HPC placement group policies:

requirements, specifications, and comparisons

Asser Tantawi

IBM TJ Watson Research Center

Proposed

Policies for HPC placement groups

- Single level (Basic)
  - Affinity/Anti-affinity at one level (server, rack, zone)
- Multiple levels
  - Affinity/Anti-affinity at multiple levels
  - May be opposite, e.g., spread at one level, but pack on another
- Constrained
  - Affinity/Anti-affinity at multiple levels with constraints
  - Number allocated at a level
    - Spread, but not too thin (min)
    - Pack, but not too thick (max)
    - same (fixed)
  - Type of constraint
    - Soft/Hard

Single

```
kind: GroupPlacement
spec:
  group:
    name: MyApp
    size: 20
    type: bx2-16x64
  constraints:
    - level: rack
      affinity: spread
```

```
kind: GroupPlacement
spec:
  group:
    name: MyApp
    size: 20
    type: bx2-16x64
  constraints:
    - level: server
      affinity: pack
```

Multiple

```
kind: GroupPlacement
spec:
  group:
    name: MyApp
    size: 20
    type: bx2-16x64
  constraints:
    - level: rack
      affinity: spread
    - level: server
      affinity: pack
```

Constrained

```
kind: GroupPlacement
spec:
  group:
    name: MyApp
    size: 24
    type: bx2-16x64
  constraints:
    - level: rack
      affinity: spread
      soft: true
      min: 4
    - level: server
      affinity: pack
      max: 2
```

Open issues

- Group of groups
- Constraint templates
- Constraints other than fixed and range
  - preferred value
  - one of a set of values
  - relationships
- Network topology
  - not necessarily overlaying containment hierarchy
  - heterogeneous

Comparisons

Problem:

P1

Infrastructure:

- 3 zones
- 8 racks per zone
- 20 servers per rack

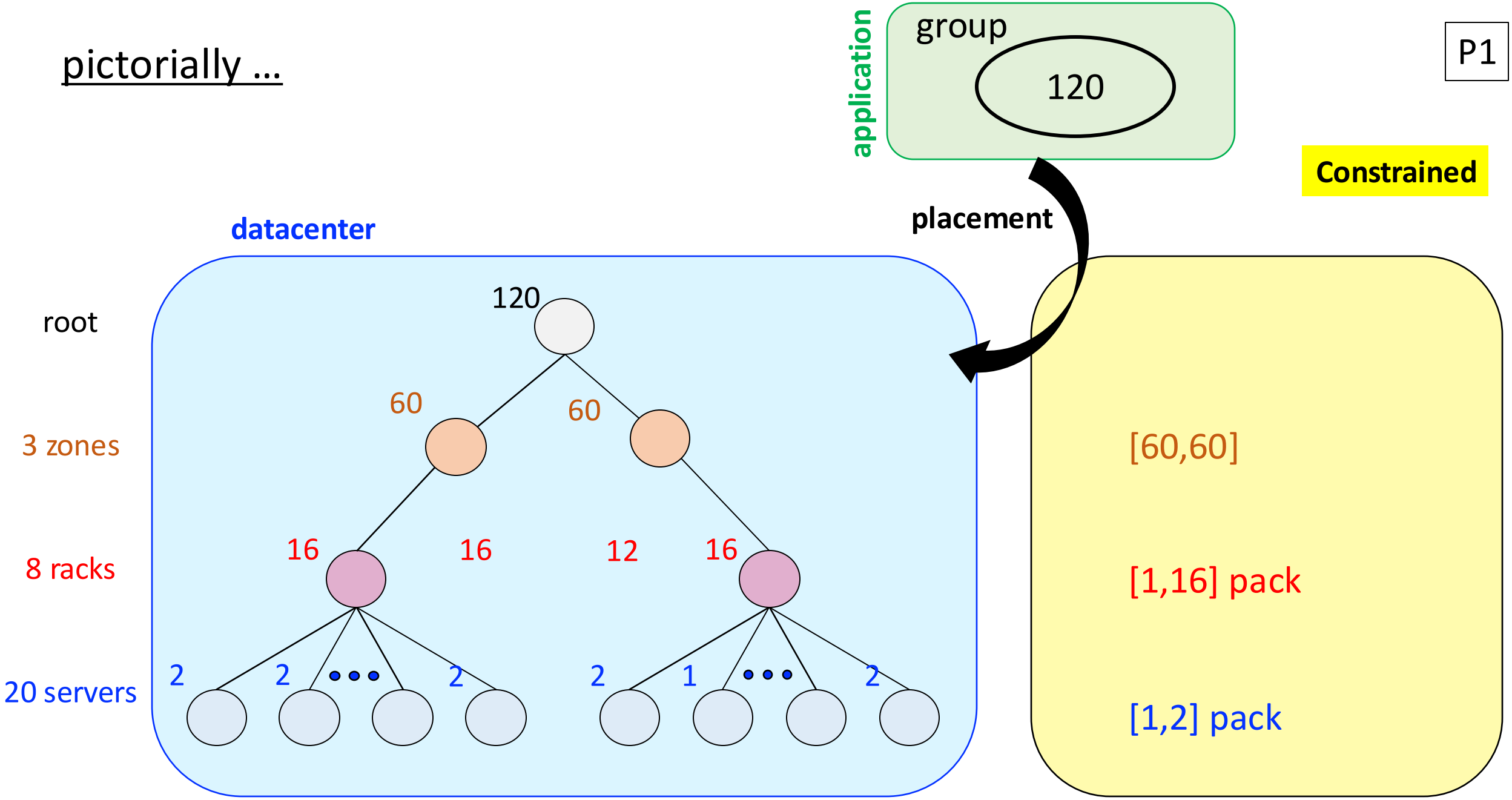
Placement group:

- size 120

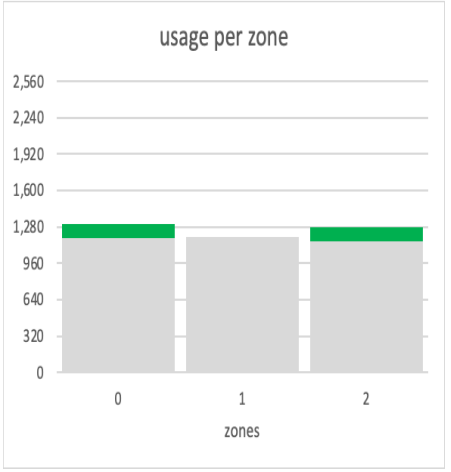
Goal:

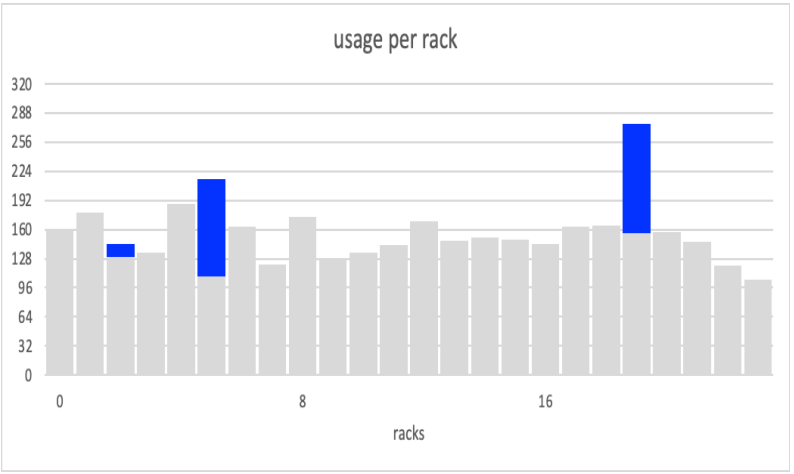
- placed in 2 zones, 60 in each
- rack affinity, but no more than 16 per rack
- server affinity, but no more than 2 per server

