facebook

Managing MySQL at Scale

Pradeep Nayak & Junyi (Luke) Lu

Production Engineers - MySQL Infra

Agenda

1 Terminology

2 Lifecycle of a MySQL instance

3 How do we migrate MySQL instance

4 How do we migrate shard

5 Balancing

6 Testing Infrastructure for automations

Terminology

Terminology

What’s a instance ?

What’s a shard ?

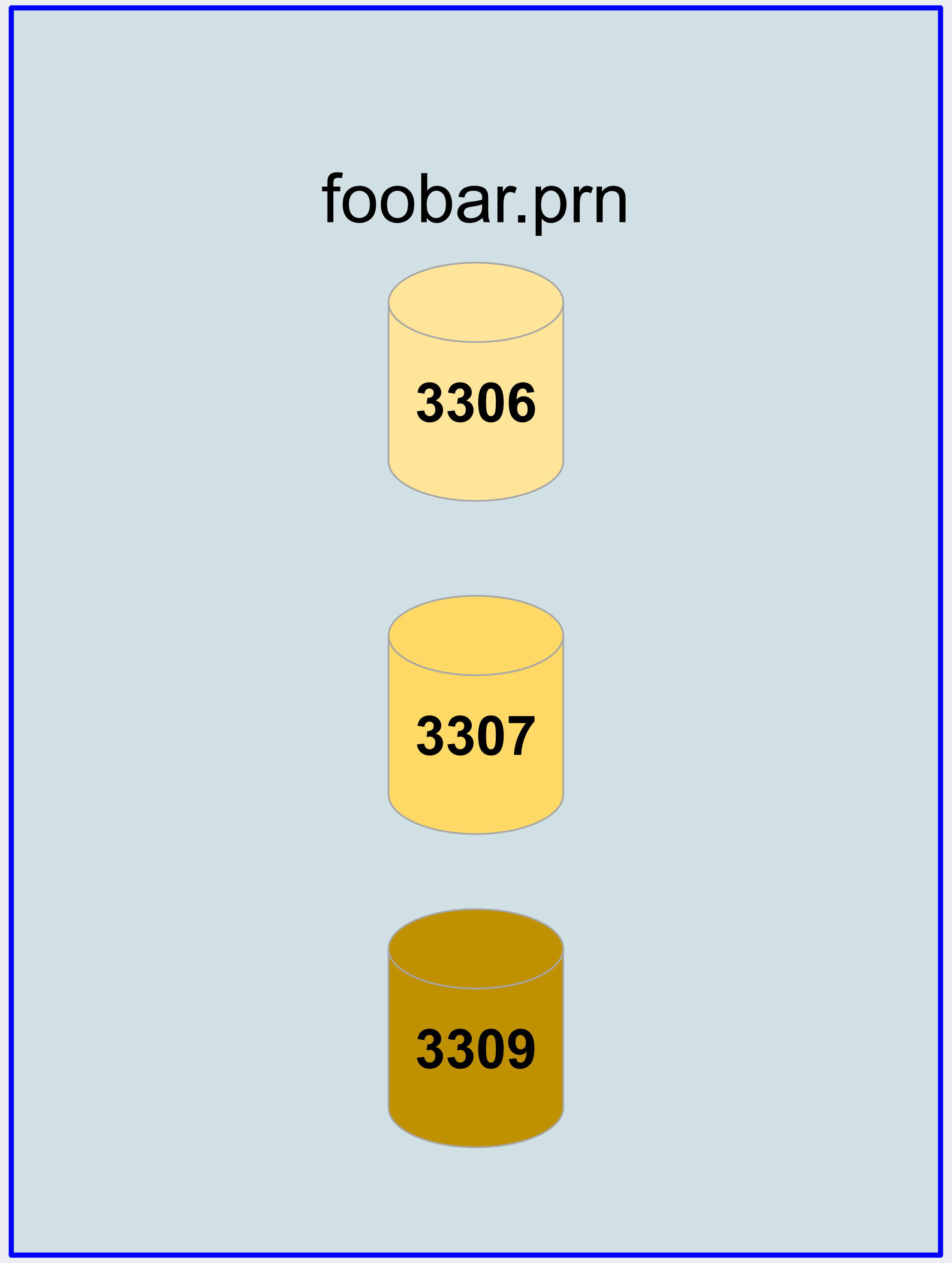
What’s a replicaset ?

