

Return Results and Completeness

Uber Ringpop is an open-source Node.js library that brings application-layer sharding to many of their dispatching platform services.

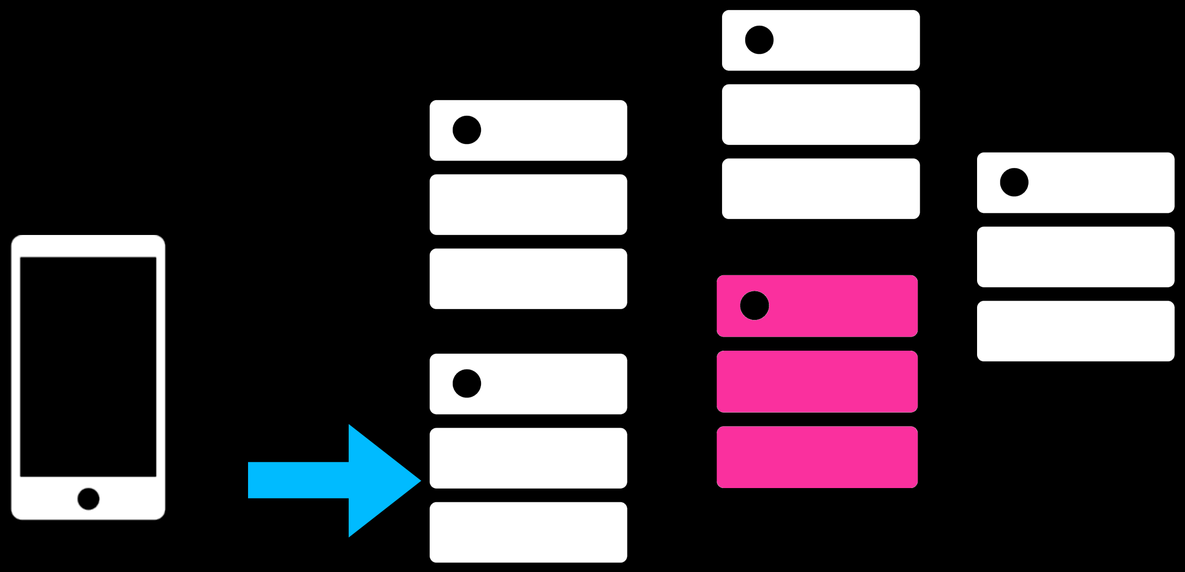


Swim Gossip Protocol

+

Consistent Hashing

Uber Ringpop is an open-source Node.js library that brings application-layer sharding to many of their dispatching platform services.

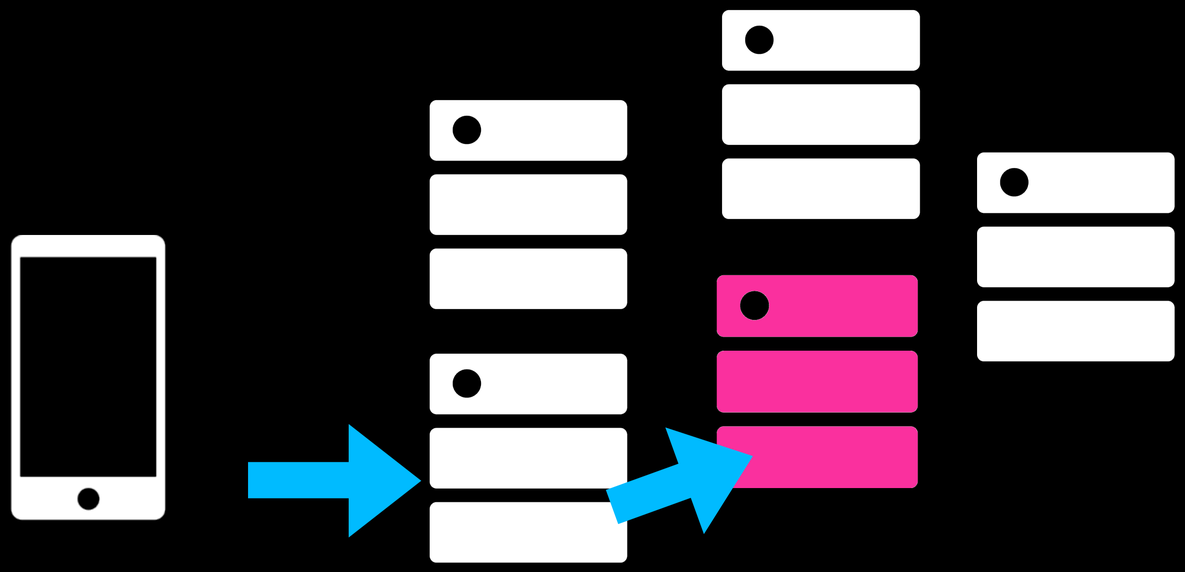


Swim Gossip Protocol

+

Consistent Hashing

Uber Ringpop is an open-source Node.js library that brings application-layer sharding to many of their dispatching platform services.