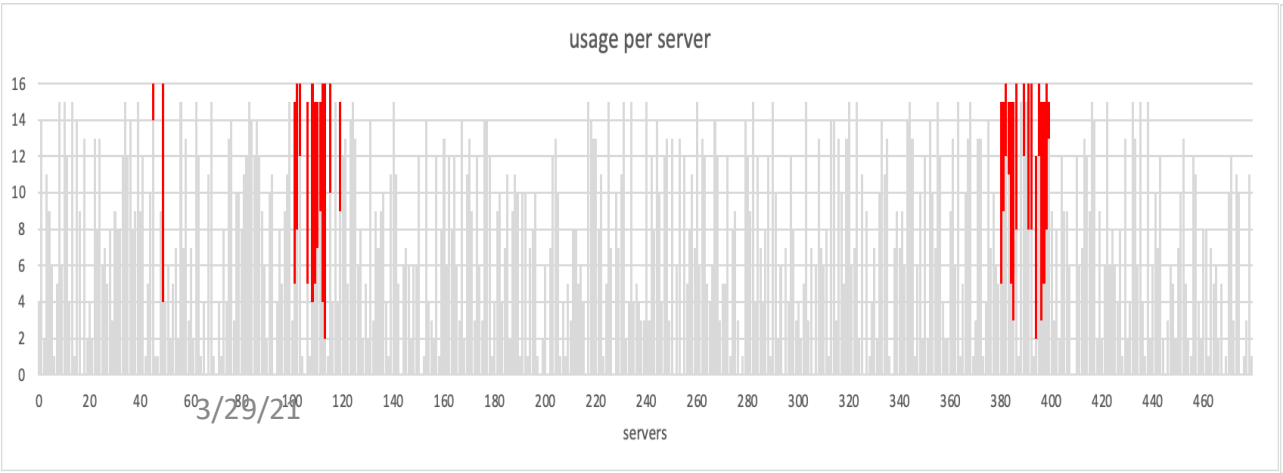
pictorially …



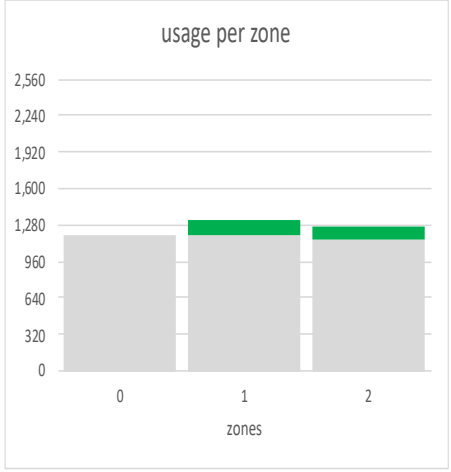
Multiple

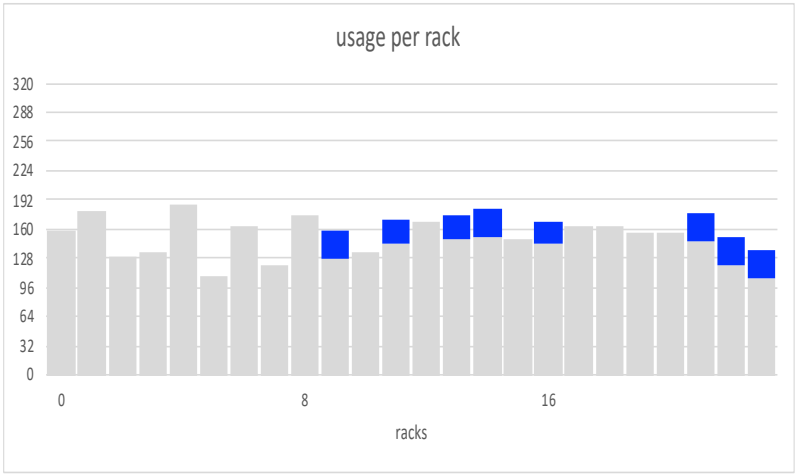


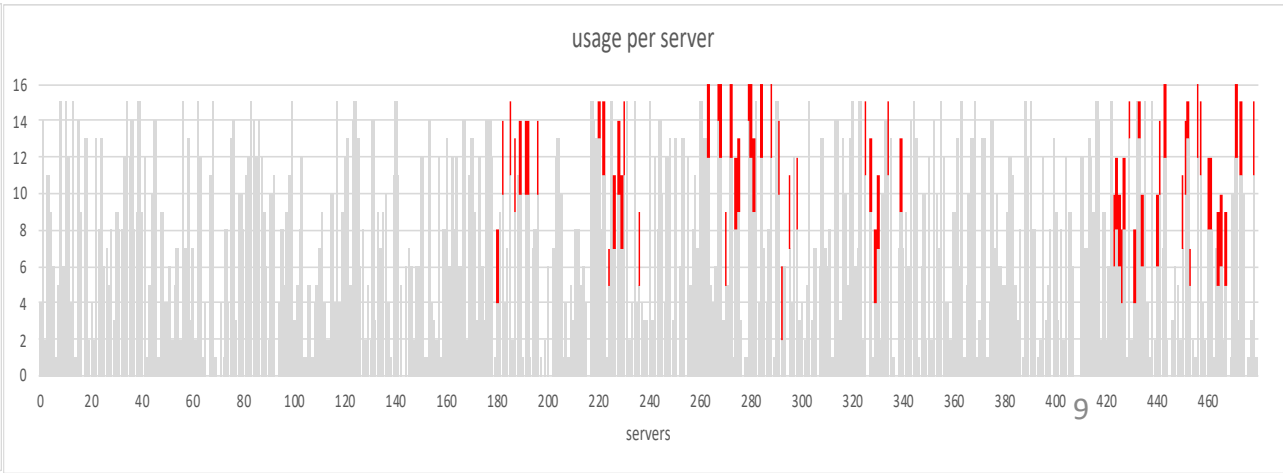




Constrained

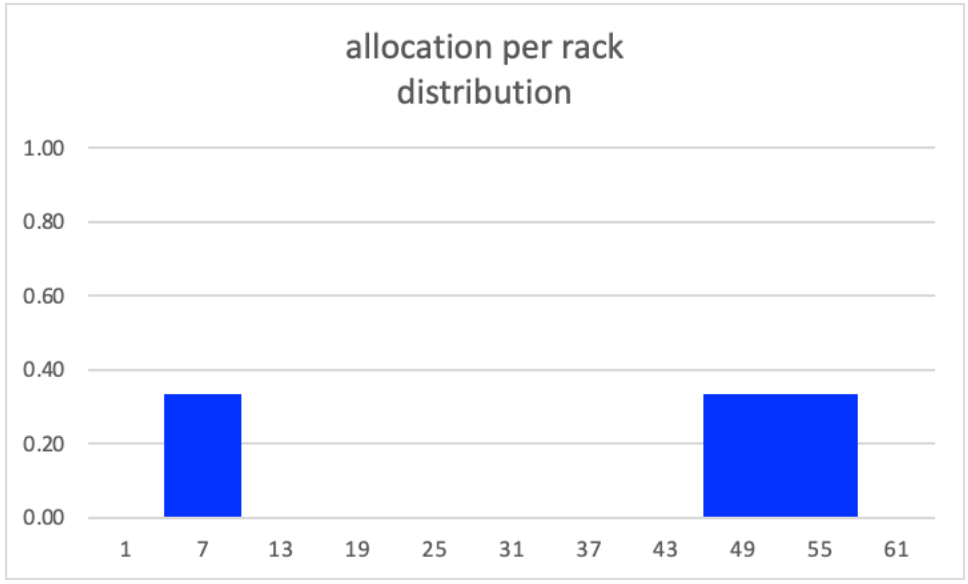


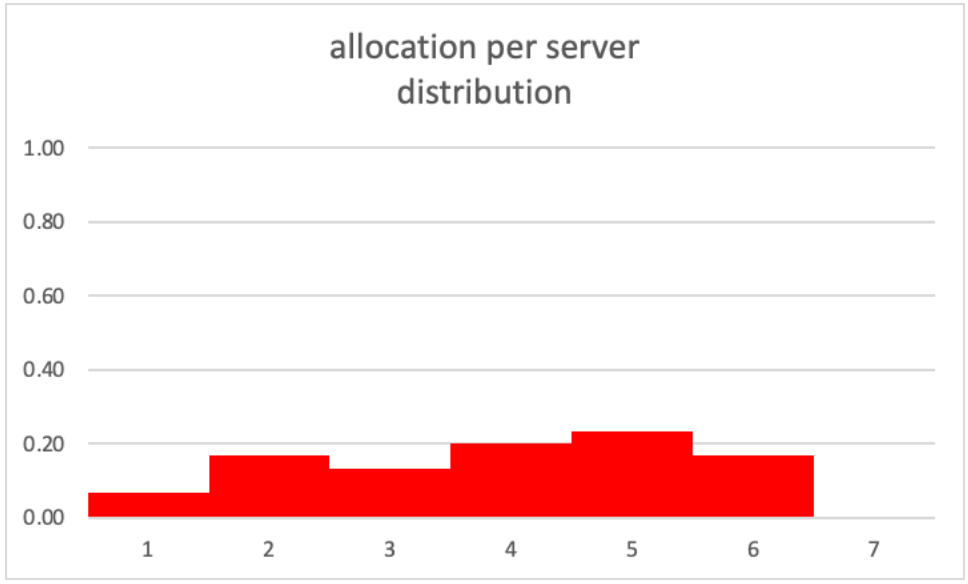




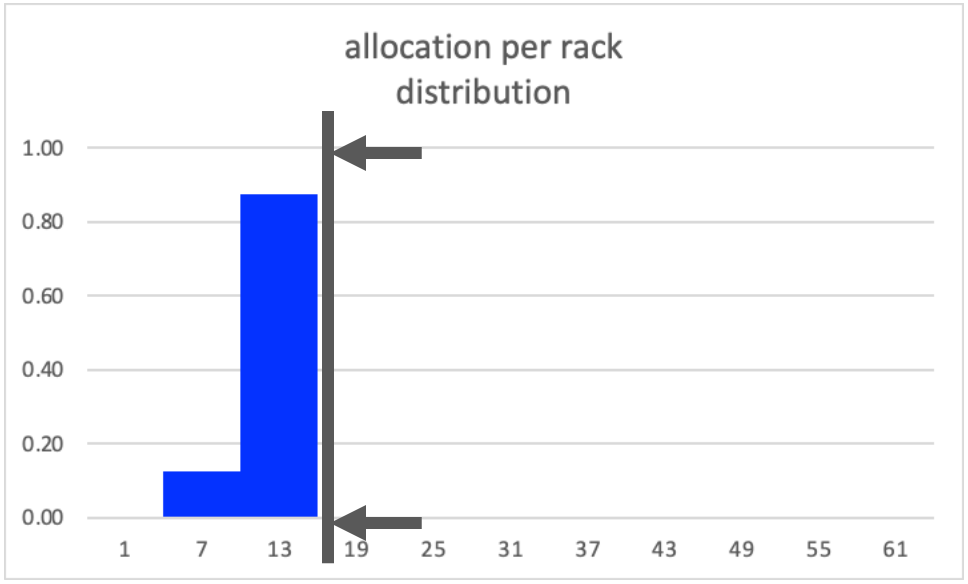
P1

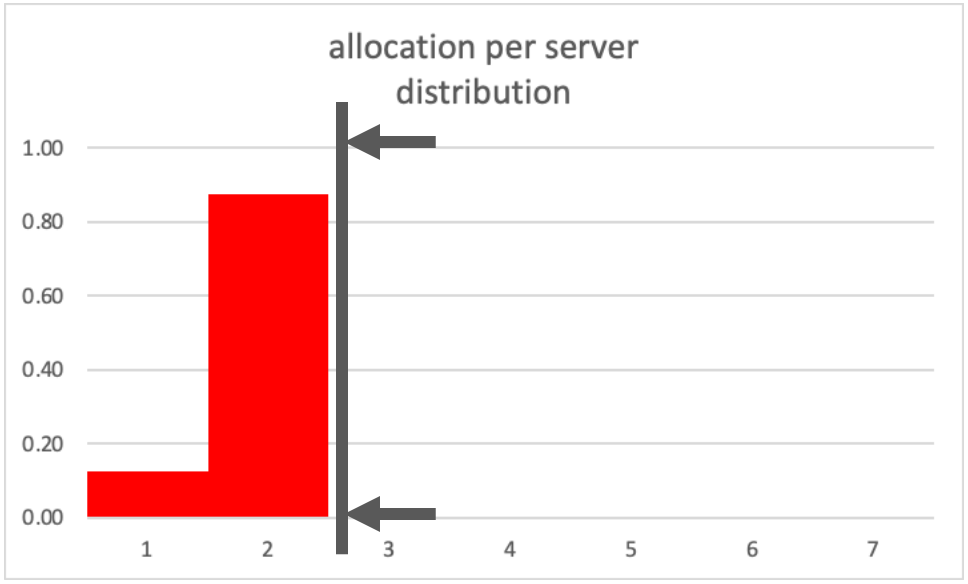
Multiple





Constrained





Problem:

P2

Infrastructure:

3 zones

8 racks per zone

20 servers per rack

Placement group:

size 120

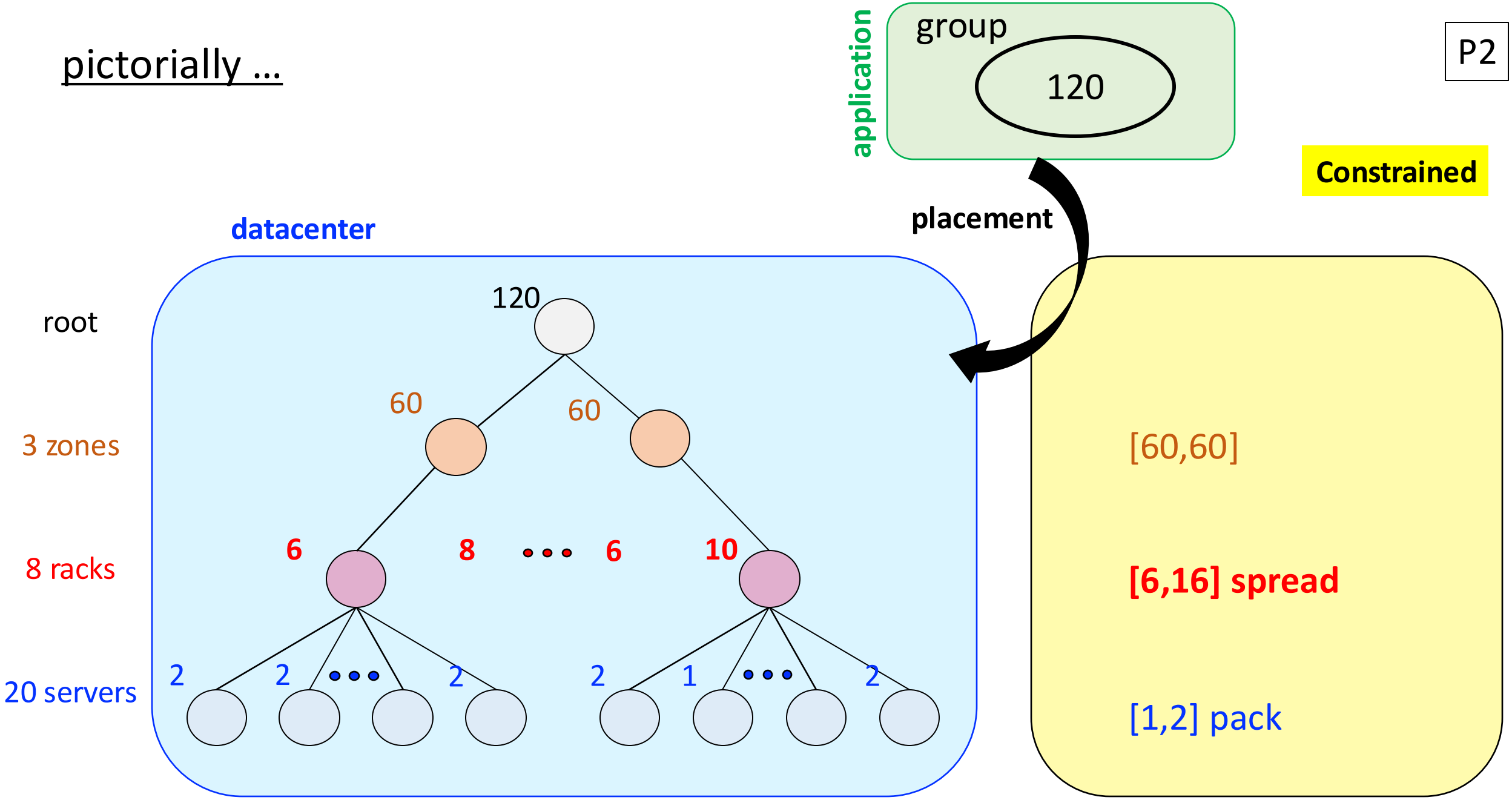
Goal:

placed in 2 zones, 60 in each

rack **anti-affinity**, but **no less than 6** per rack

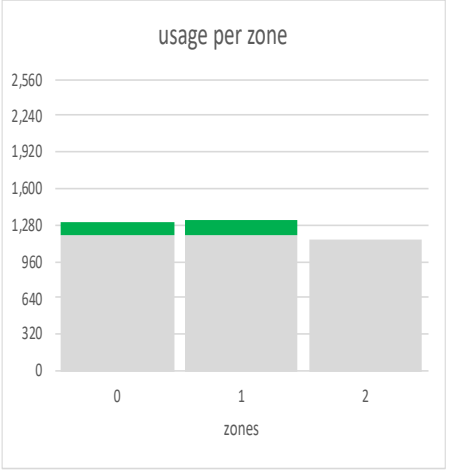
server affinity, but no more than 2 per server

