# Distributed Systems

## Spring Semester 2020

## Lecture 14: PNUTS

**John Liagouris**
**liagos@bu.edu**

# Why this paper

- Massive Global scale system
- Performance + Flexible Consistency
- Still explicitly designed but less complicated than Google Spanner (also less powerful)

DATABASE FEATURES not KV store based
but uses a Pub Sub communications layer

# The Big Picture

Multiple Datacenters in Regions

—

Entire Database replicated


Region 1


Region 2


Region 3

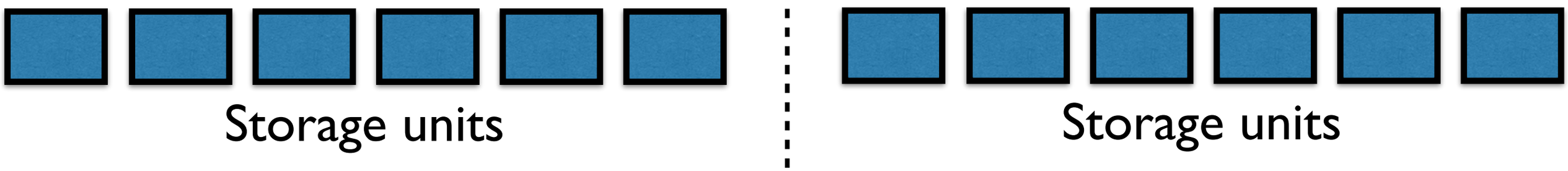
# Replication

- GOAL: Make the reads fast!
  - Clients connect to local data center
    - Web app in that data center can do the reads fast — no remote communication
  - BUT reads possible stale
- TRADEOFF: Writes slow!

# PNUTS

REGION 1

REGION 2

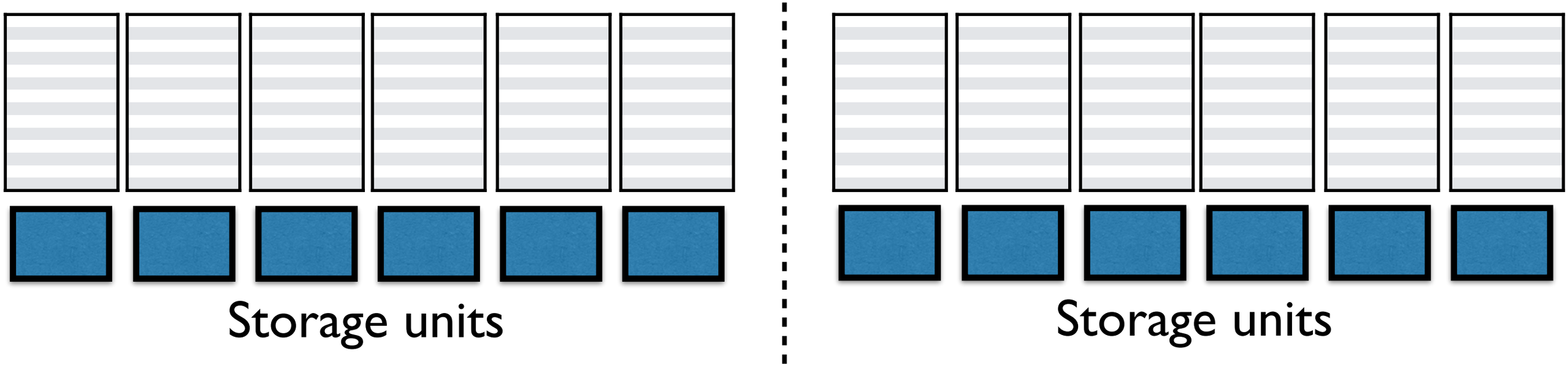



get(), set(), scan()