Instance

foobar.prn:3307

foobar.prn:3309

foobar.prn:3306



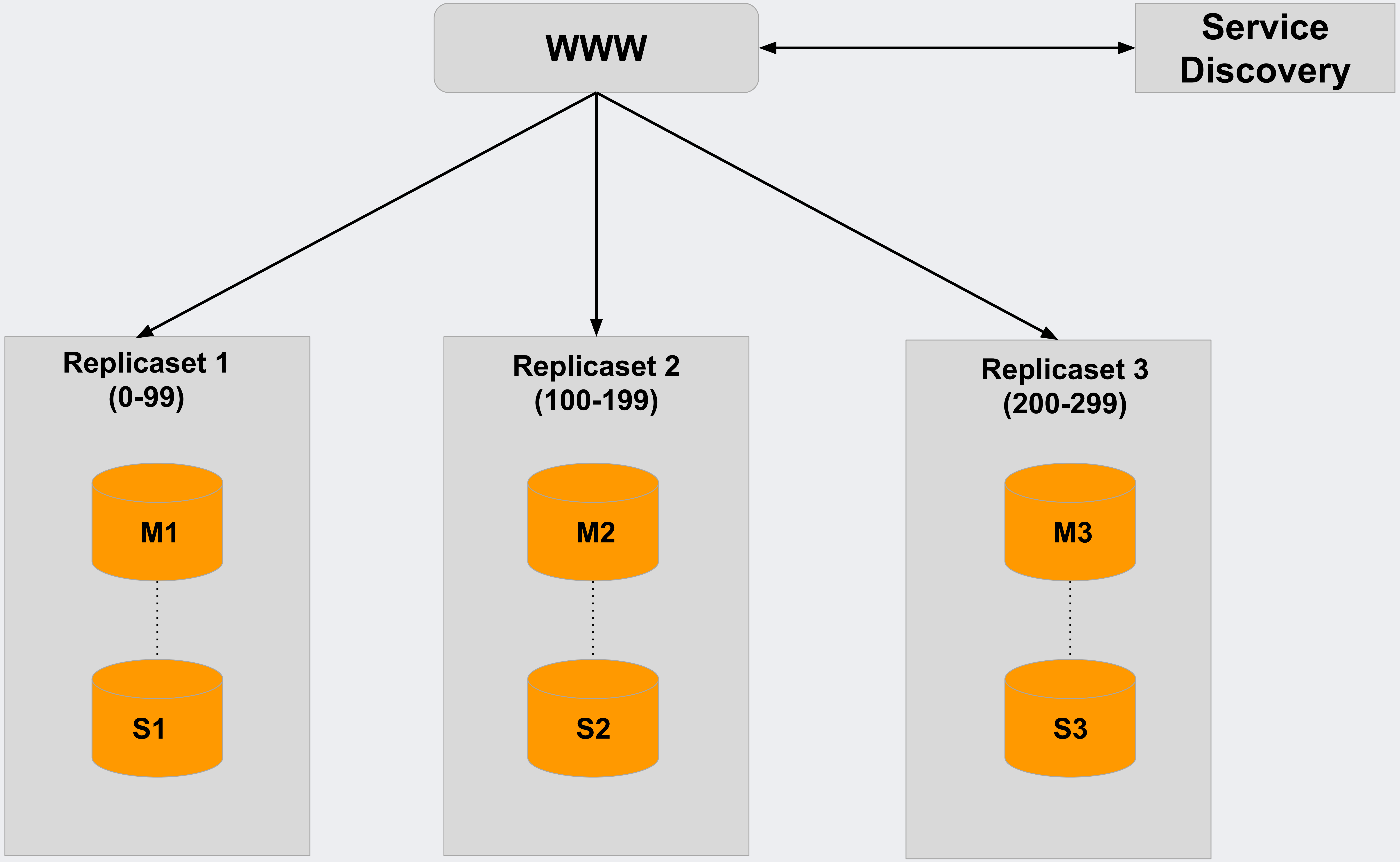
Shard

db.helloworld

db.12345

db.44365

Replicaset

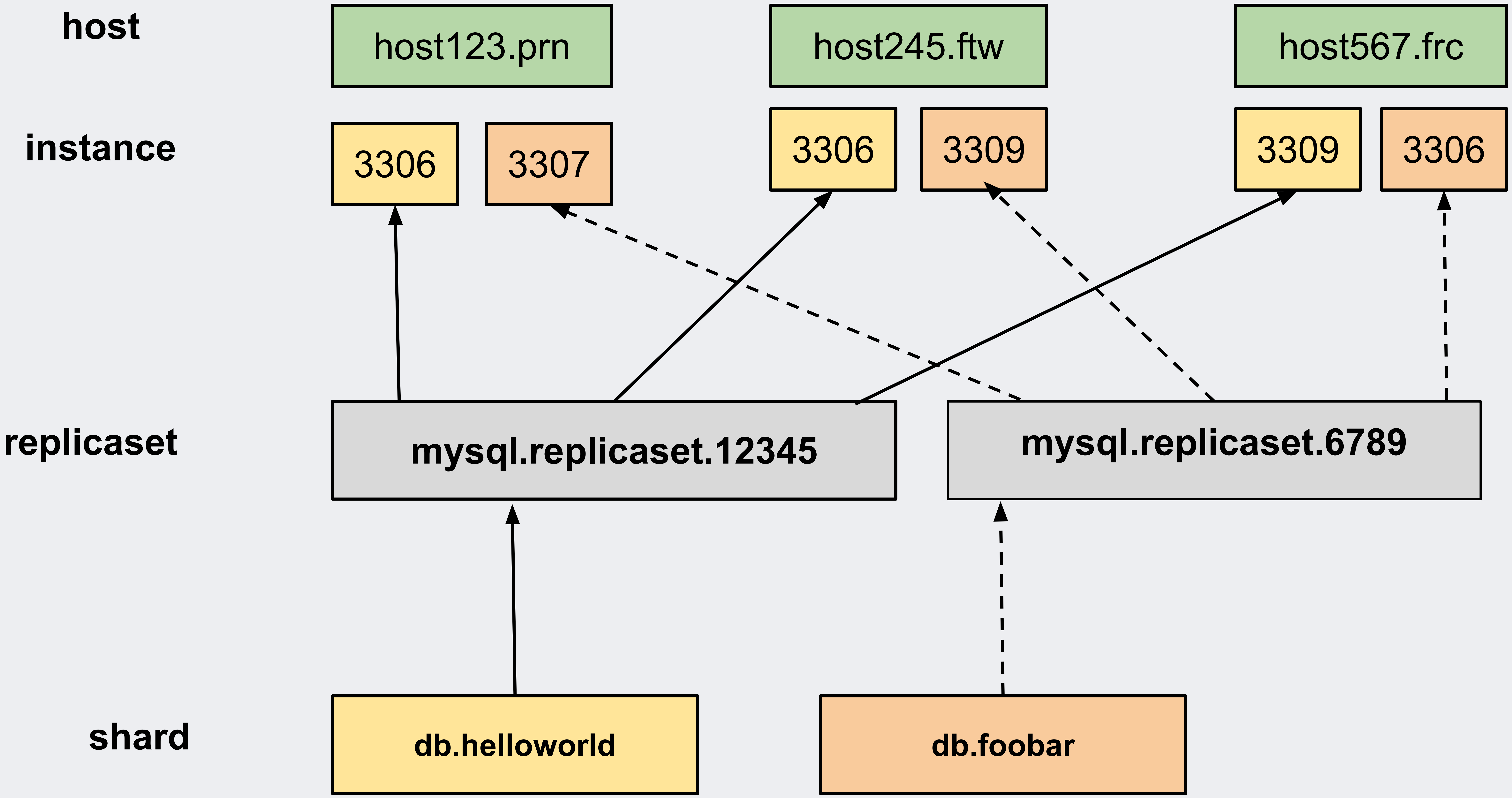

WWW
Service Discovery
Replicaset 1
(0-99)
M1
S1
Replicaset 2
(100-199)
M2
S2
Replicaset 3
(200-299)
M3
S3

Service Discovery

Shard ID	Replicaset	Master	Slave
0-99	Replicaset 1	db1234.prn1:3306	db1234.frc1:3306
100-199	Replicaset 2	db4567.ftw1:3306	db4567.prn1:3307
200-299	Replicaset 3	db1234.atn1:3306	db1234.frc2:3308

Service Discovery

Shard ID	Replicaset	Master	Slave
0-99	Replicaset 1	db1234.prn1:3306	db1234.frc1:3306
100-199	Replicaset 2	db4567.ftw1:3306	db1234.prn1:3309
200-299	Replicaset 3	db1234.frc2:3308	db1234.atn1:3309