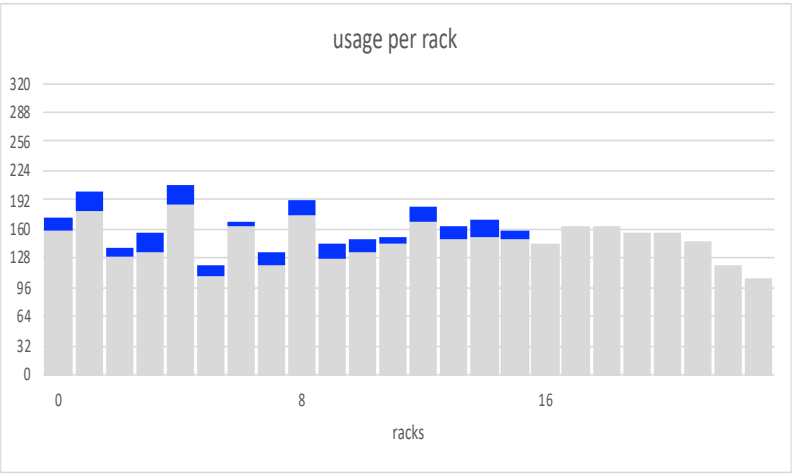
pictorially …

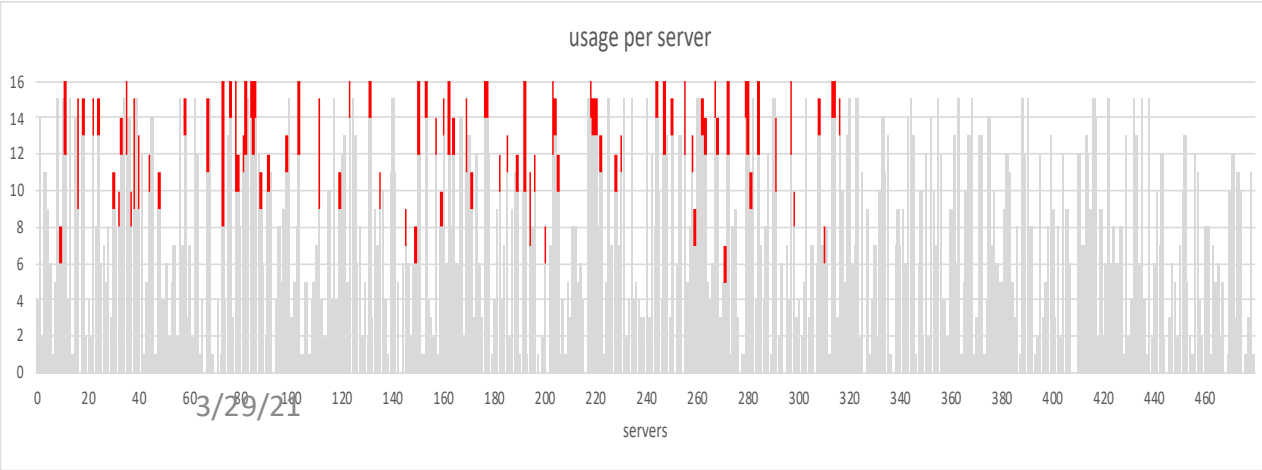


P2

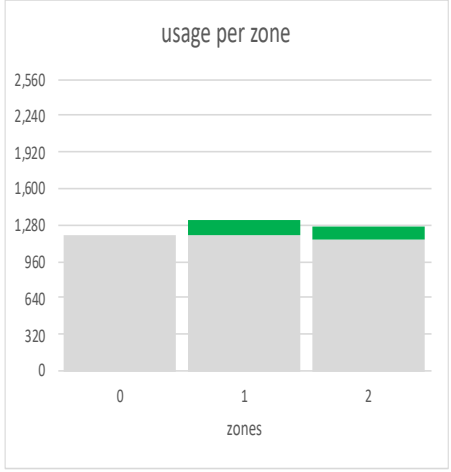
Multiple

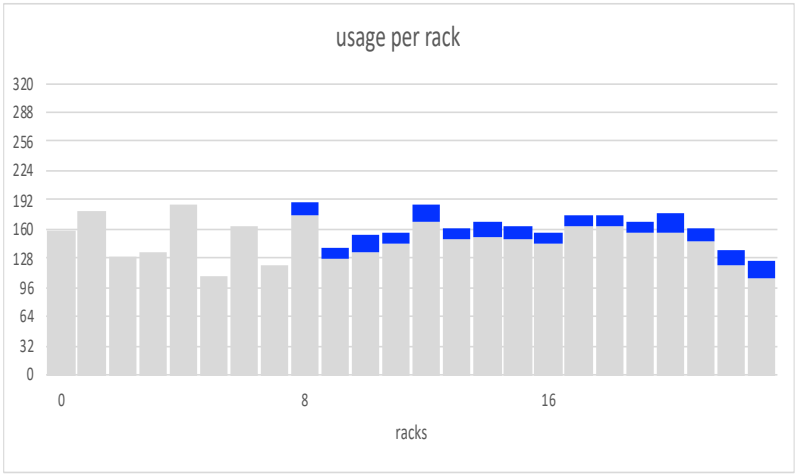


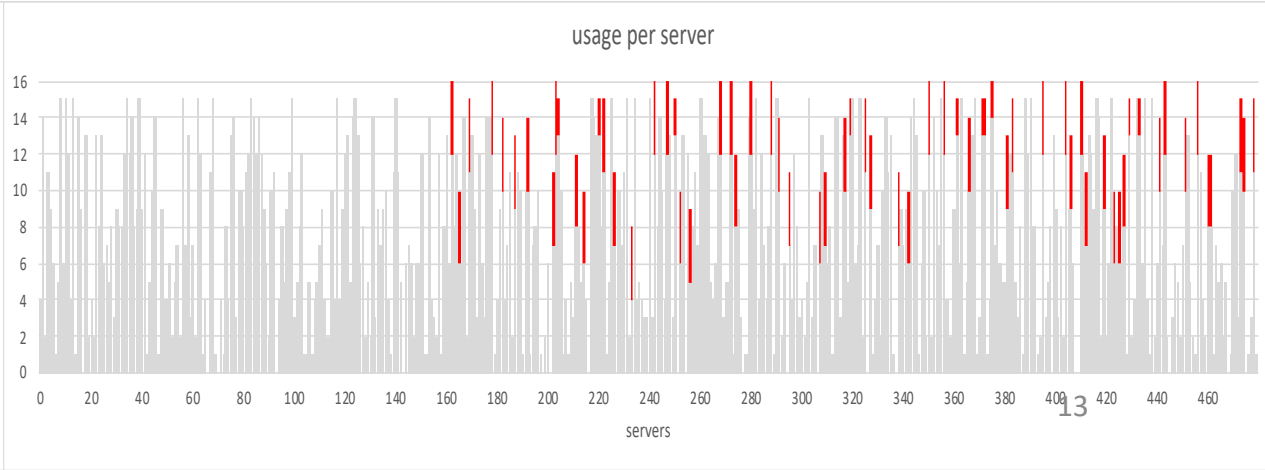




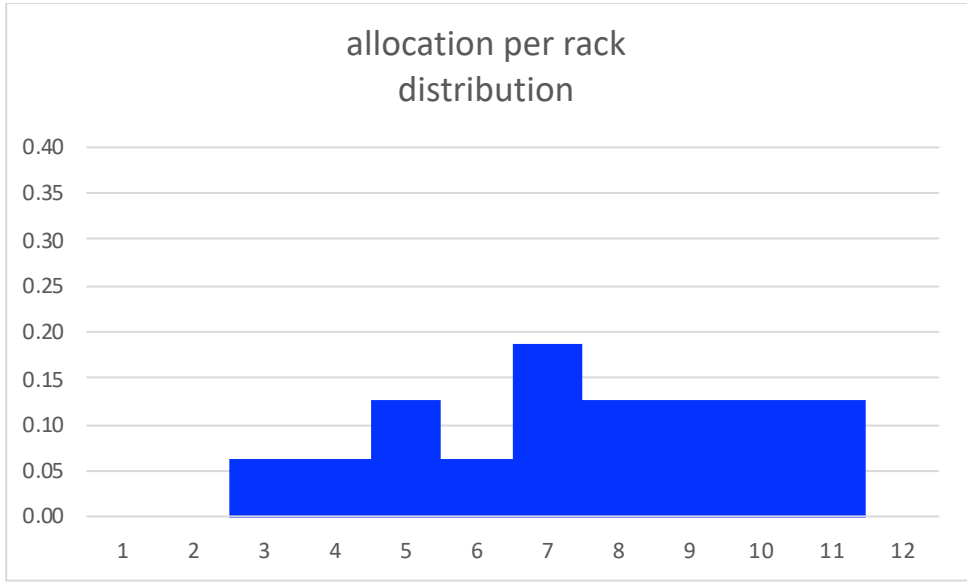
Constrained

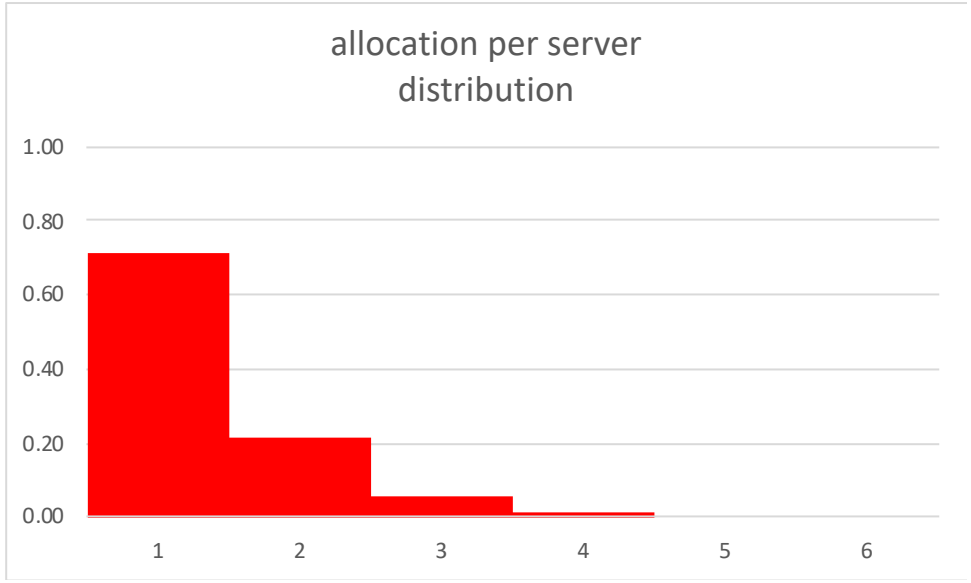




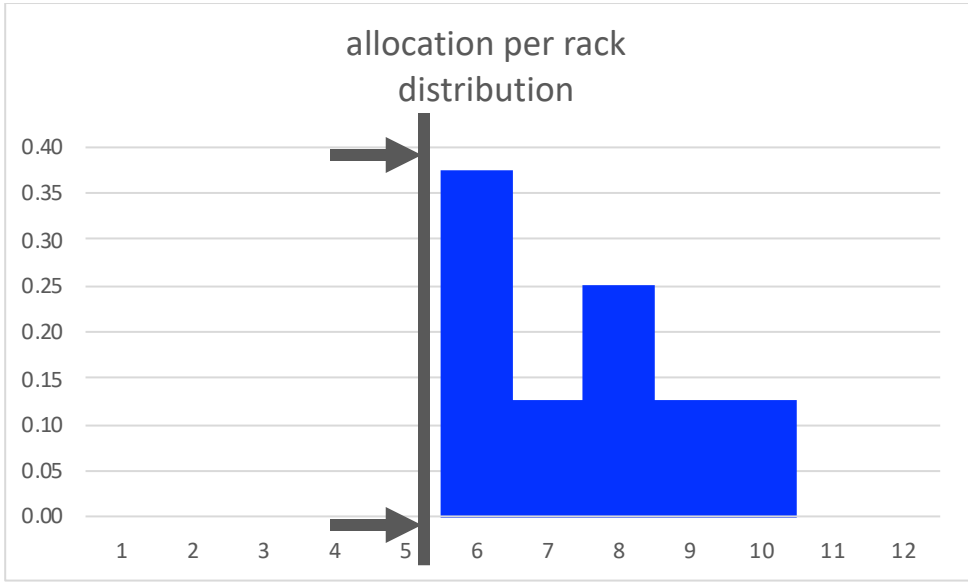


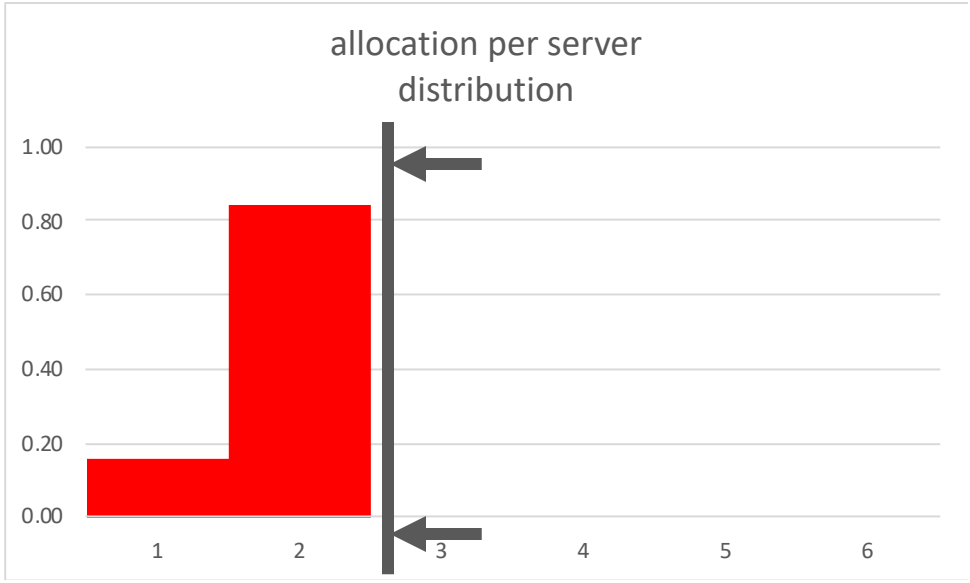
Multiple





Constrained





P2

Problem:

P3

Infrastructure:

3 zones

8 racks per zone

20 servers per rack

Placement group:

size 120

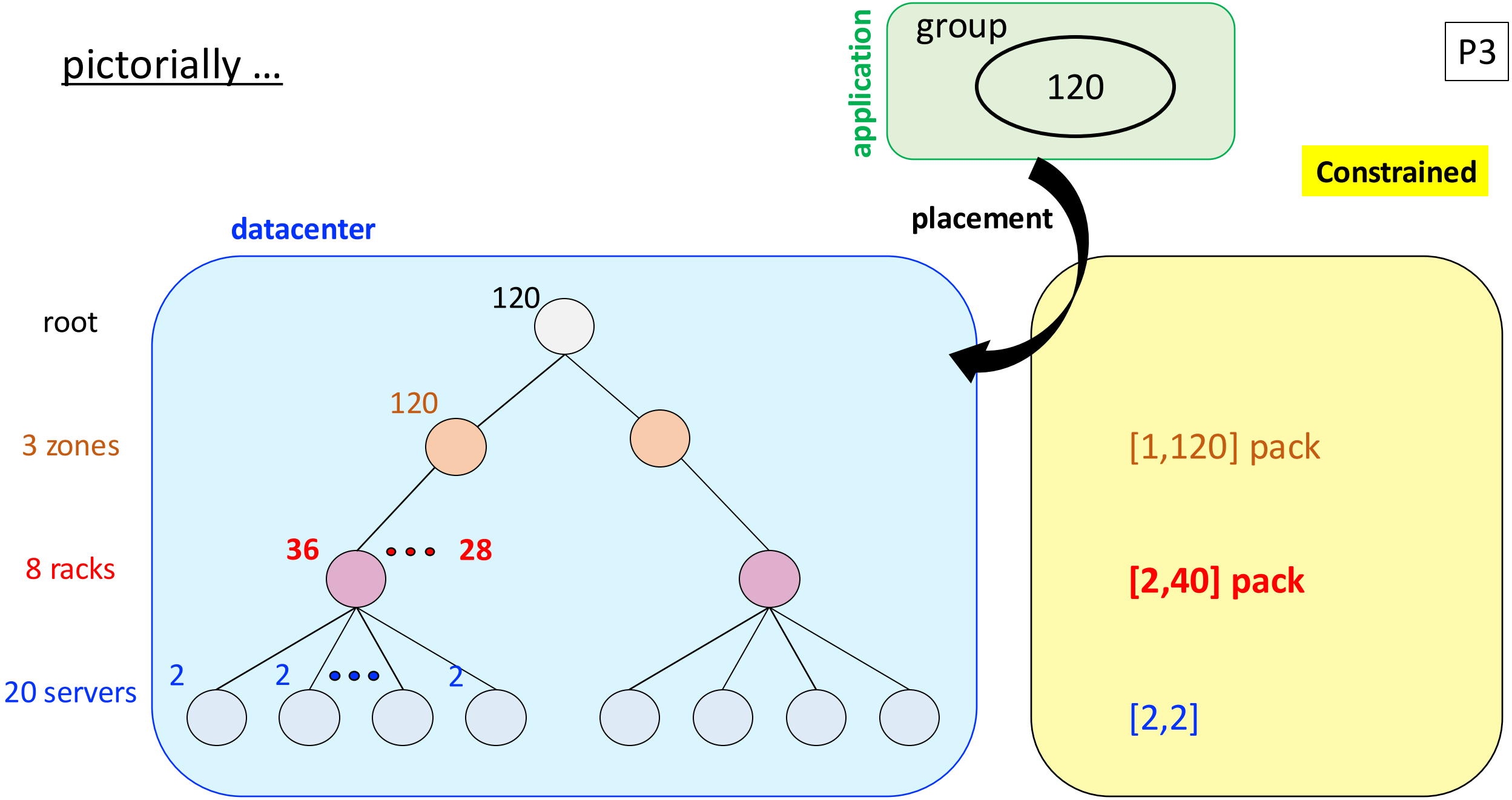
Goal:

zone **affinity**

rack **affinity**

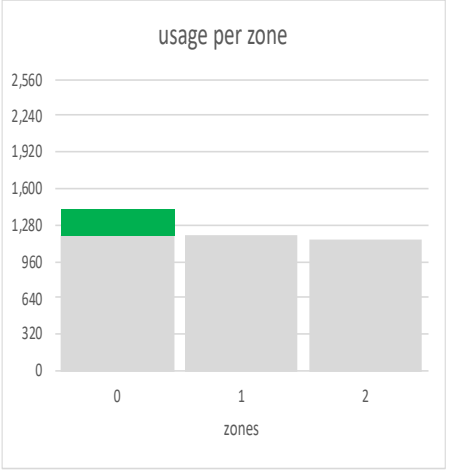
exactly 2 per server

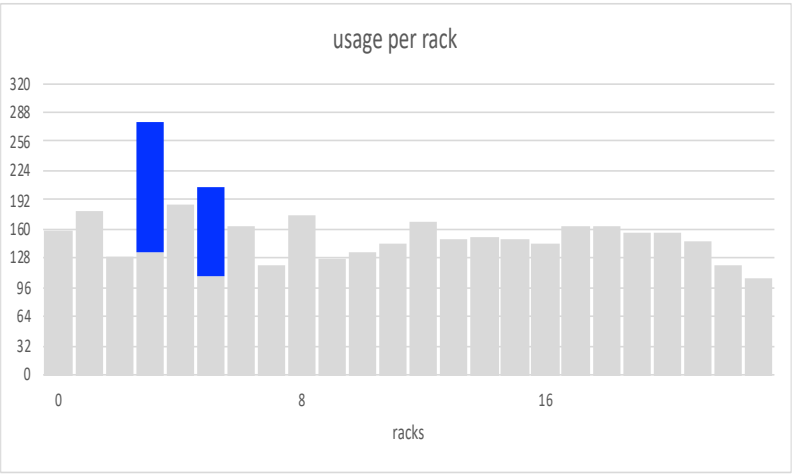
pictorially …

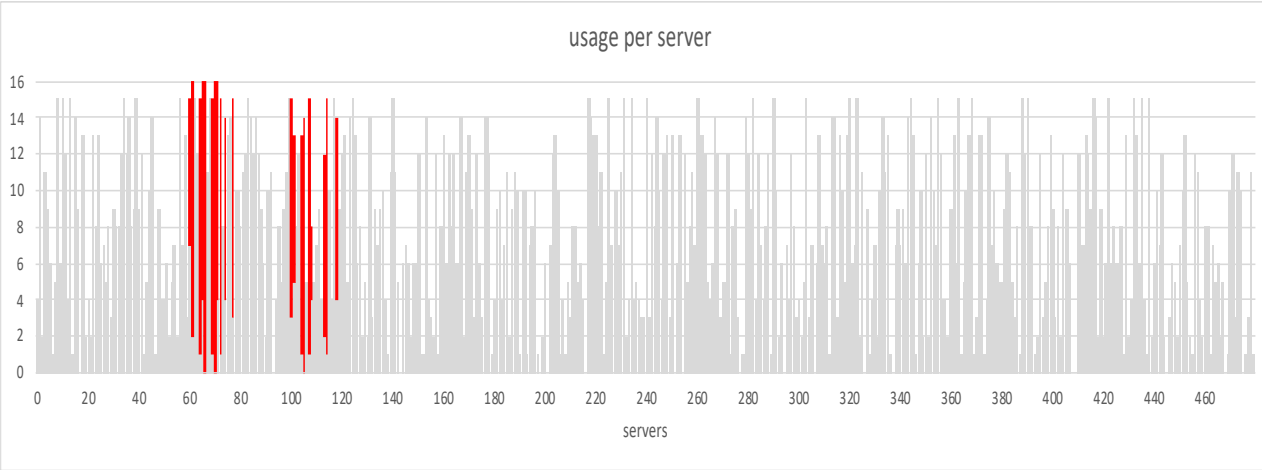


P3

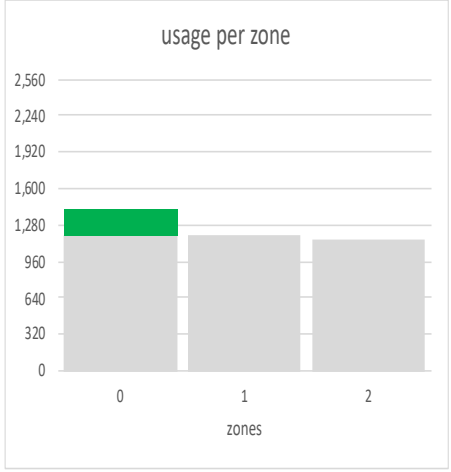
Multiple

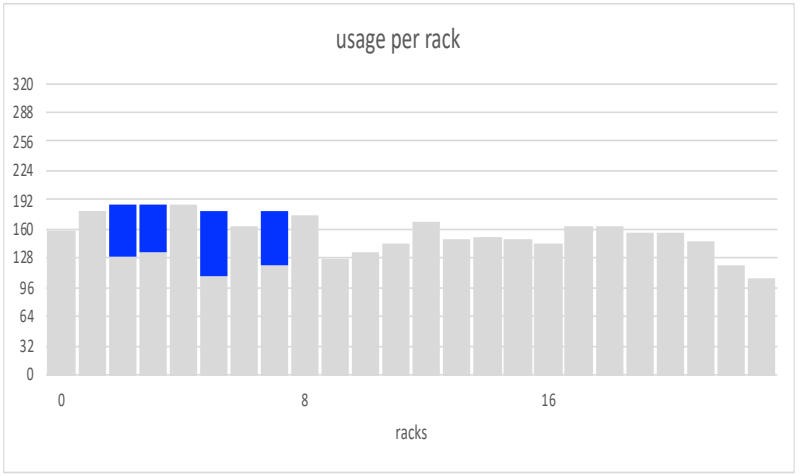


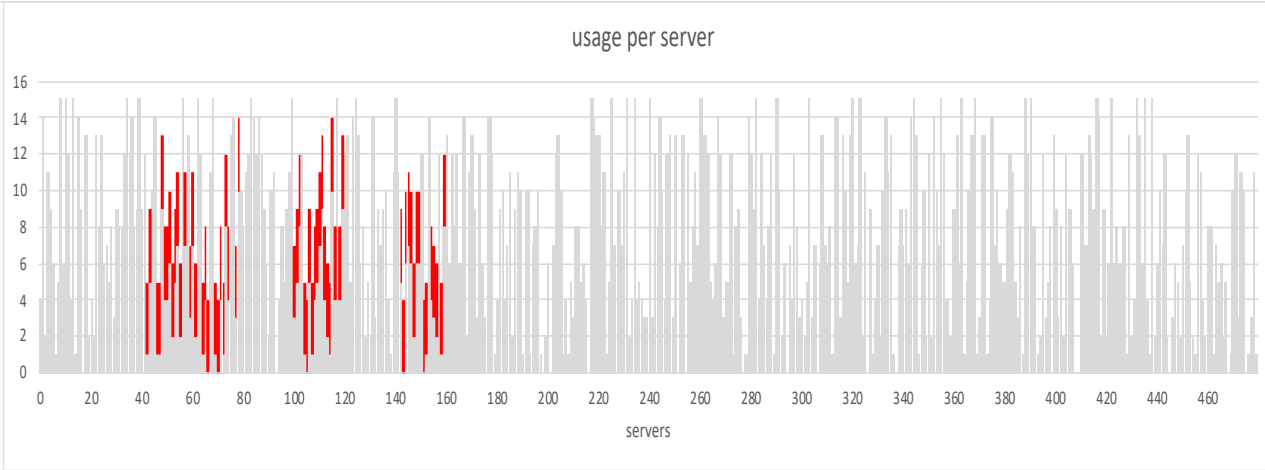




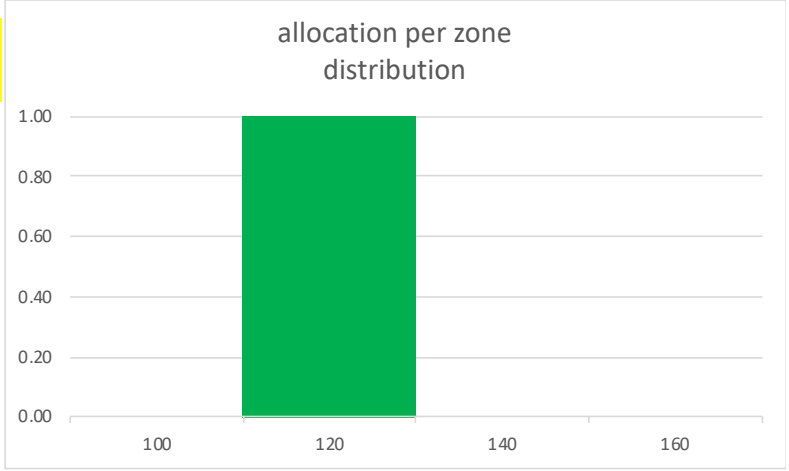
Constrained

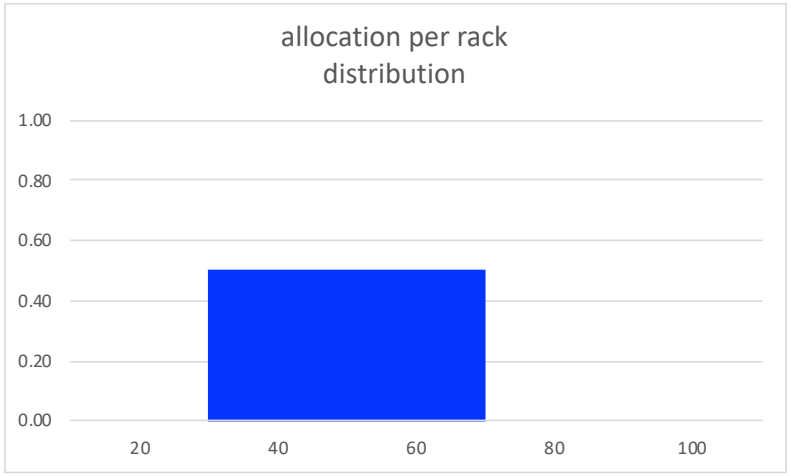


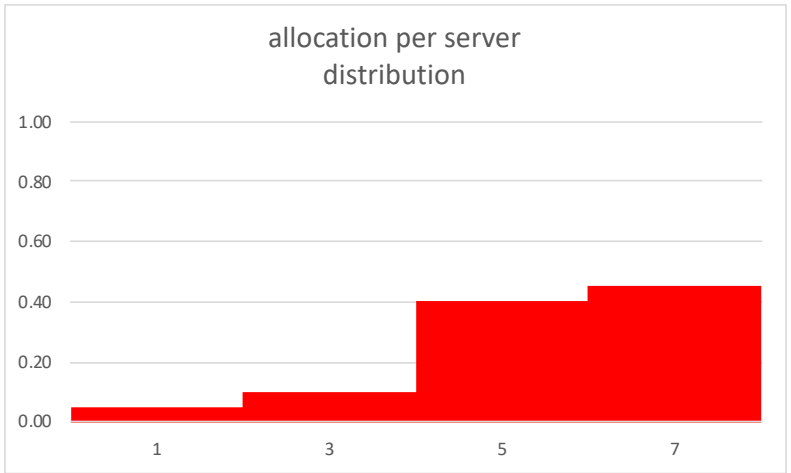




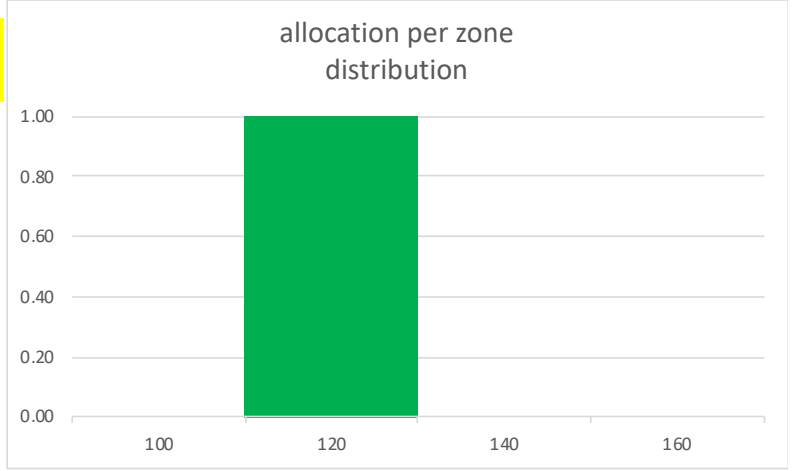
Multiple

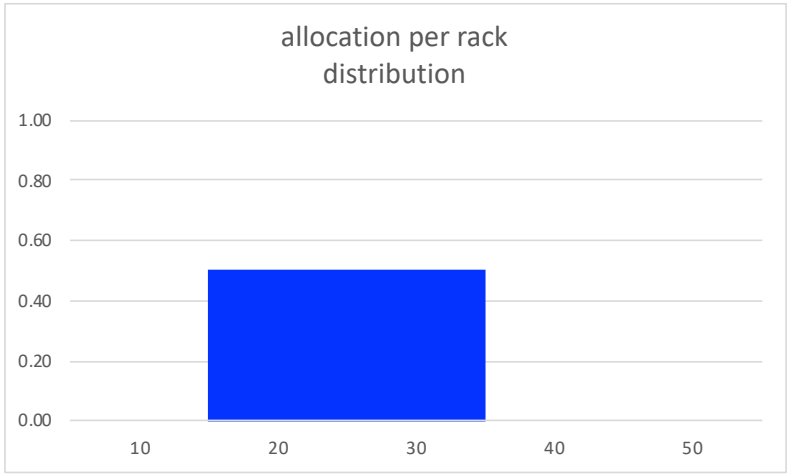






Constrained





P3

Partition Placement Groups

Partitions

- Given placement group of size N instances
  - divide it into M partitions at level L (zone, rack, server)
  - such that the M partitions
    - are placed on separate L-level units (zone, rack, server)
    - each partition contains close to N/M instances
- Example: AWS
  - L is a rack
  - partitions may span zones in same region
  - maximum 7 partitions per zone
  - maximum number of instances per partition depends on account

Specifications

Constrained

```
kind: GroupPlacement
spec:
  group:
    name: MyPartitionGroup
    size: 96
    type: bx2-16x64
  constraints:
    - level: rack
      partitions: 6
```

Problem:

P4

Infrastructure:

3 zones

8 racks per zone

20 servers per rack

Placement group:

size 120

Goal:

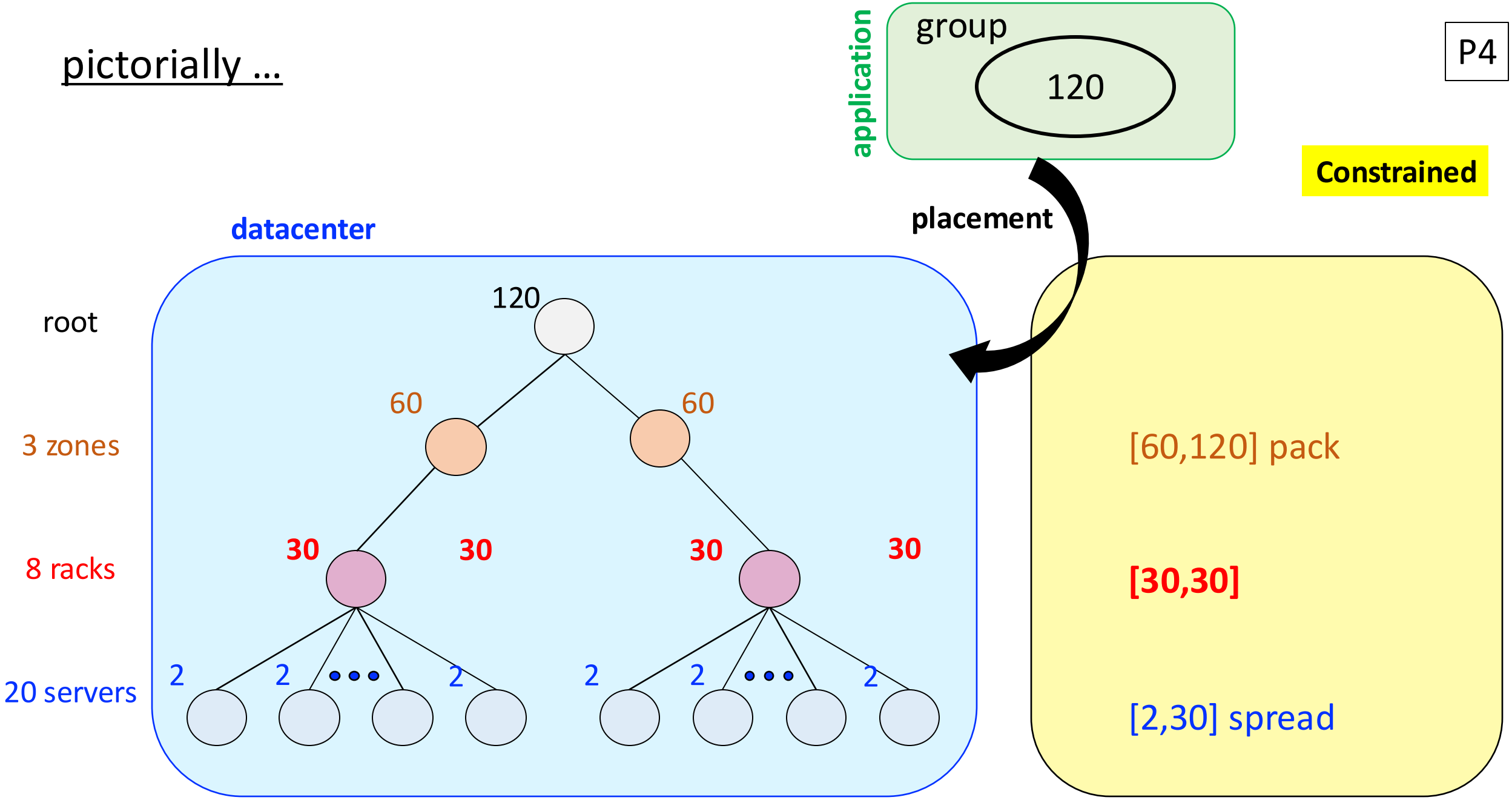
4 **partitions** at rack level

divided **equally** between 2 zones

server **anti-affinity**, but **no less than** 2 per server

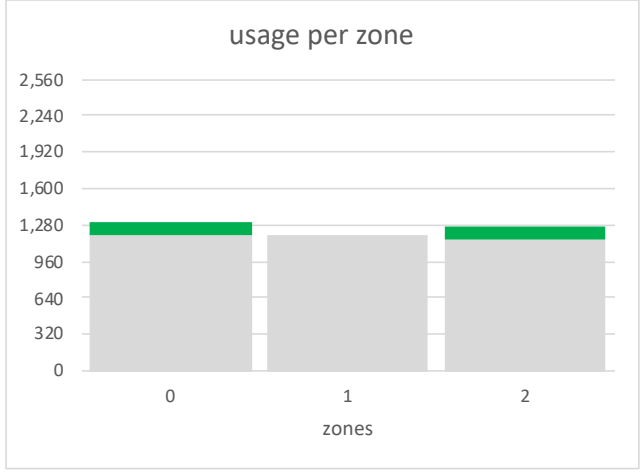
```yaml
kind: GroupPlacement
spec:
  group:
    name: MyApplication4
    size: 120
    type: bx2-16x64
  constraints:
    - level: zone
      partitions: 2
    - level: rack
      partitions: 4
    - level: server
      affinity: spread
      min: 2
```