# PNUTS

REGION 1

REGION 2

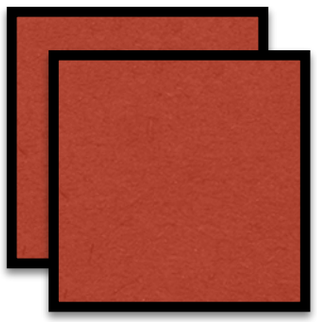


# PNUTS

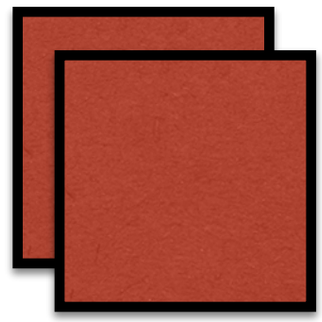
REGION 1

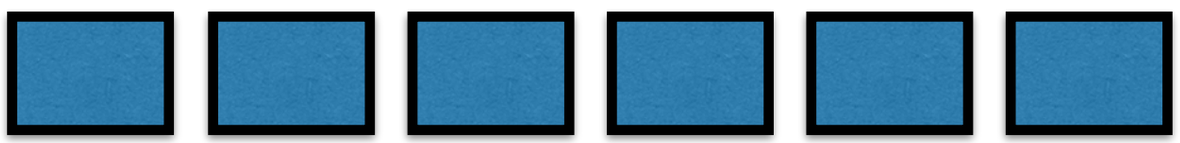
REGION 2

Tablet controller