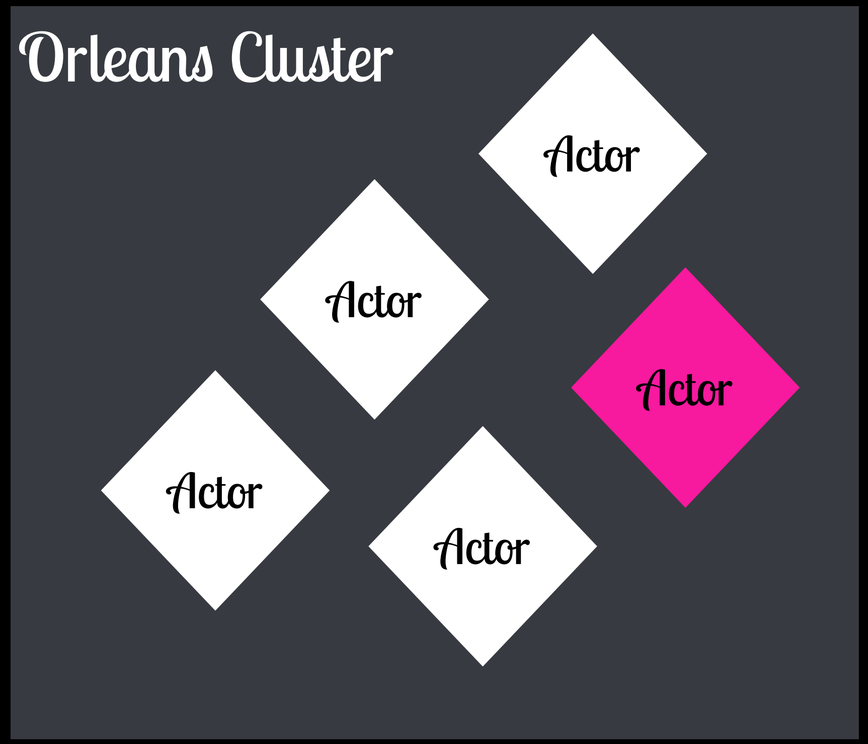


Swim Gossip Protocol

+

Consistent Hashing

Orleans is a runtime and Programming model for building distributed systems based on the Actor Model from the eXtreme Computing Group at MSR



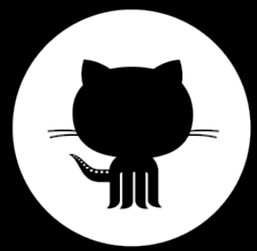
Gossip Protocol

+

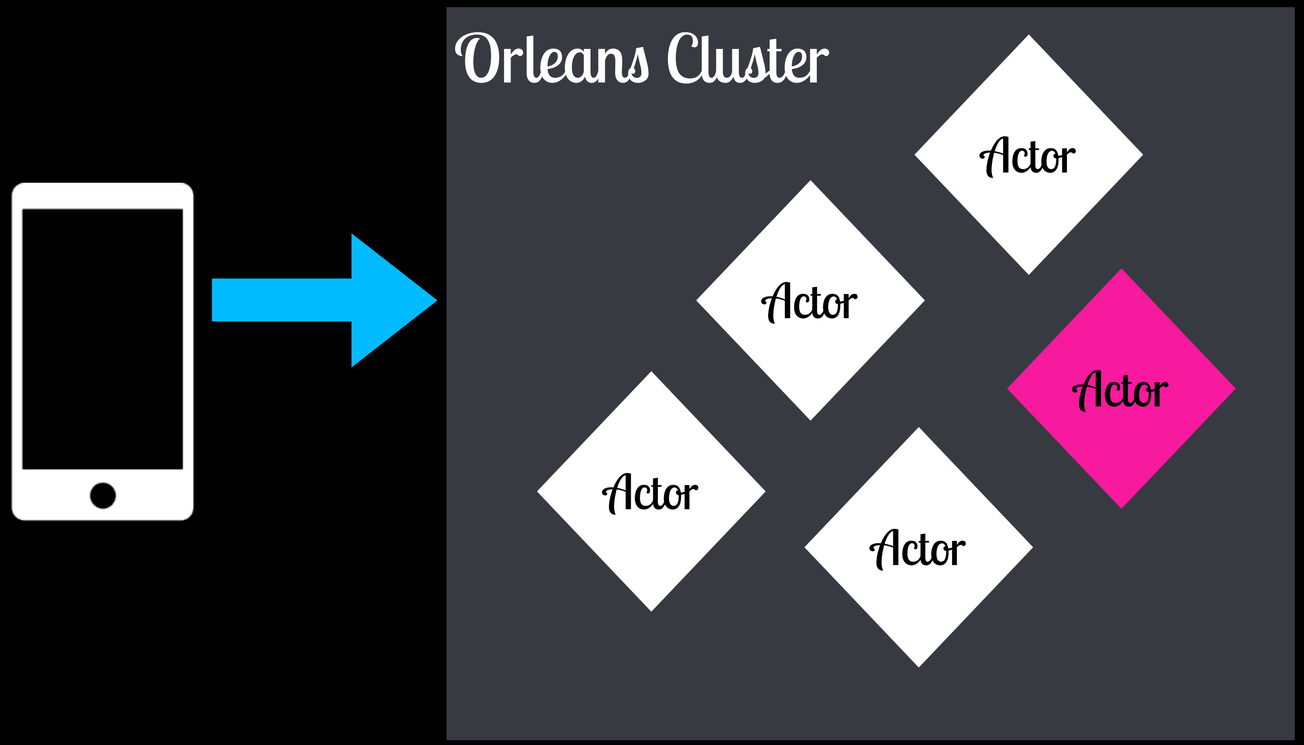
Consistent Hashing

+

Distributed Hash Table



Orleans is a runtime and Programming model for building distributed systems based on the Actor Model from the eXtreme Computing Group at MSR