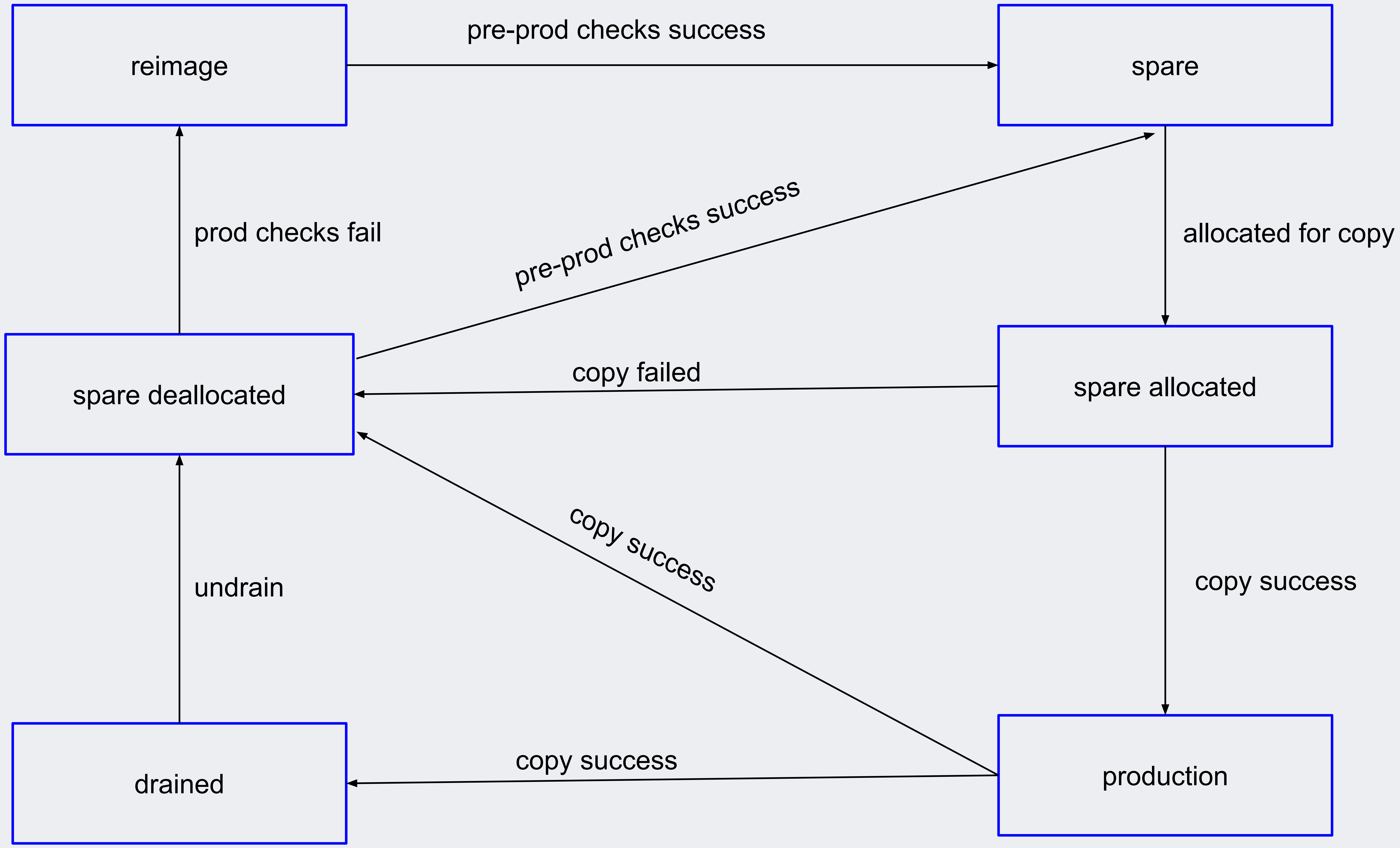

host
host123.prn
host245.ftw
host567.frc
instance
3306
3307
3306
3309
3309
3306
replicaset
mysql.replicaset.12345
mysql.replicaset.6789
shard
db.helloworld
db.foobar

Lifecycle of an instance

Lifecycle of an instance

- States are production, spare, spare allocated, spare deallocated, reimage, drained
- Metadata includes instance properties like name, port, mysql rpm version, state etc
- A mysql shard hosts metadata of all instances in the fleet

Lifecycle of an instance



reimage

pre-prod checks success

spare

prod checks fail

pre-prod checks success

allocated for copy

spare deallocated

copy failed

spare allocated

copy success

undrain

copy success

drained

copy success

production

Lifecycle of an instance

- Each state has its own processor to do the work
- Each state has a queue where work is queued
- Runs constantly scanning the fleet

Instance Migration

Clone an instance

Use case of cloning a production MySQL instance

- Replace a broken instance/host
- Move data around for maintenance
- Balancing host utilization

MPS Copy

A workflow system that manages the requests for cloning MySQL instances

- spare allocation
- set up MySQL config
- copy data
- replication
- validation
- bring it online & remove the old instance if necessary

MPS Copy - Allocation

Choose the best slot for the instance based on its footprint

- Disk usage
- CPU utilization
- Failure domain

Allocation	**Setup**	**Migration**	**Replication**	**Validation**	**Registeration**

MPS Copy - Setup

Turn up an empty instance using the right configuration

- Install the right RPM version
- Bootstrap the correct directory
- Generate the right my.cnf based on its use case
- Make sure the empty instance is connectable

Allocation | Setup | Migration | Replication | Validation | Registeration

MPS Copy - Data Migration

We support three different ways of cloning a production instance

- Physical copy: xtrabackup and myrocks_hotbackup
- Logical copy:
  - mysqldump
  - Restore from backup

Allocation | Setup | Migration | Replication | Validation | Registeration

MPS Copy - Replication

- Setup replication
  - From current production master
  - From Binlog Server
- Catchup

Allocation	Setup	Migration	Replication	Validation	Registeration

MPS Copy - Validation

If the data migration is a logical one, we will use snapshot based checksum to verify the correctness of data by comparing to its current master

Allocation | Setup | Migration | Replication | Validation | Registeration

MPS Copy - Service Registration

Register the new instance in our service discovery system so that the MySQL users will be able to notice this new instance that has been recently turned up

Allocation	Setup	Migration	Replication	Validation	Registeration

Online Shard Migration

Online Shard Migration

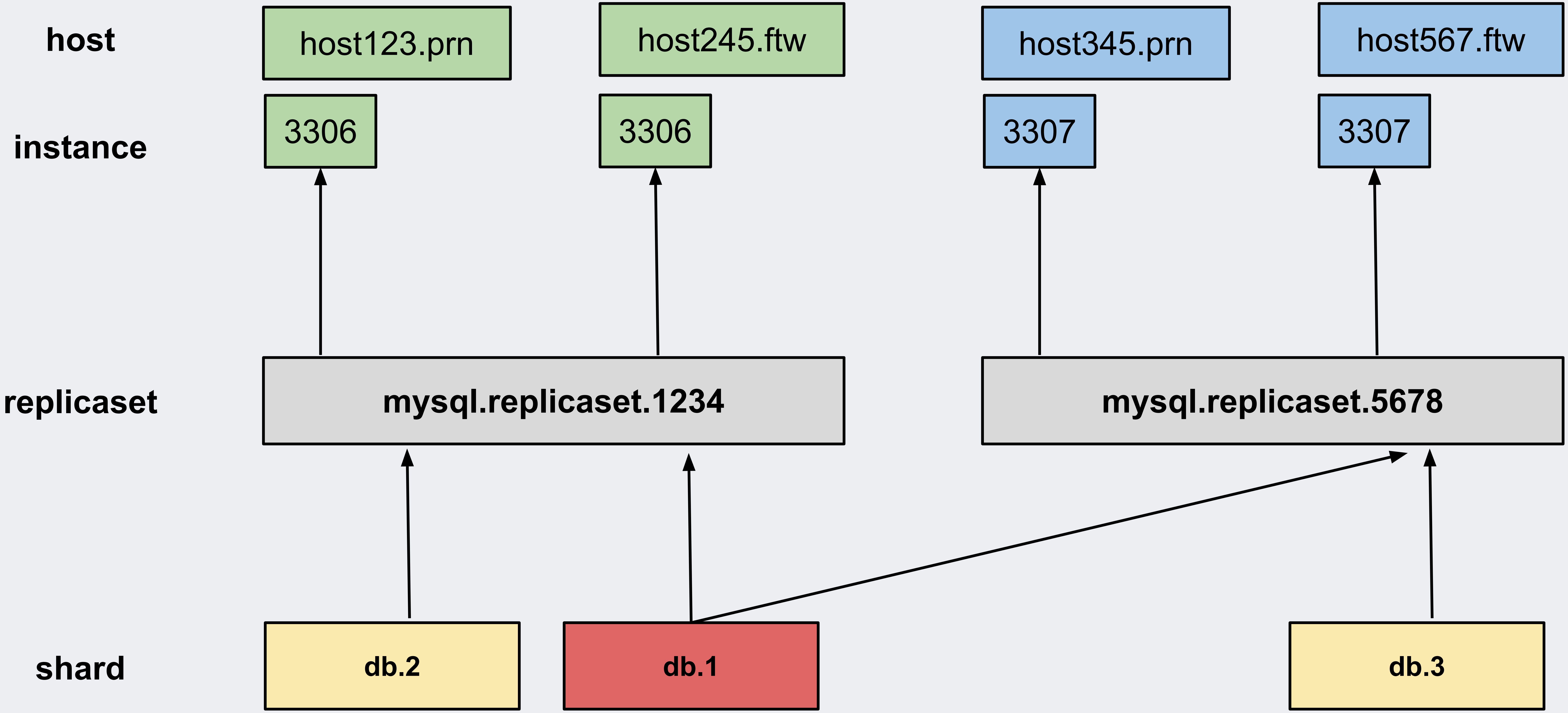
Another fundamental piece of our infra to control the growth of each MySQL instance