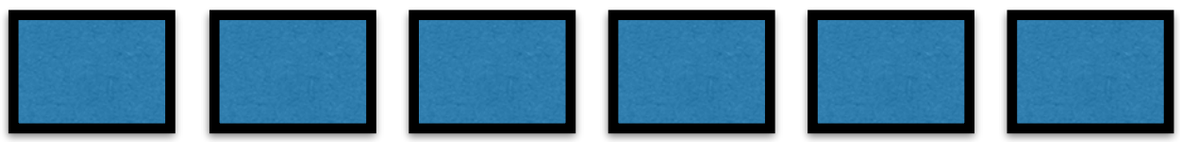


Tablet controller



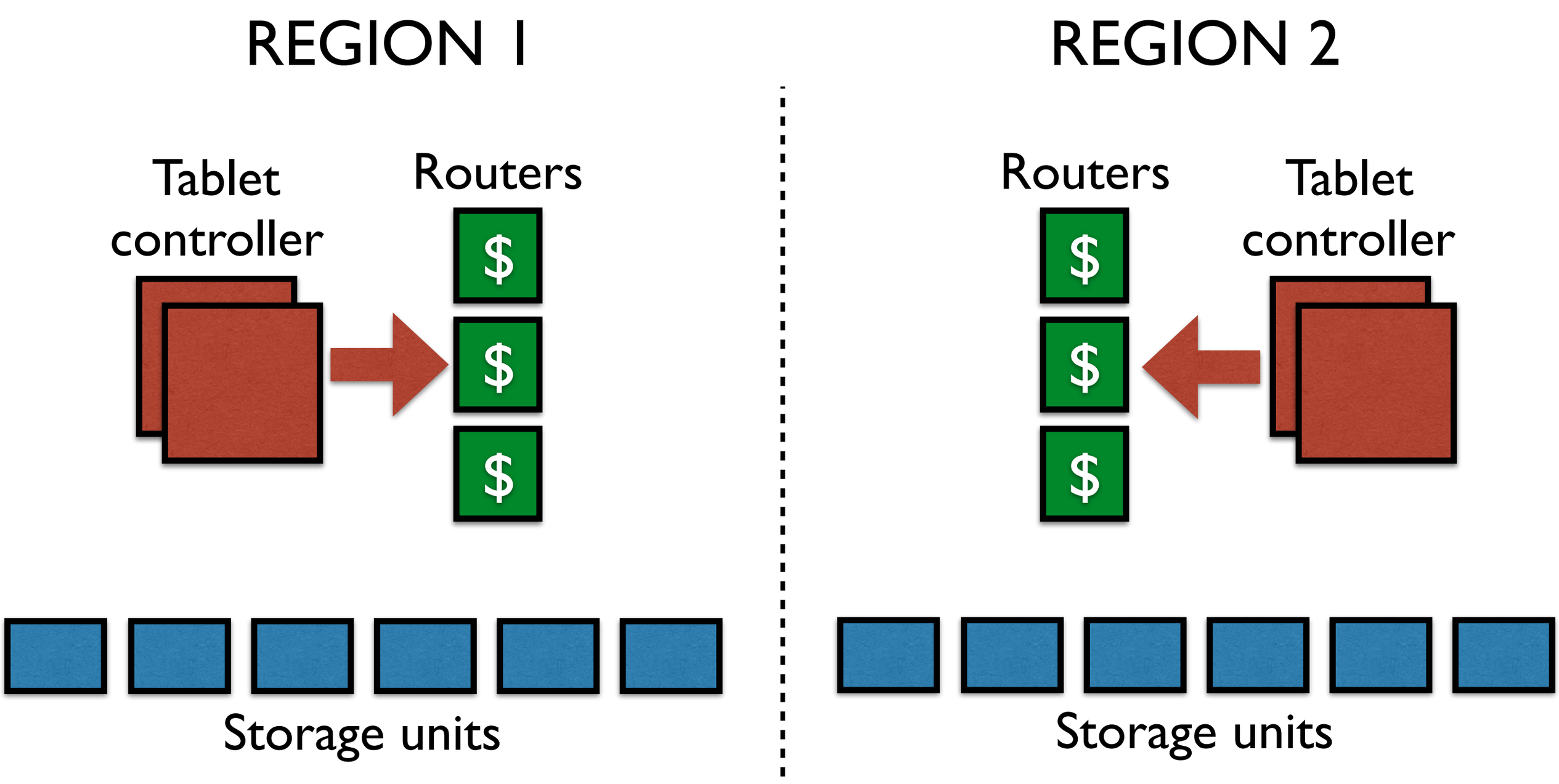


Storage units

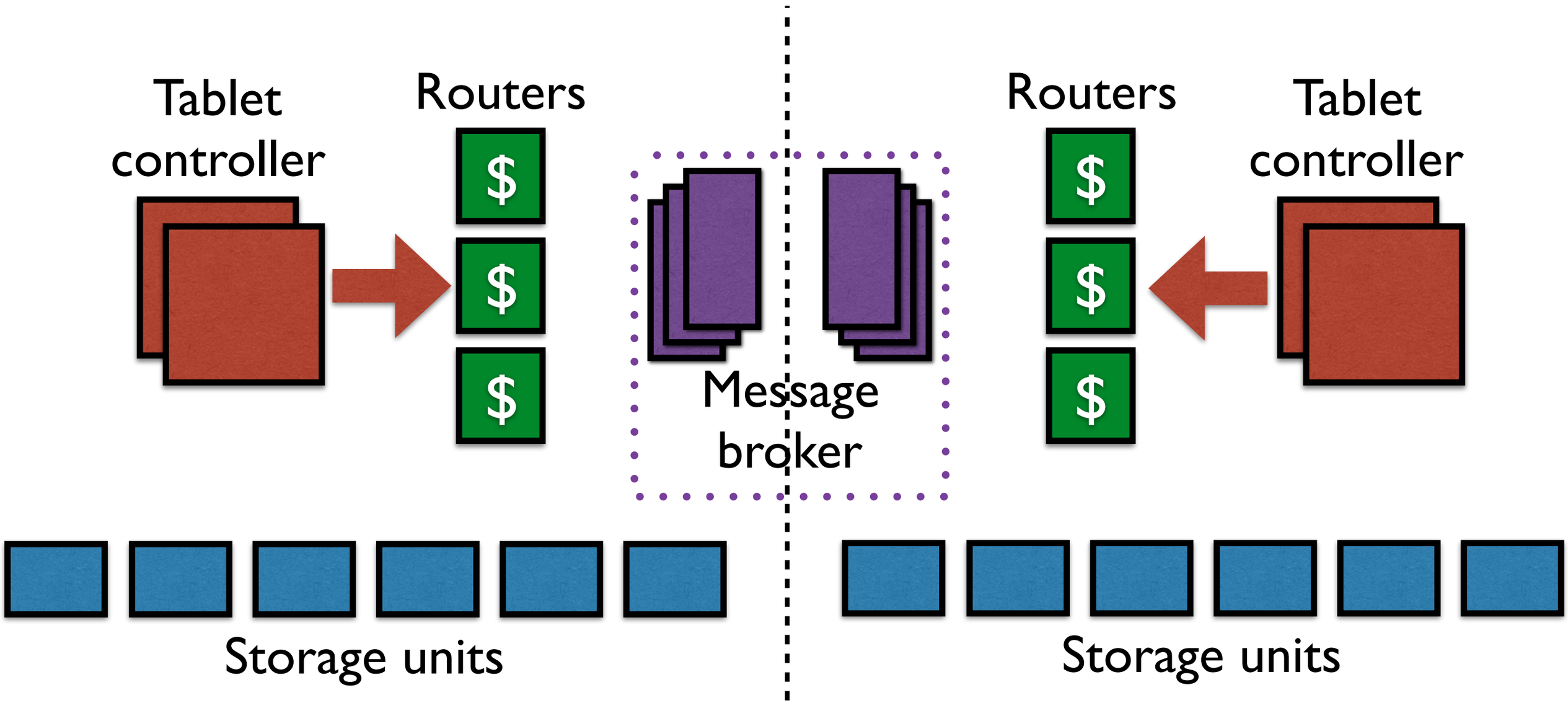


Storage units

# PNUTS


REGION 1
REGION 2
Tablet controller
Routers
Routers
Tablet controller
$
$
$
$
$
$
Storage units
Storage units

# PNUTS


REGION 1
REGION 2
Tablet controller
Routers
$
$
$
Message broker
Routers
$
$
$
Tablet controller
Storage units
Storage units