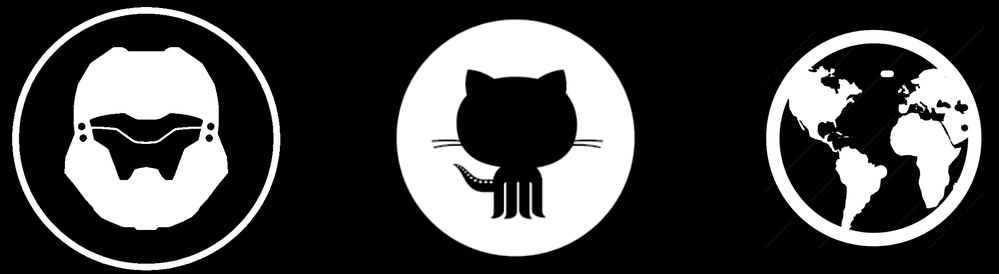
Gossip Protocol

+

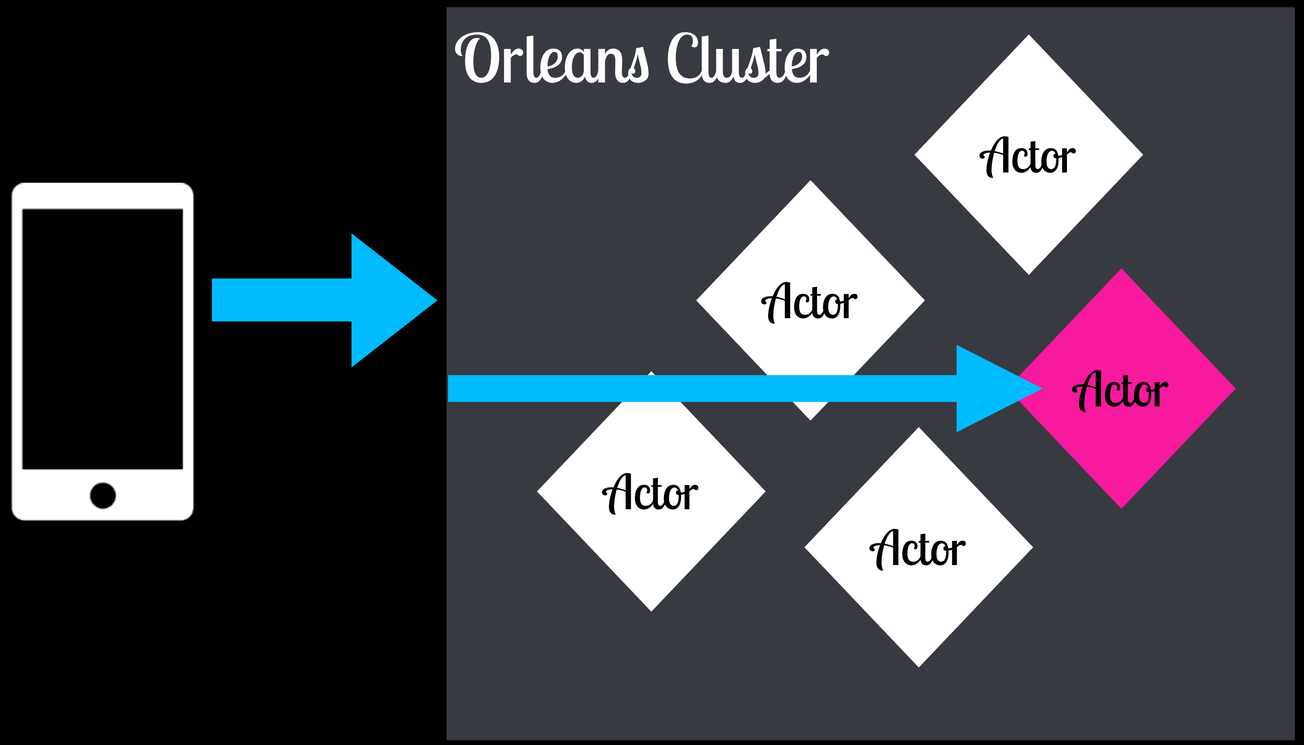
Consistent Hashing

+

Distributed Hash Table



Orleans is a runtime and Programming model for building distributed systems based on the Actor Model from the eXtreme Computing Group at MSR