- Instance can grow beyond the host level limit
  - Too big
  - Too hot

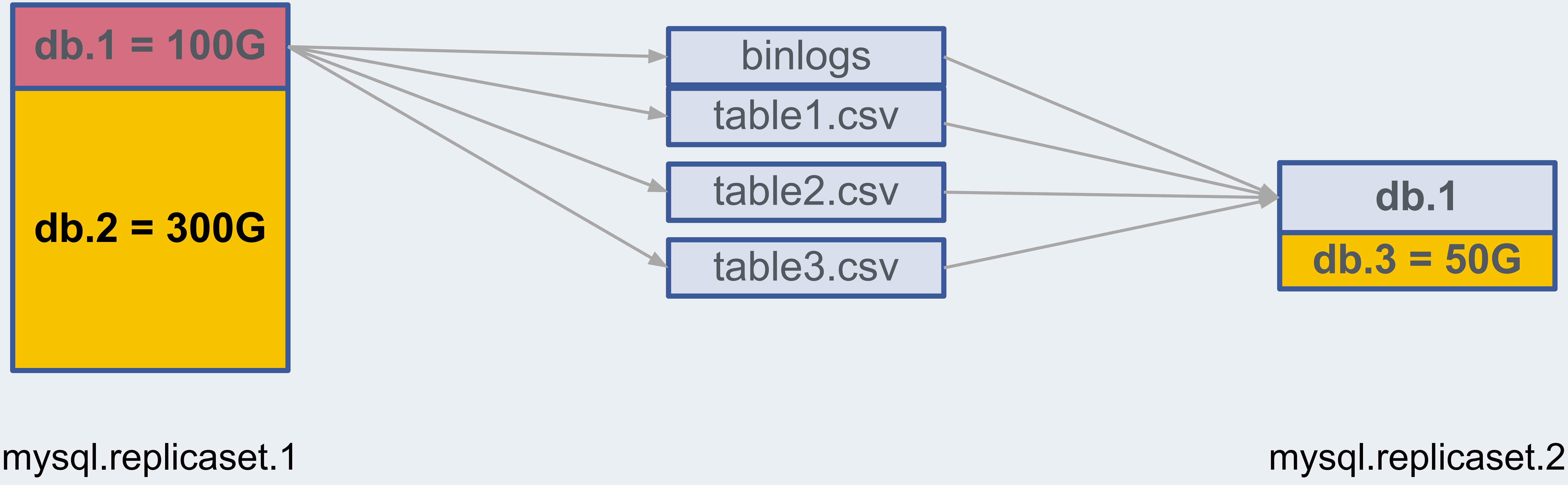
Online Shard Migration (OLM)

Key concept: Move the data of a shard into other smaller/cooler instance through logical migration and register the new address into the service discovery system

OLM


host
host123.prn
host245.ftw
host345.prn
host567.ftw
instance
3306
3306
3307
3307
replicaset
mysql.replicaset.1234
mysql.replicaset.5678
shard
db.2
db.1
db.3

OLM



Shard	Replicaset
db.1	**mysql.replicaset.2**
db.2	mysql.replicaset.1
db.3	mysql.replicaset.3

OLM Processor

Workflow management for massive OLM operations

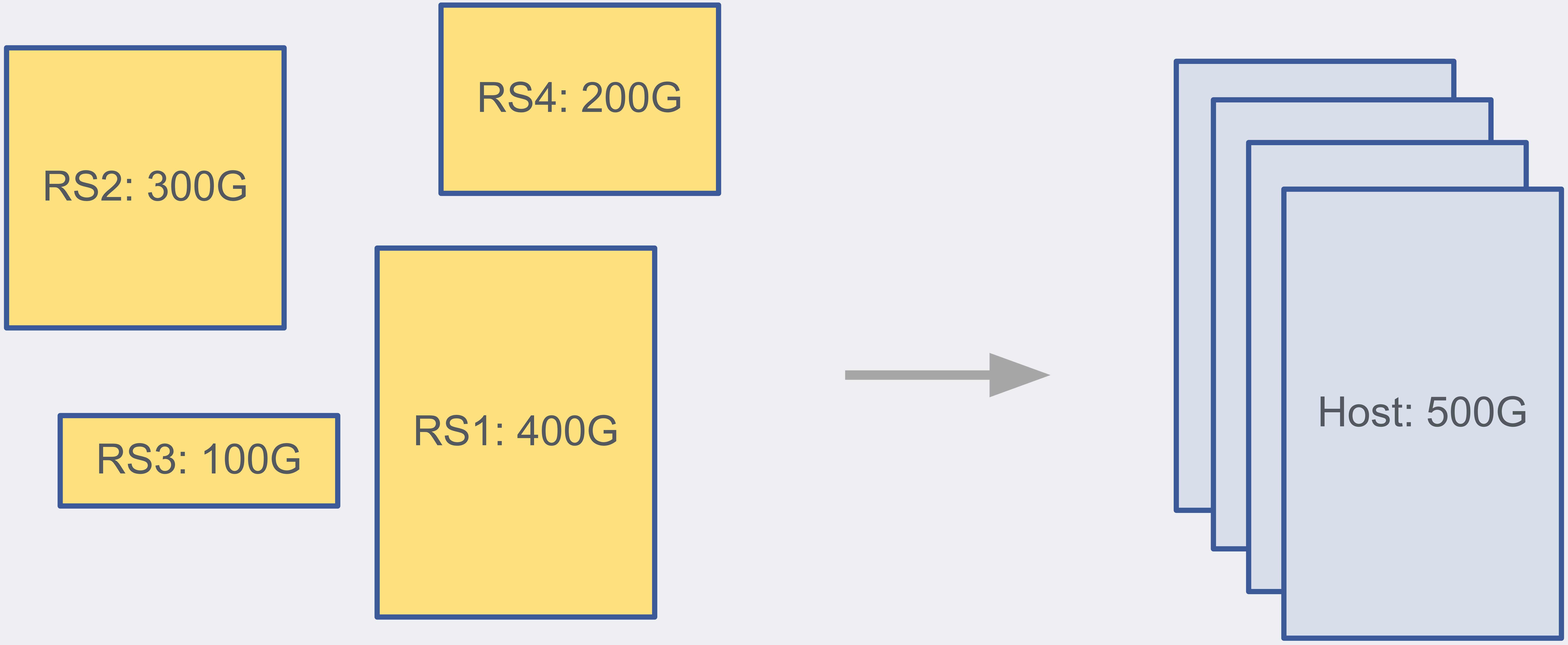
- Conflict solver
- Picking the best destination replicaset
- Kickoff the actual move
- Proper retry and cleanup