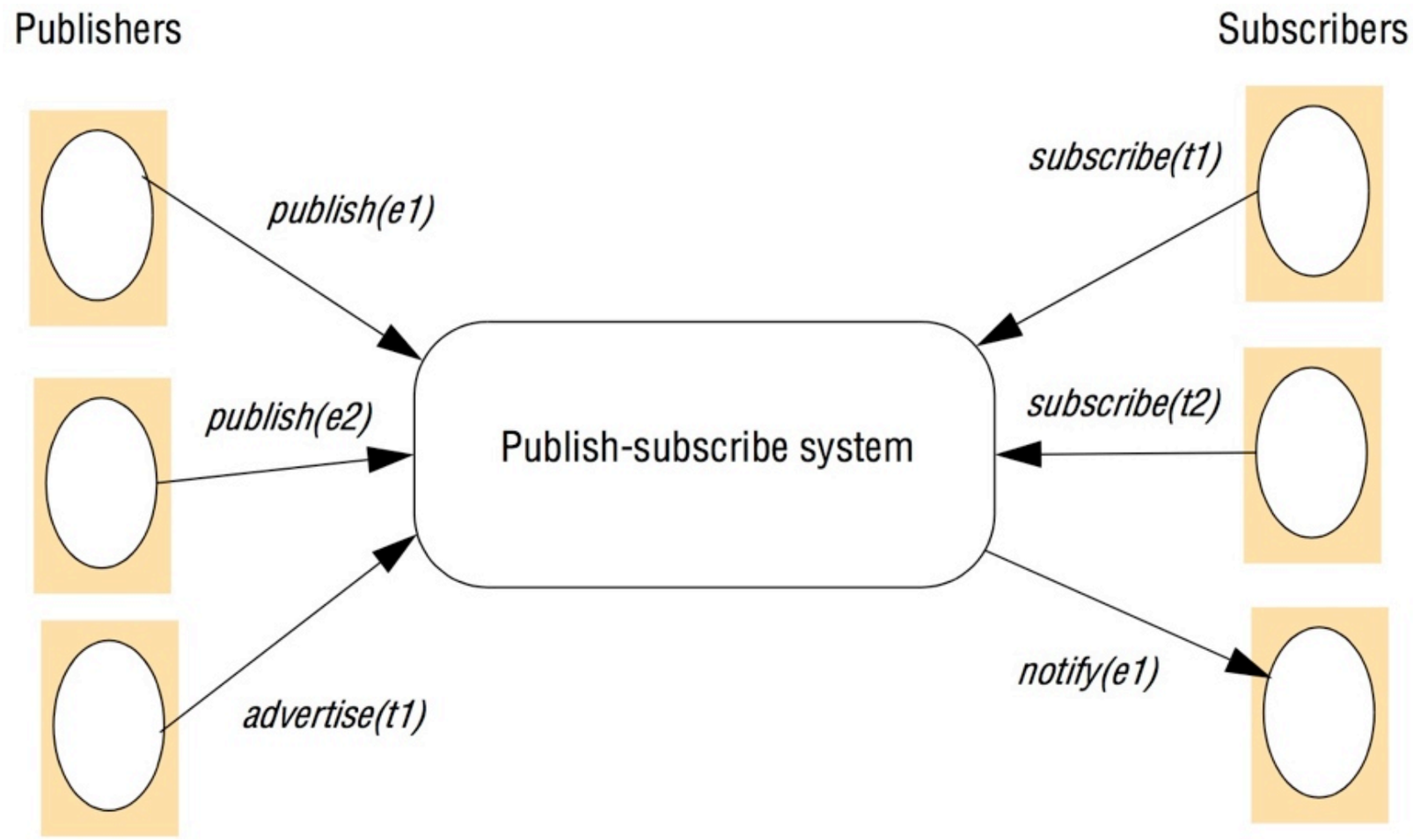
# Publish-subscribe

## Distributed Event Based Systems

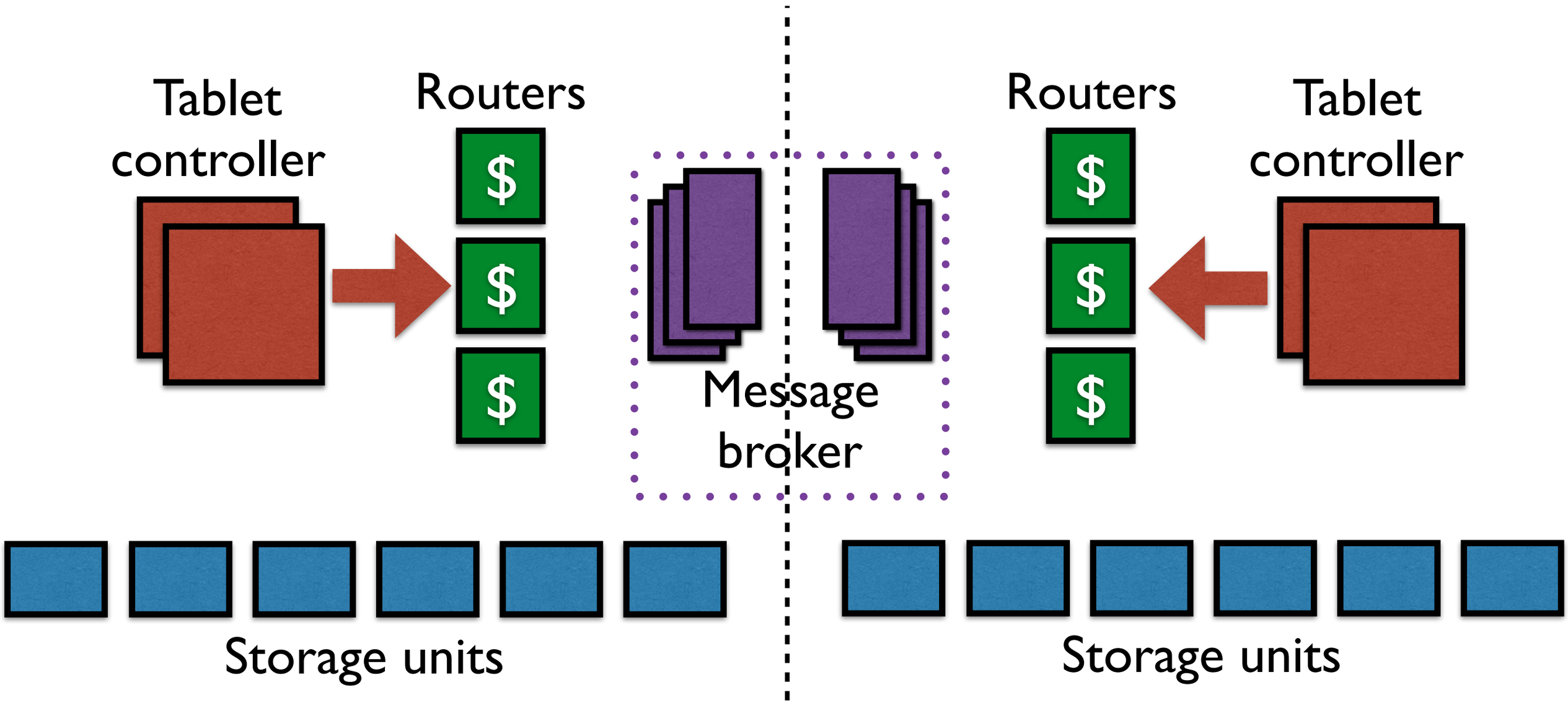
- "Decoupled and reactive style of programming"
- ***Publishers*** publish structured events to an event service
- ***Subscribers*** subscribe to particular events through subscriptions
- System must match subscriptions against published events and ensure the correct delivery of ***event notifications***
- ***one-to-many*** paradigm of communications
- Asynchronous
- ***Delivery guarantees:*** reliability, agreement, latency, etc.

# Publish-subscribe

## Distributed Event Based Systems



# PNUTS


REGION 1
REGION 2
Tablet controller
Routers
$
$
$
Message broker
Routers
$
$
$
Tablet controller
Storage units
Storage units

# Updates

REGION 1

webserver

App
r/w to PNUTS

Tablet

Routers

controller

$

$

$

Message broker

REGION 2

Routers

Tablet

controller

$

$

$

Storage units

Storage units

# Updates

REGION 1

REGION 2

Why not just have app logic send update to every region?

Tablet

Routers

Routers

Tablet

controller

$

$

$

Message broker

$

$

$

controller

Storage units

Storage units

# Updates

REGION 1

REGION 2

What if app crashes after updating only some regions?
What if concurrent updates to same record?