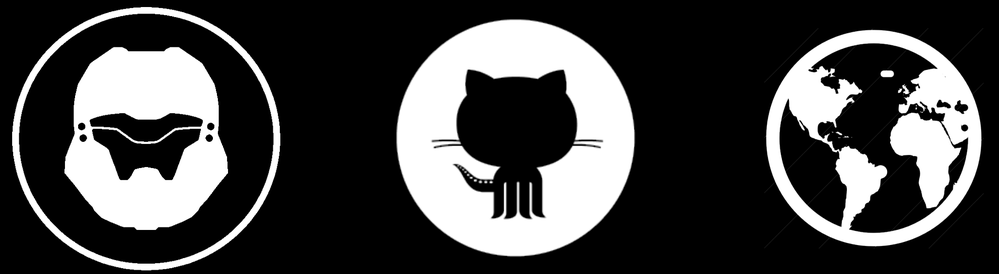
Gossip Protocol

+

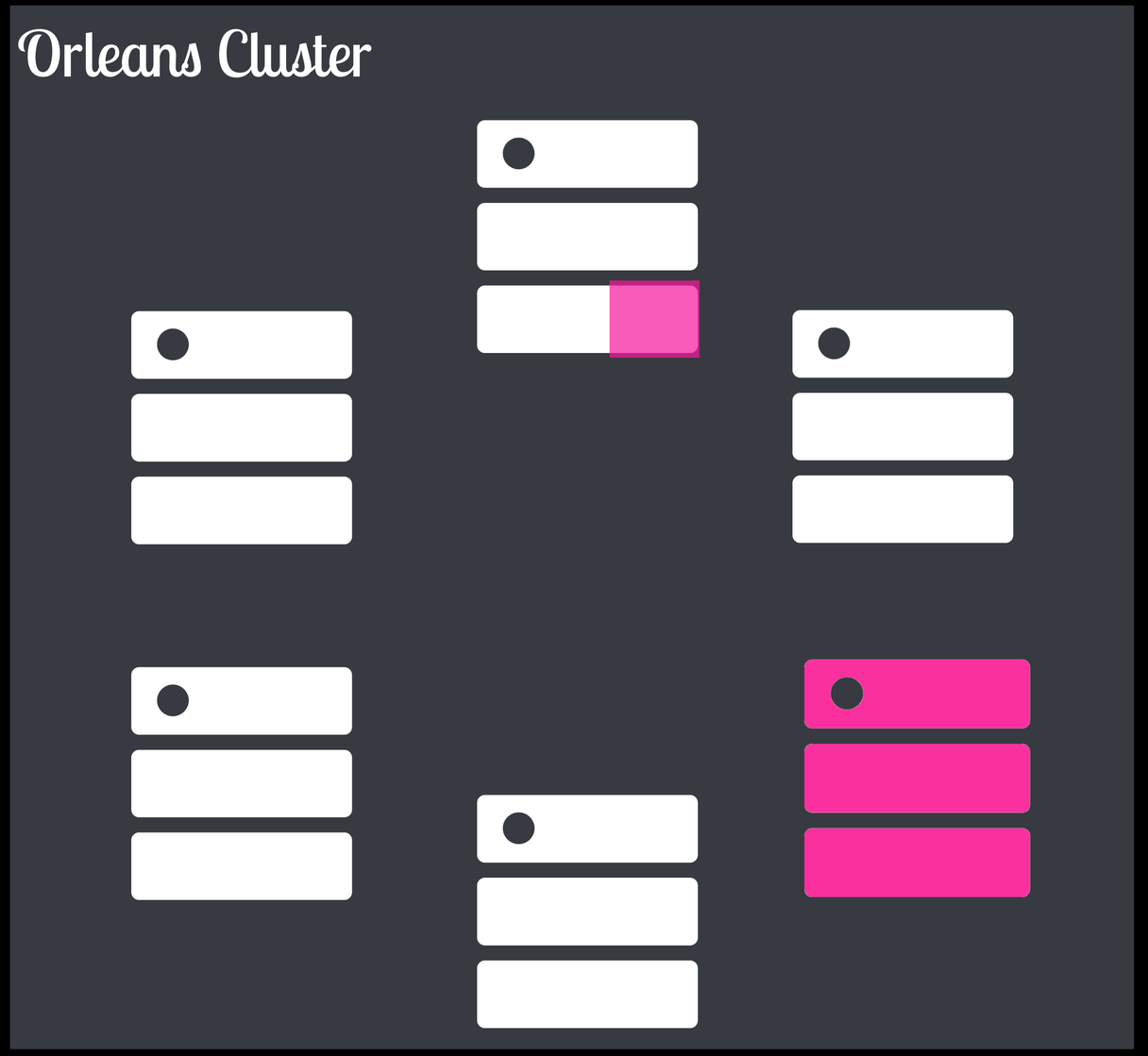
Consistent Hashing

+

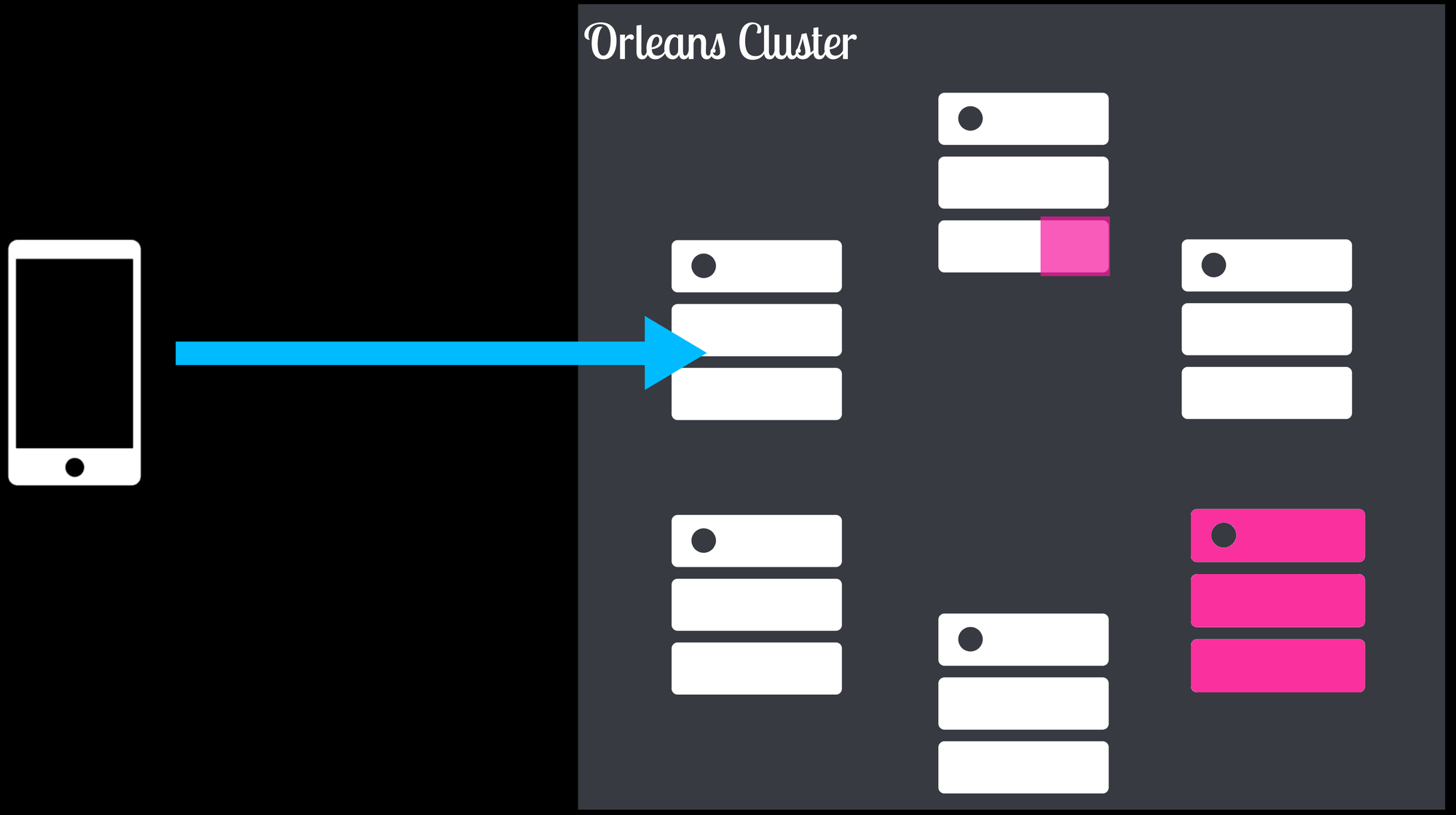
Distributed Hash Table



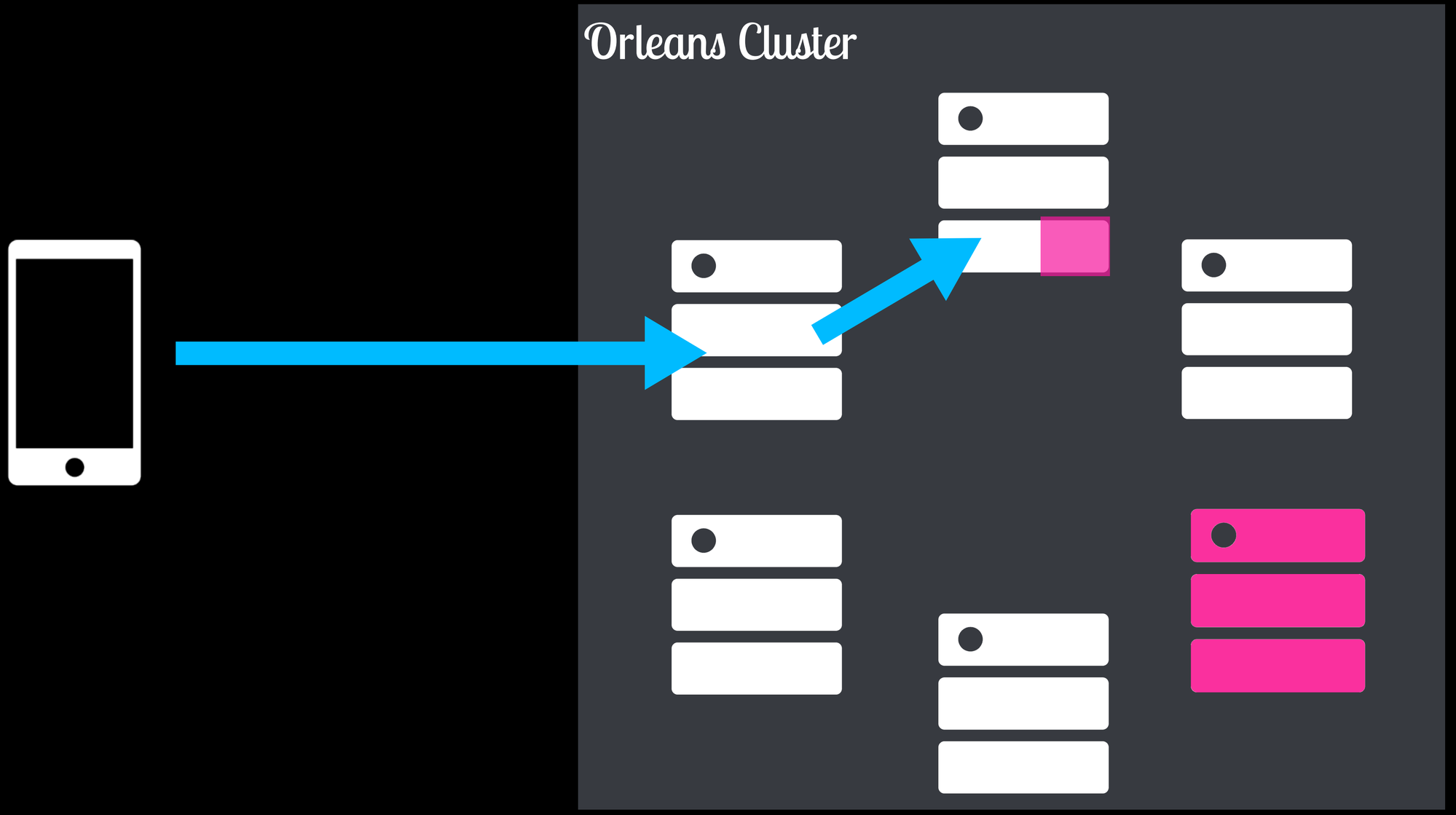
Orleans Distributed Hash Table



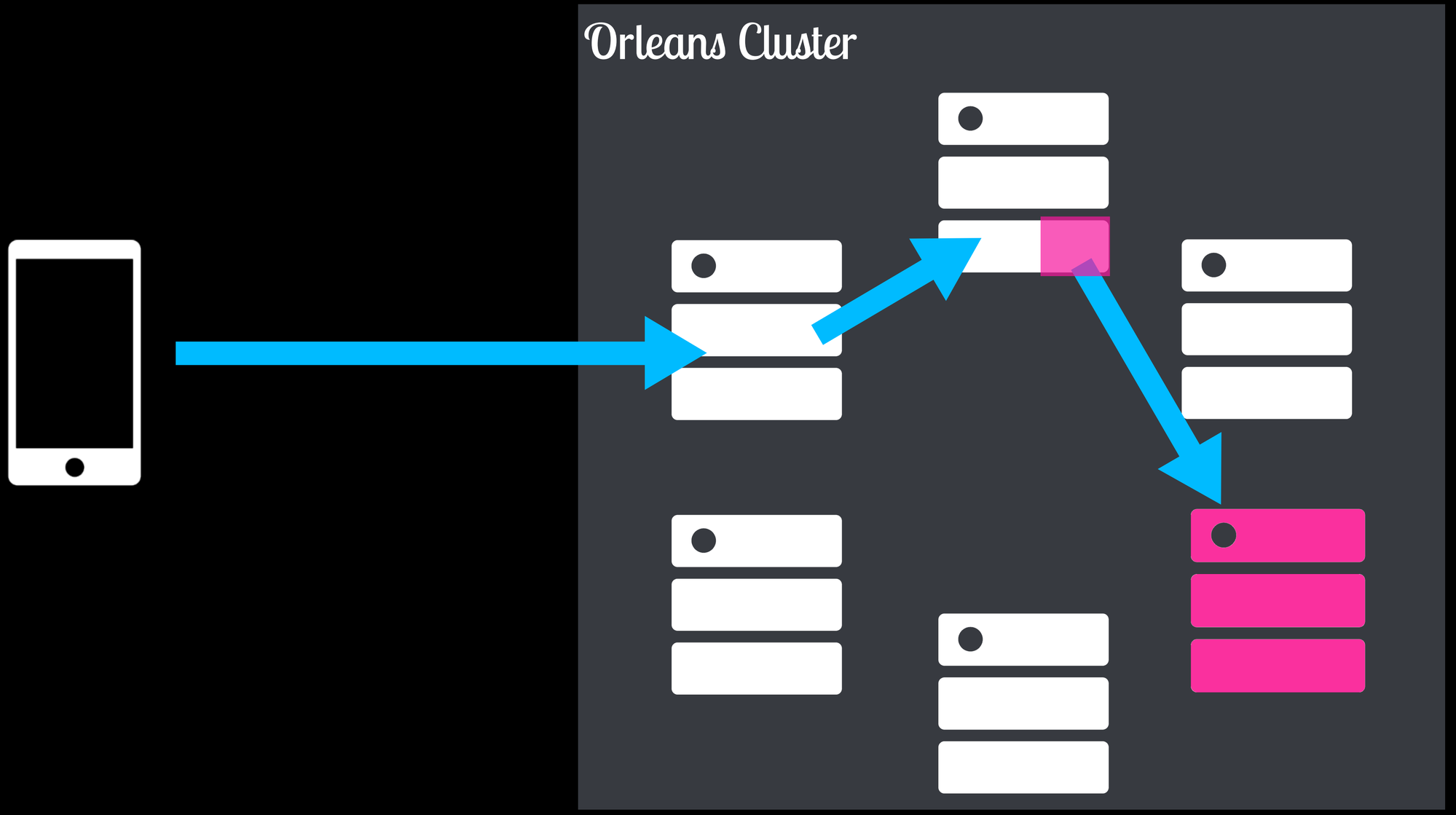