Balancing

Balancing

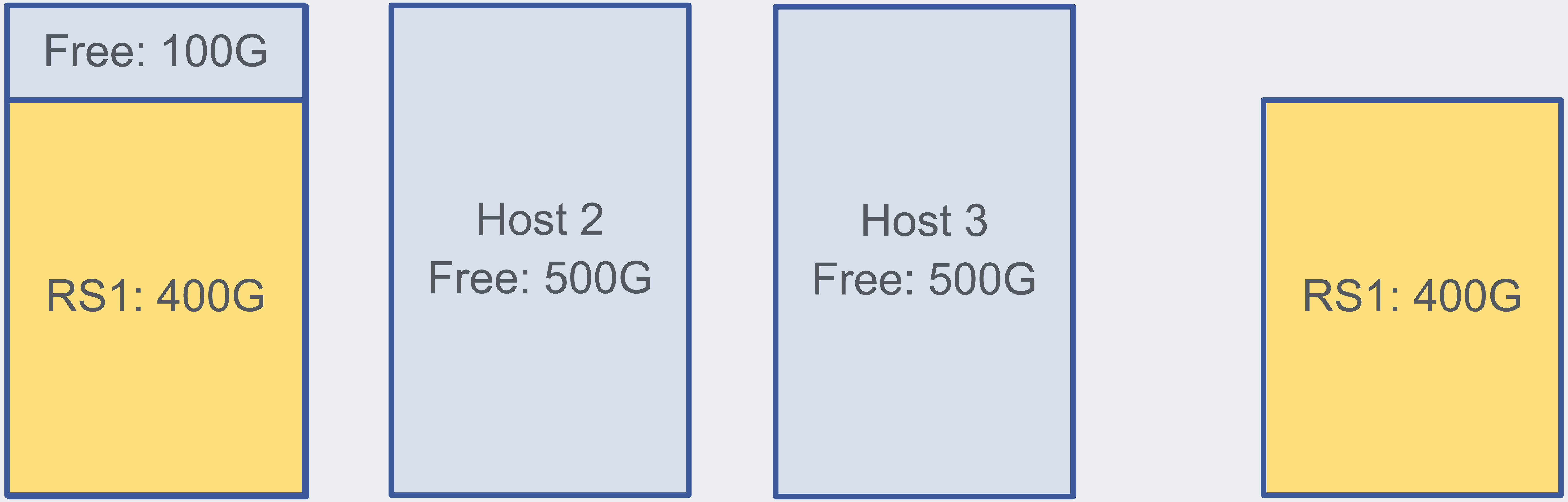
Find the right place for the workload in order to achieve maximum sustainable resource utilization