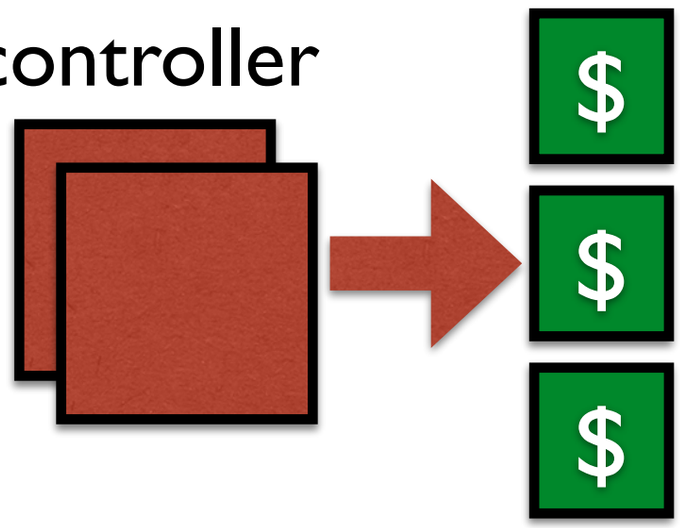
Tablet

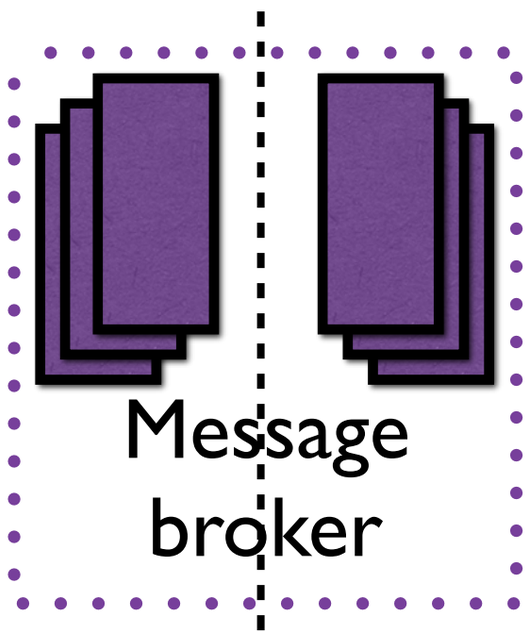
Routers

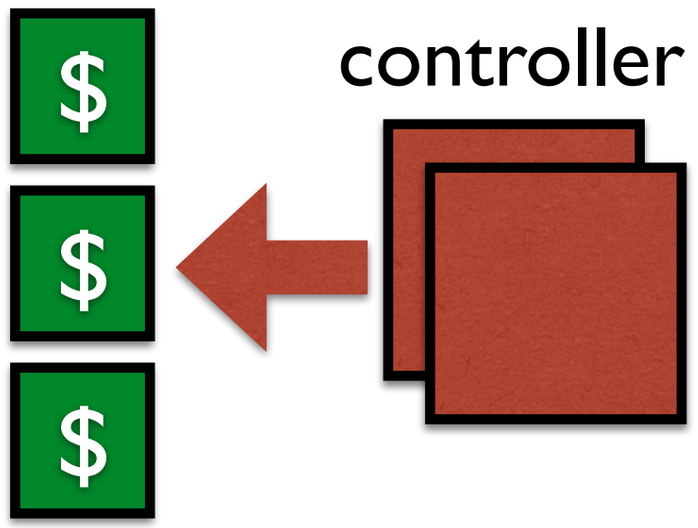
Routers

Tablet

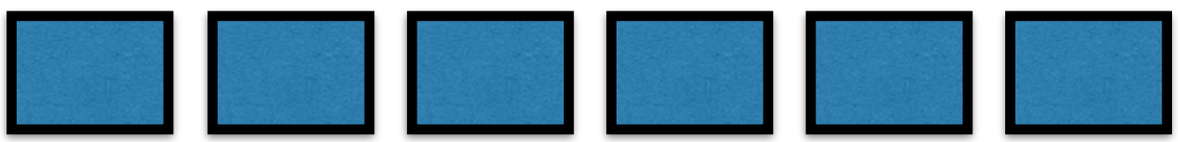
controller



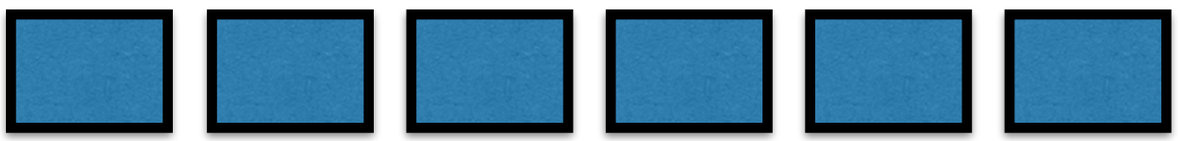




controller



Storage units



Storage units