Orleans Distributed Hash Table



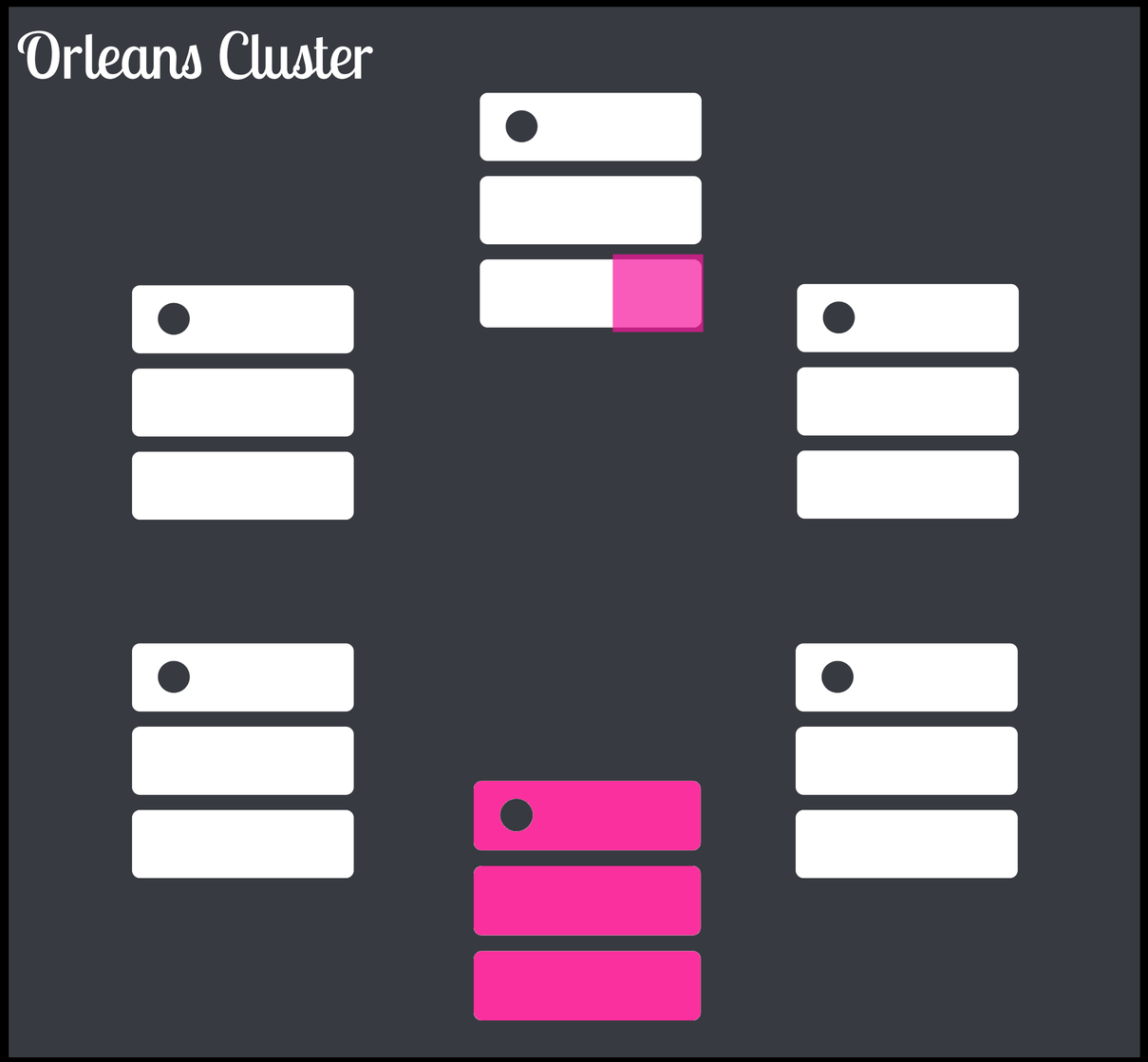
Orleans Distributed Hash Table



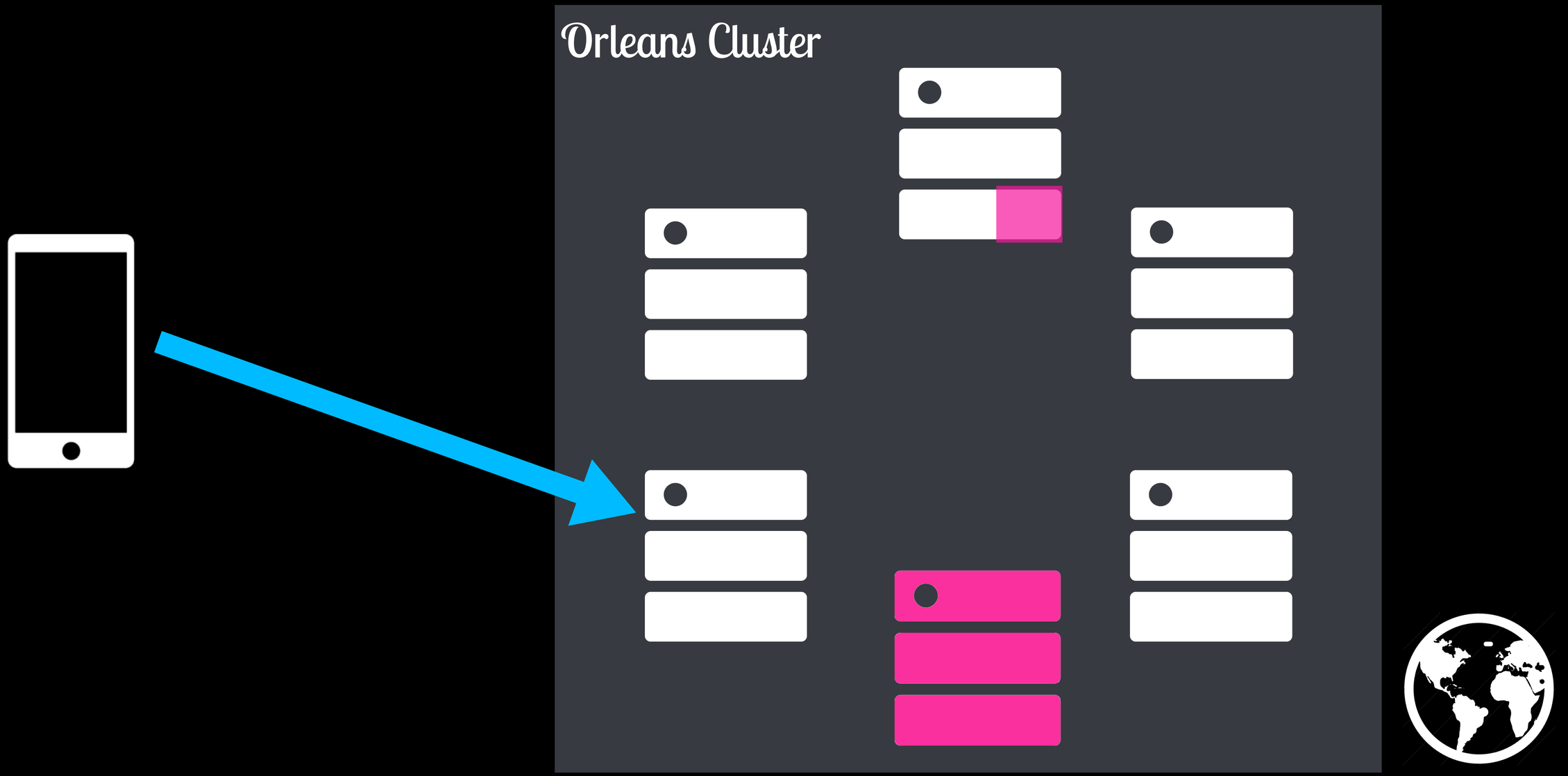
Orleans Distributed Hash Table



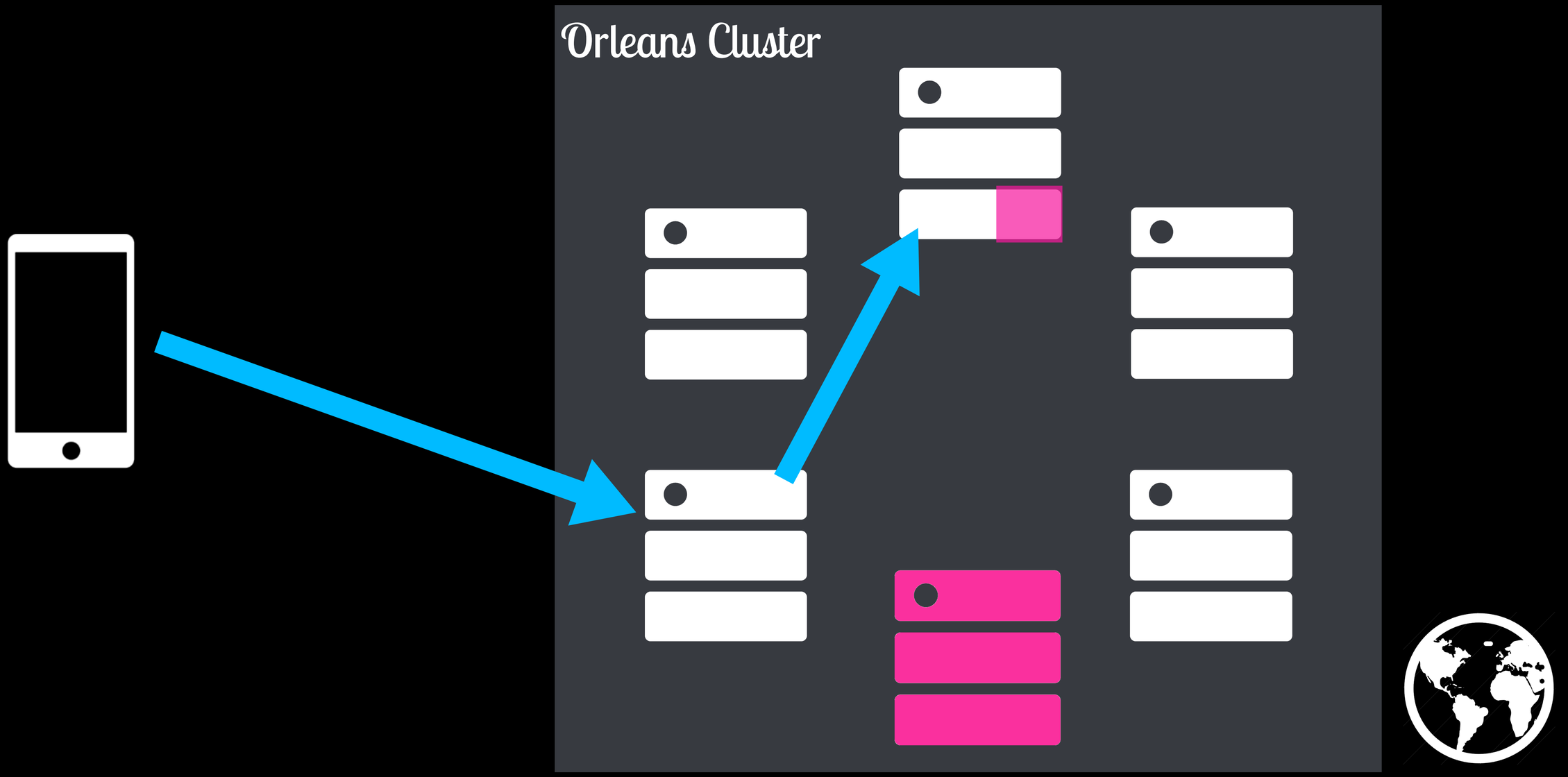
Orleans Distributed Hash Table