Poor Stacking

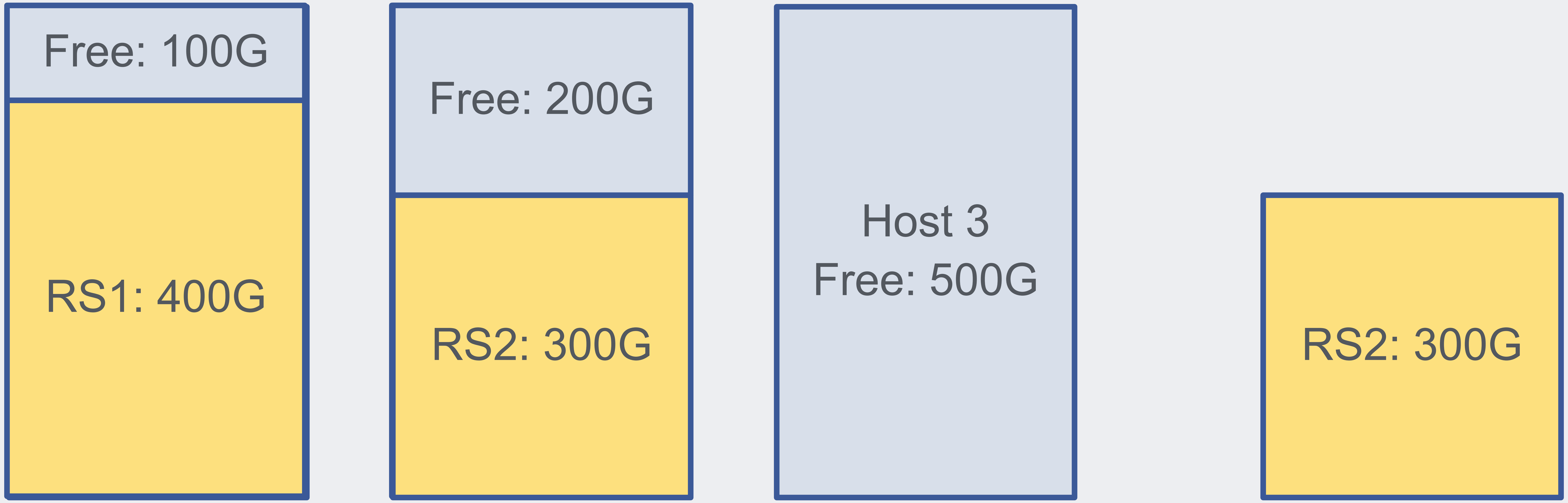


Total: 1000G

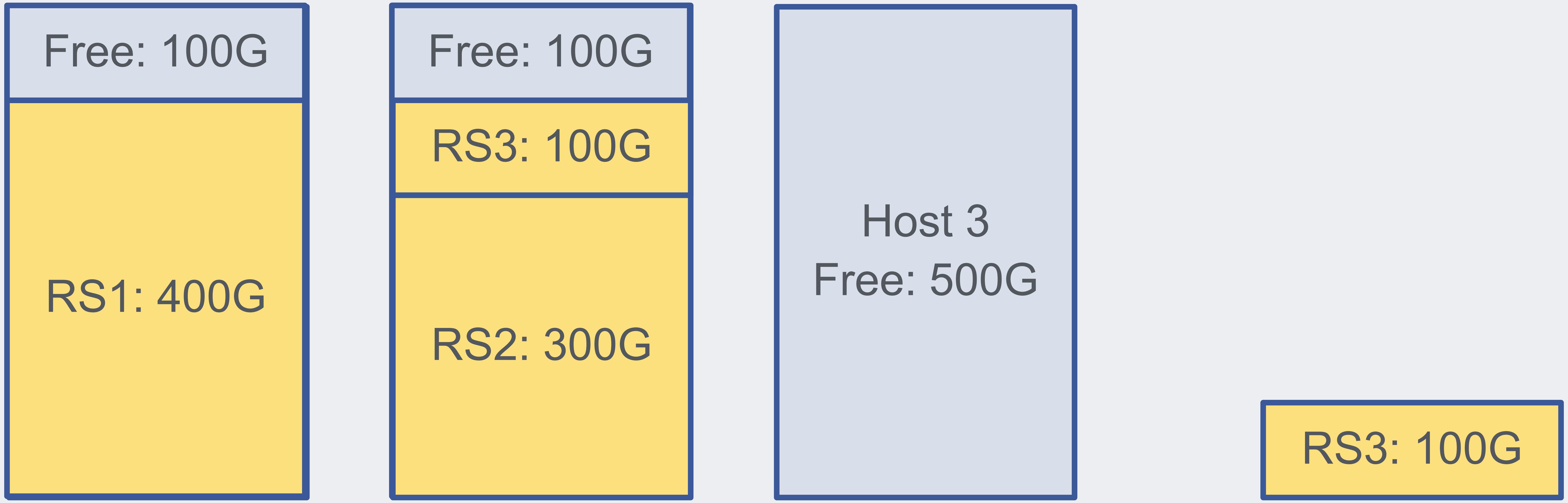
Poor Stacking



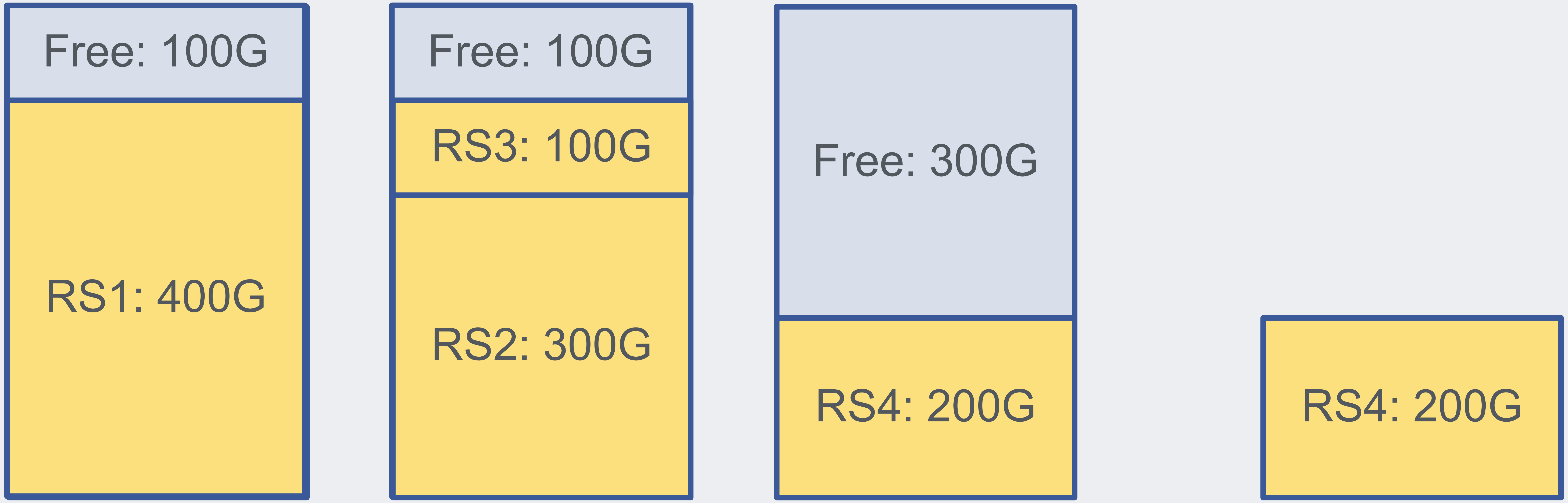
Poor Stacking



Poor Stacking



Poor Stacking



Free: 100G

RS1: 400G

Free: 100G

RS3: 100G

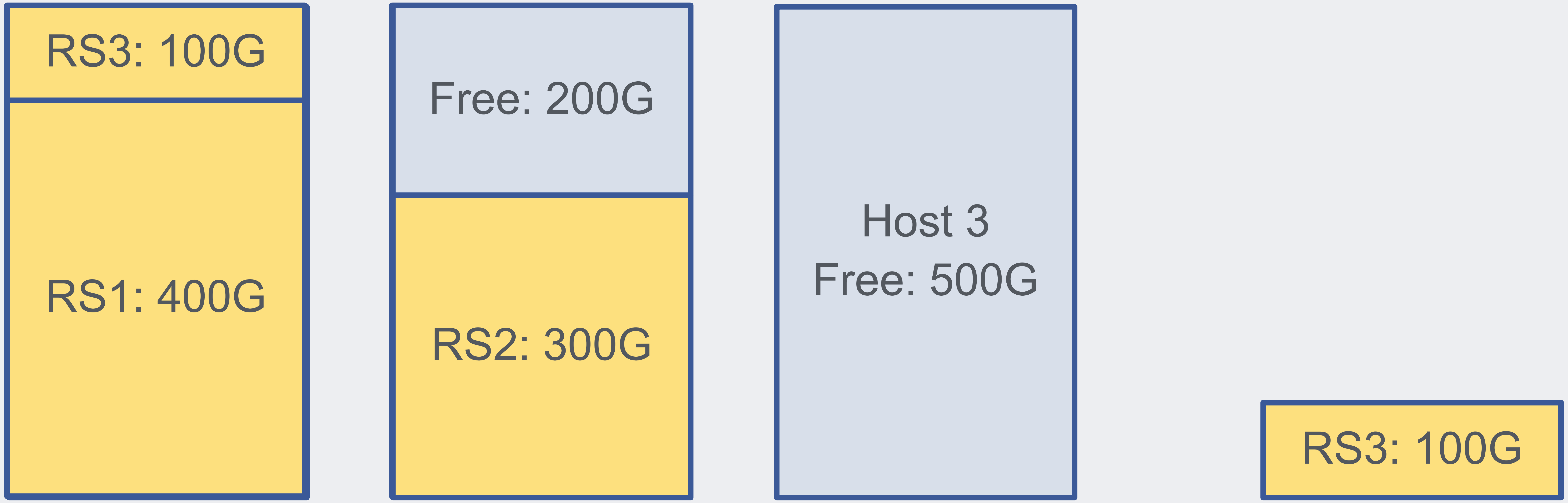
RS2: 300G

Free: 300G

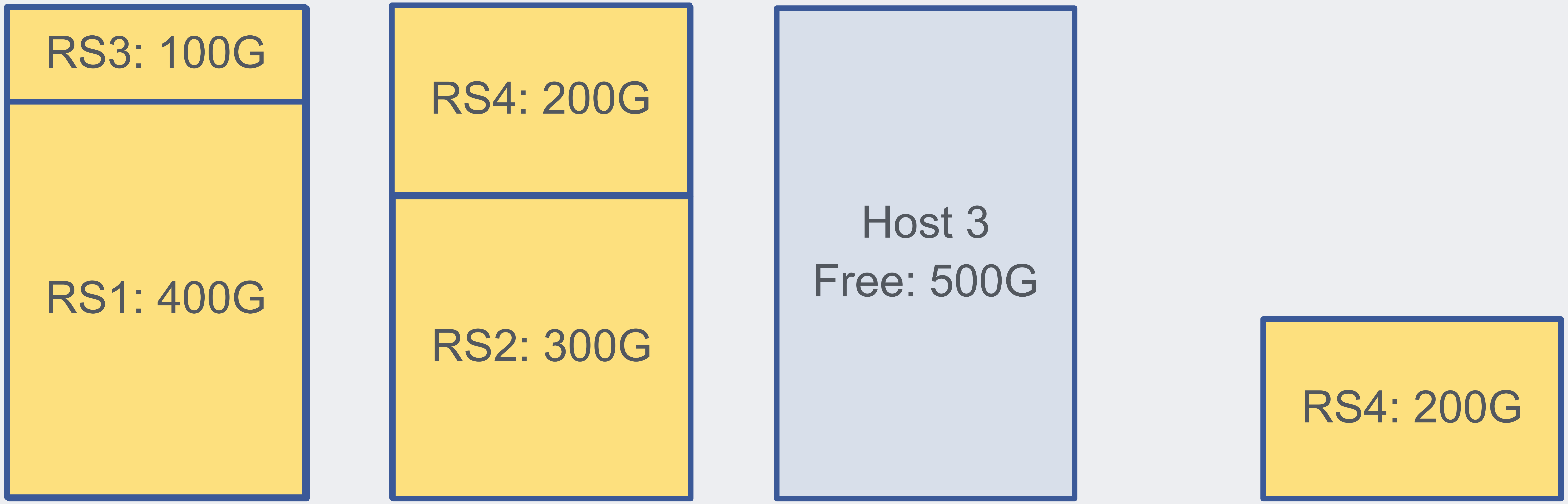
RS4: 200G

RS4: 200G

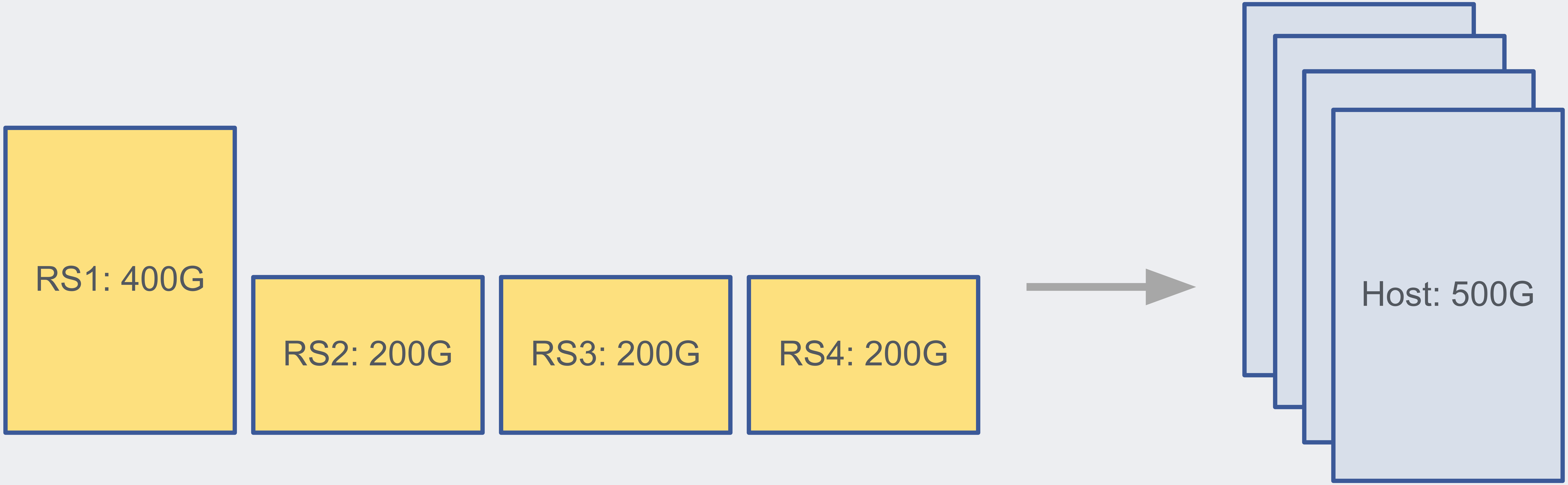
Proper Stacking



Proper Stacking



Carve the Shape



RS1: 400G

RS2: 200G

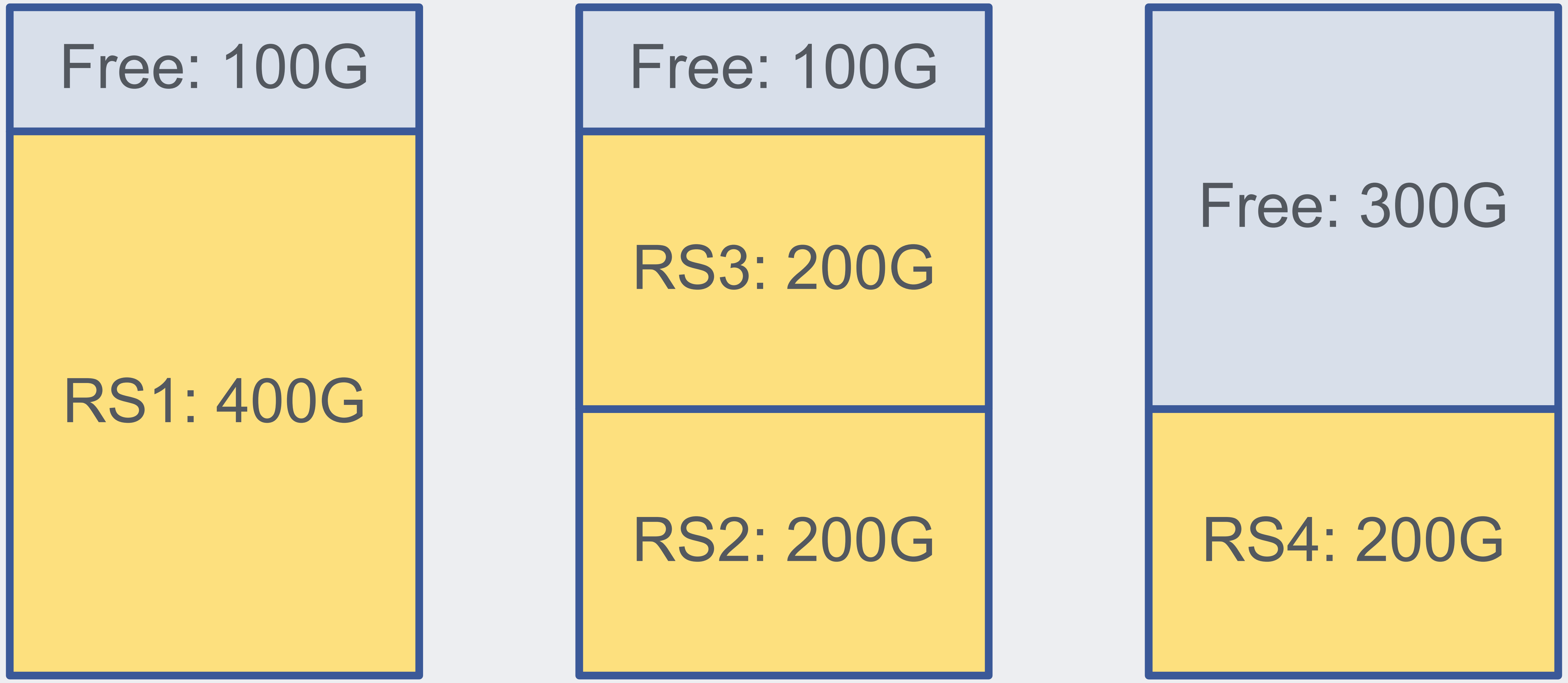
RS3: 200G

RS4: 200G