# Updates

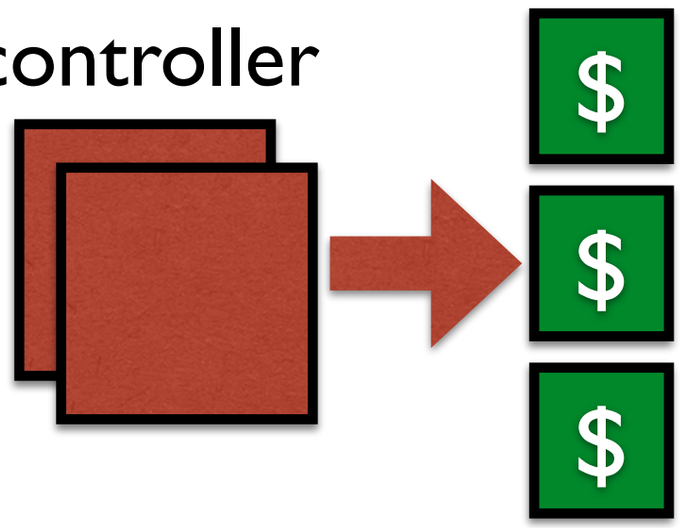
REGION 1

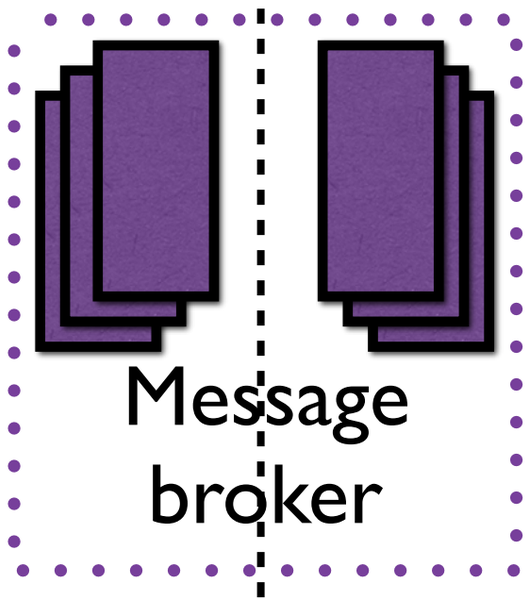
REGION 2

So what does PNUTs do?

Tablet controller

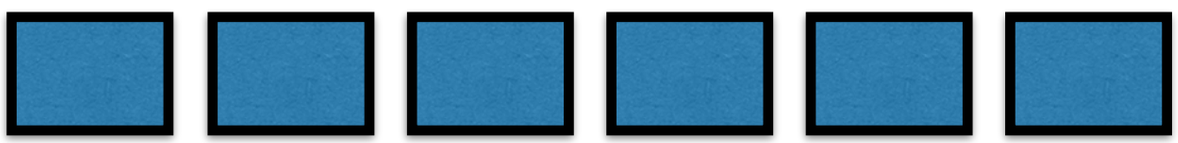
Routers



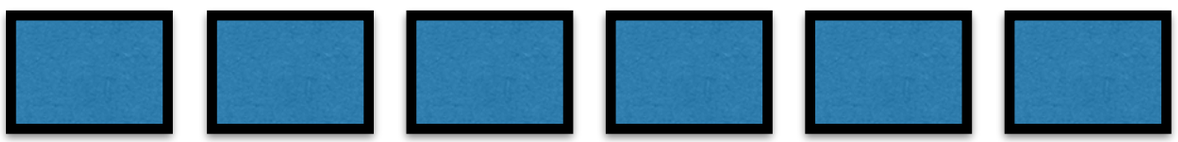




Routers


Tablet controller



Storage units



Storage units