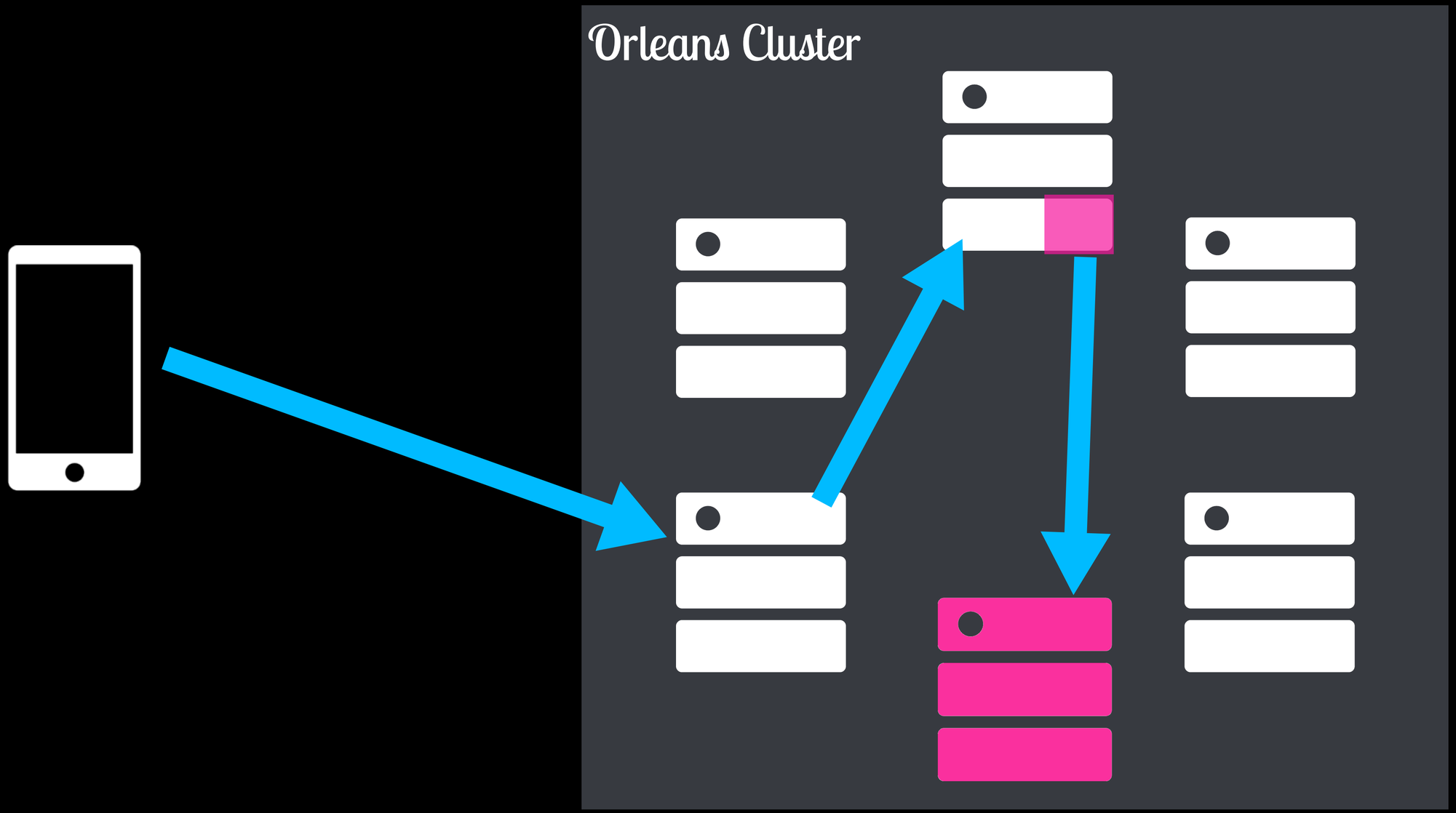
Orleans Distributed Hash Table



Orleans Distributed Hash Table



Orleans Distributed Hash Table



Caution

Stateful Services Ahead

Unbounded Data Structures

:

Implicit Assumptions are the Killer of Distributed Systems

Memory Management

Get Ready to Make Friends with the Garbage Collector Profiler

Reloading State

- First Connection
- Recovering From Crashes
- Deploying New Code

Fast Restarts at Facebook

"Our Key Observation is that we can decouple the memory lifetime from the process lifetime. When we shutdown a server for a planned upgrade."

Conclusion

Data Locality & Available Consistency

Cluster Membership & Work Distribution

Successful Statefull Real World Systems

Caution: Some New Challenges

Should I Read Papers?

Should I Read Papers?

YES!

Thank You

Kyle Kingsbury
Chris Meiklejohn
Tom Santero
Ines Sombra

 @Caitie