Host: 500G

Total: 1000G

Poor Shape



Free: 100G

RS1: 400G

Free: 100G

RS3: 200G

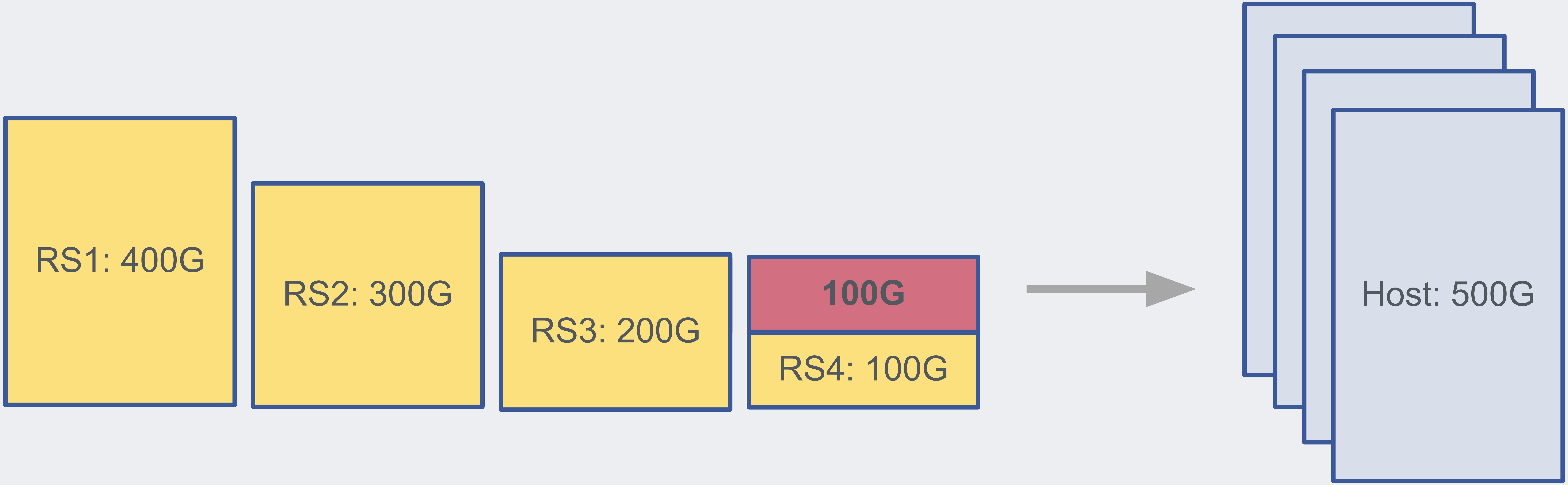
RS2: 200G

Free: 300G

RS4: 200G

X3

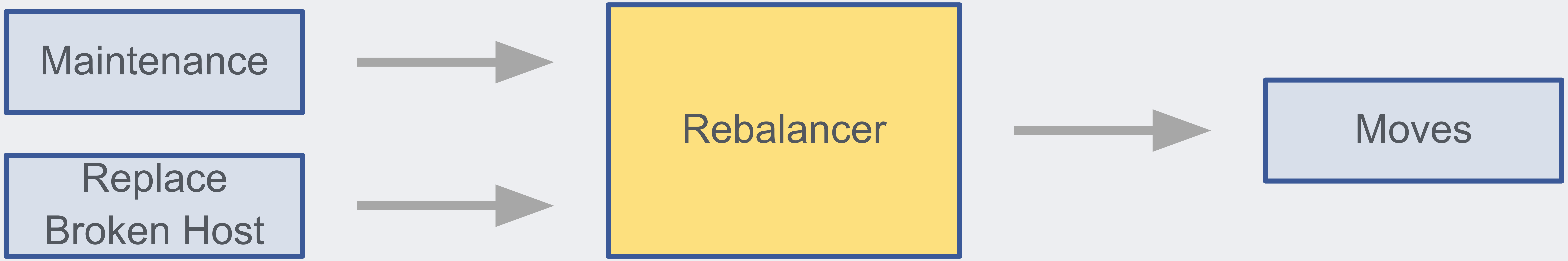
Carve the Shape



Total: 1000G

Rebalancer

Goal: Find the best slot for hosting the given workload profile and reduce the imbalance score across the fleet to be minimum



Rebalancer - Challenges

Multiple balancing factors

- CPU/Memory/Disk usage
- Fault domain spreading
- MySQL vs LBU anti affinity

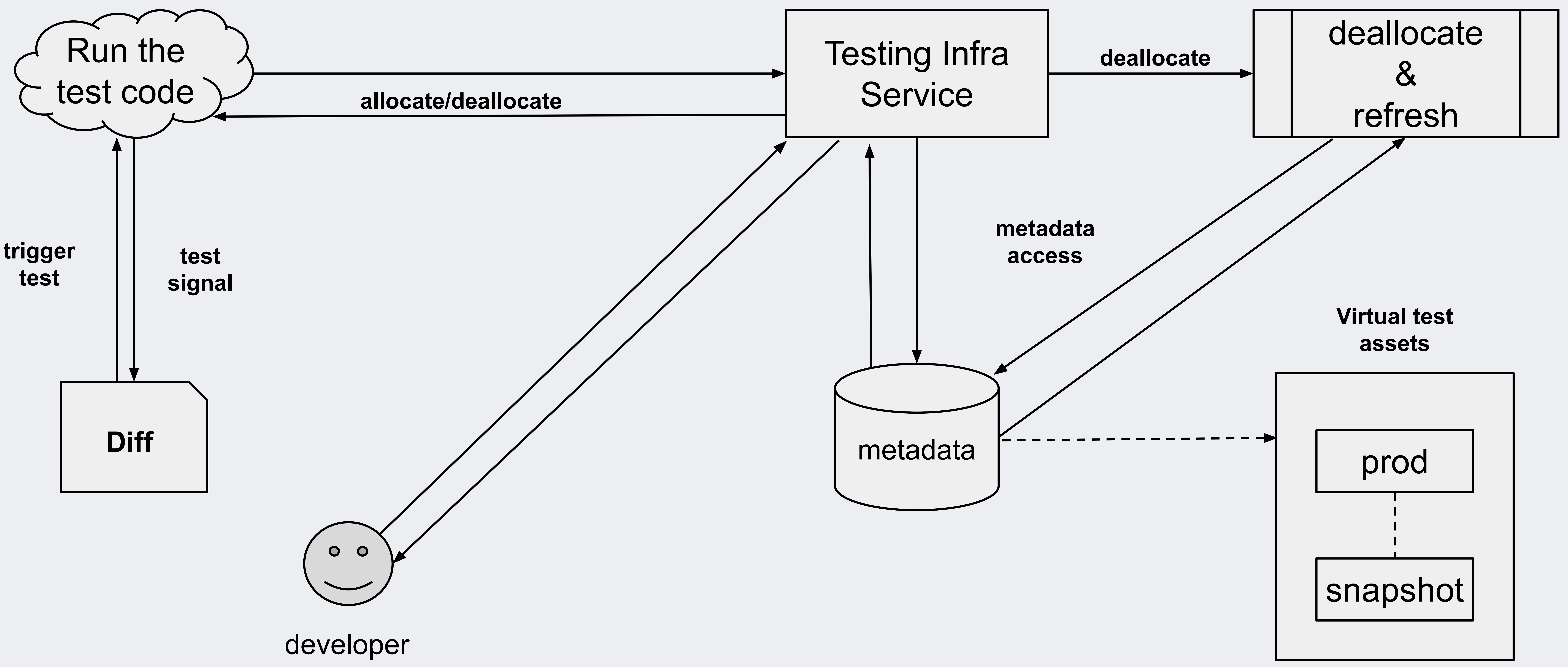
Testing Infrastructure

Testing Infra for Automations

- Lots of Automation code handling critical components of Infra
- UnitTests are good but mock backend connection
- Need to test end to end

Testing Infra Goals

- build and canary packages based on the change
- provide signals at diff time for developer
- production like setup, but isolated environment
- iterate quickly with confidence


Run the test code
allocate/deallocate
Testing Infra Service
deallocate
deallocate & refresh
trigger test
test signal
Diff
metadata access
Virtual test assets
metadata
prod
snapshot
developer

Q&A

facebook