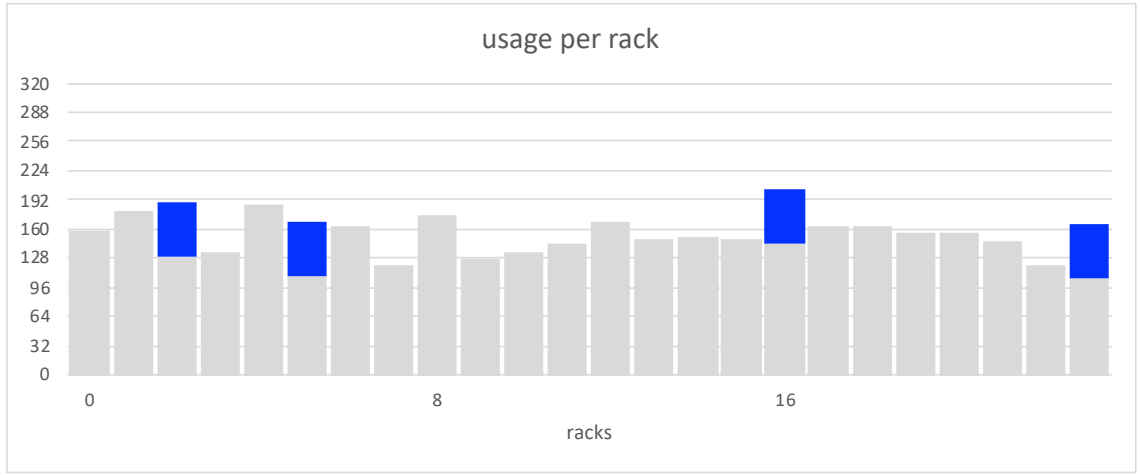
pictorially …

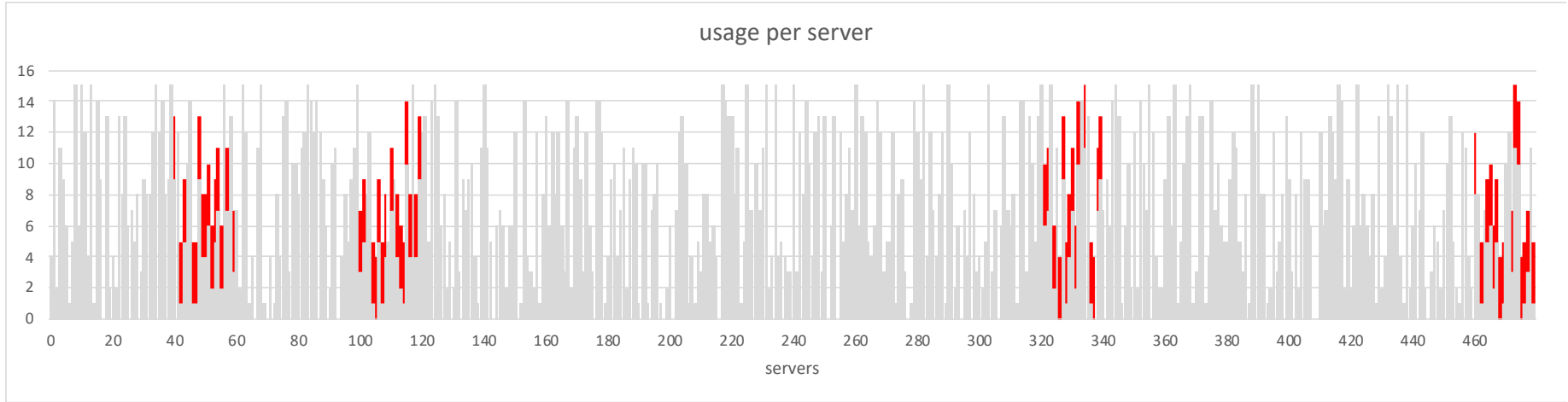


Constrained

P4

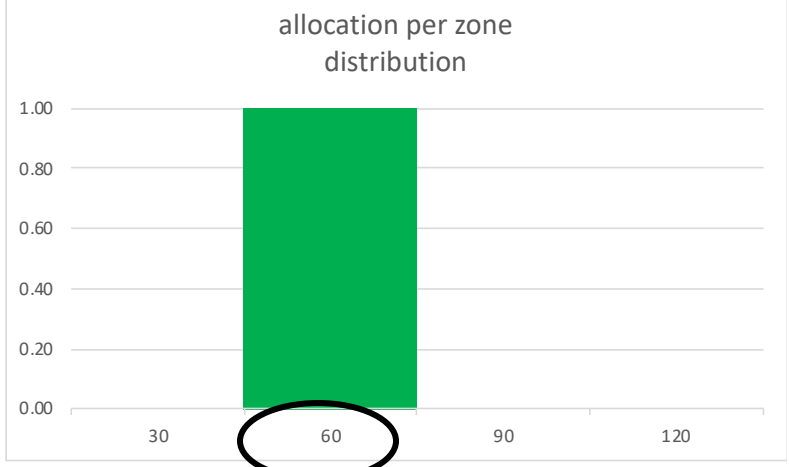


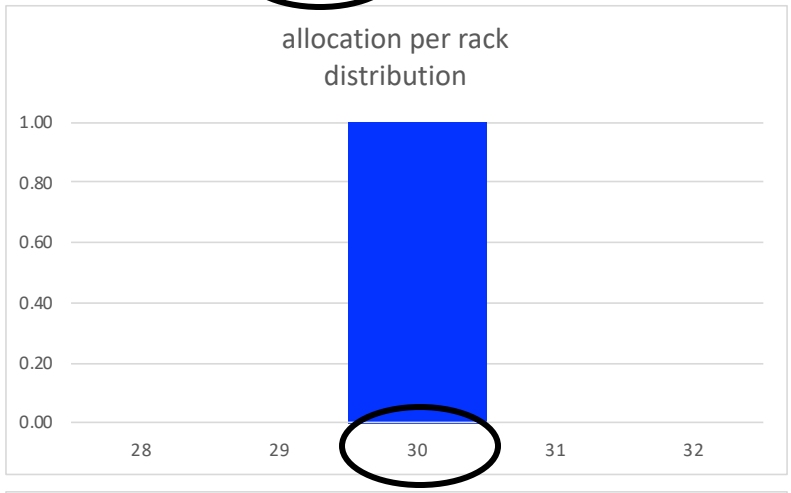


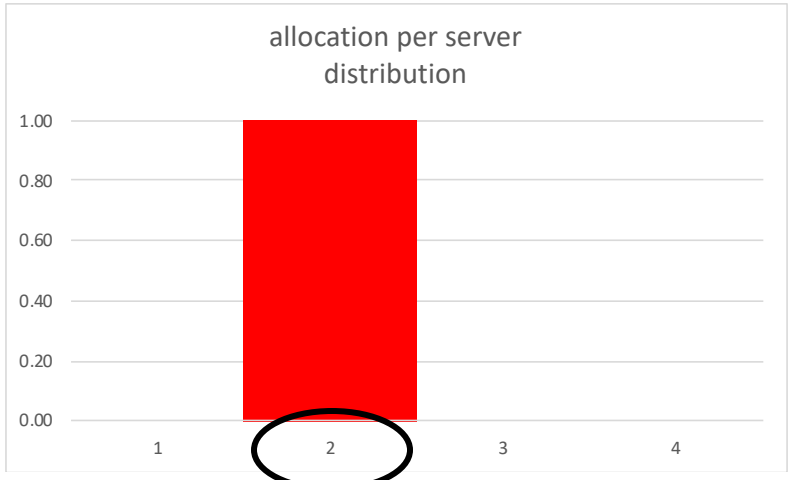


Constrained

P4



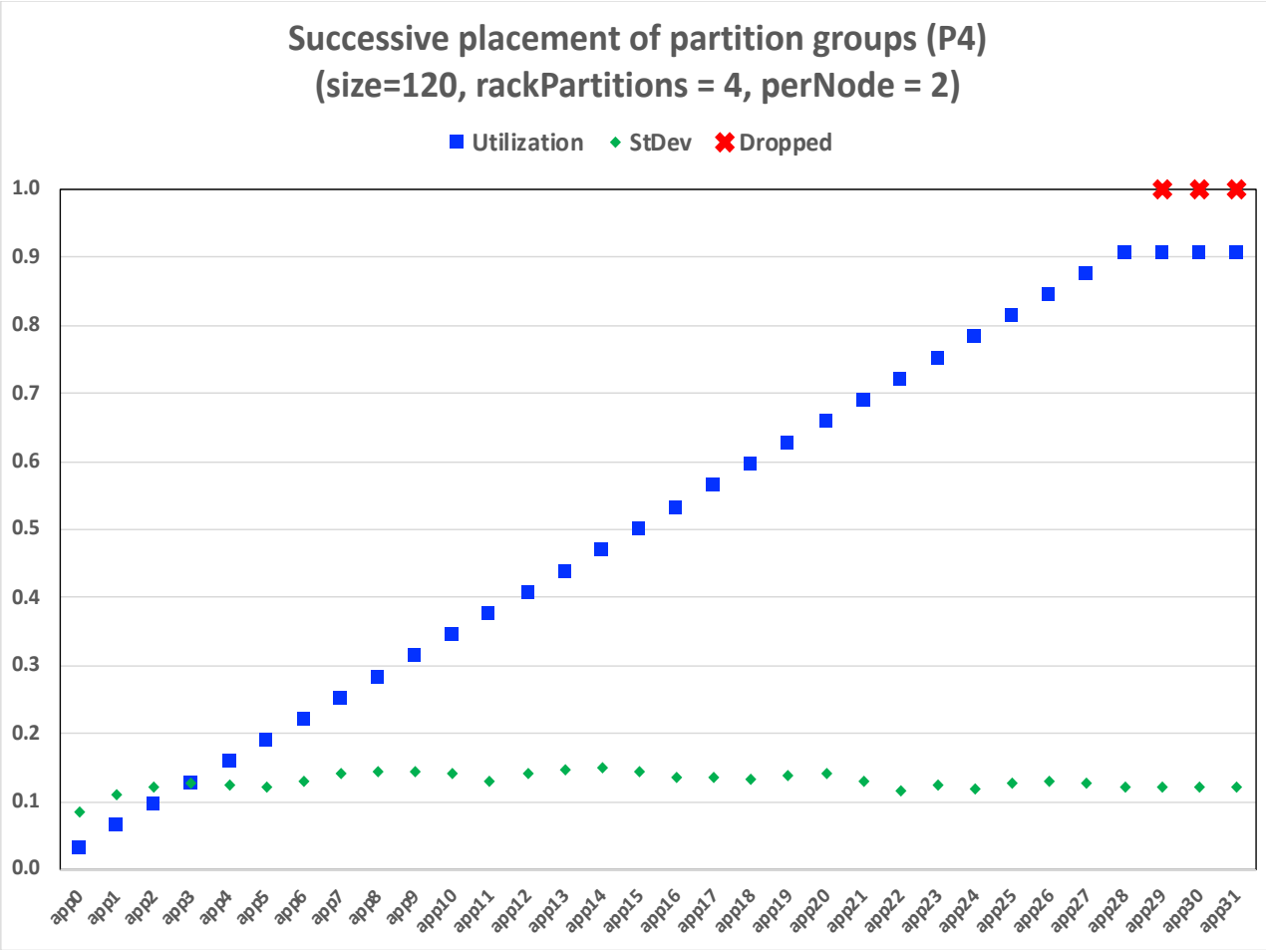




Ramp up loading experiment

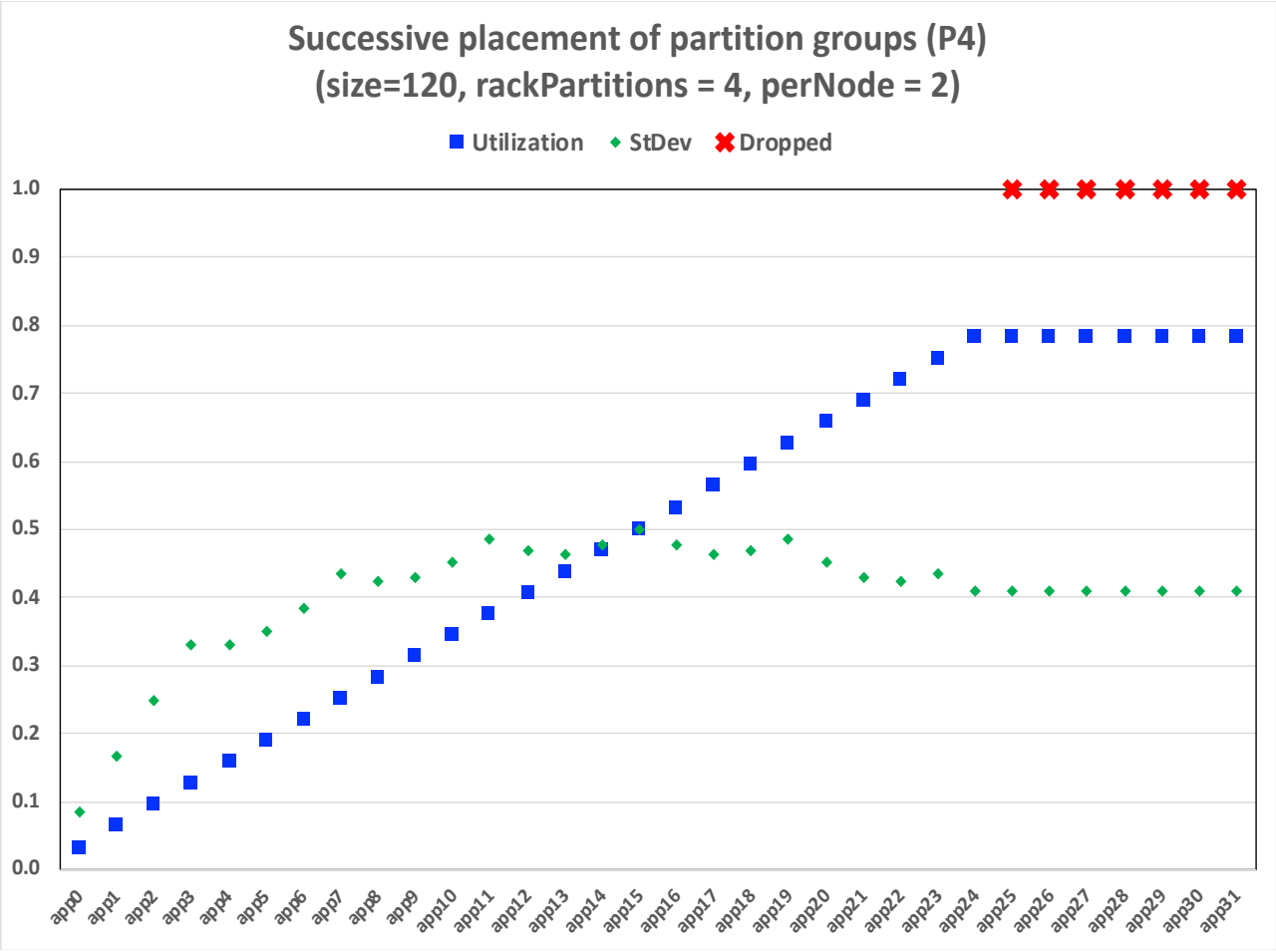
- start with empty data center
- keep placing one partition placement group at a time
  - homogeneous groups (P4)
  - no departures
- until scheduler fails to place
- record average utilization (server allocation)
- consider system placement policies
  - load balance (spread): minimize StDev of utilization across servers
  - consolidate (pack): maximize

policyObjective **LOAD_BALANCE**



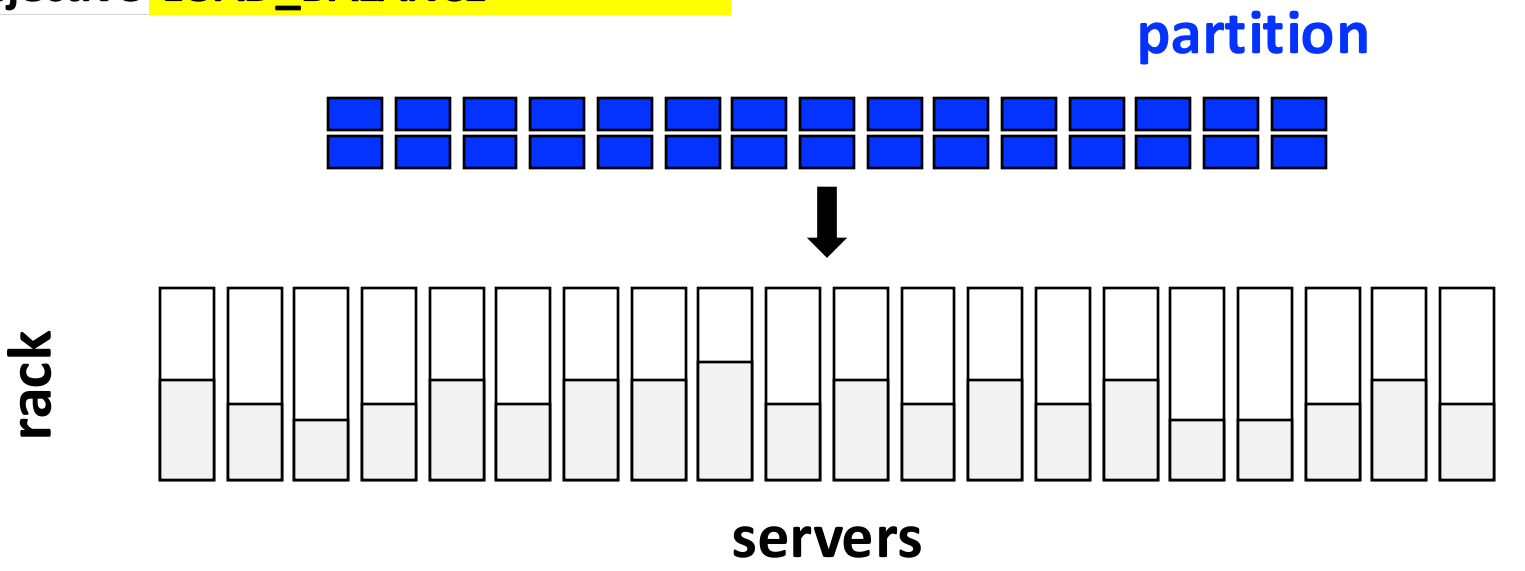
90.6 %

policyObjective **CONSOLIDATE**



78.1 %

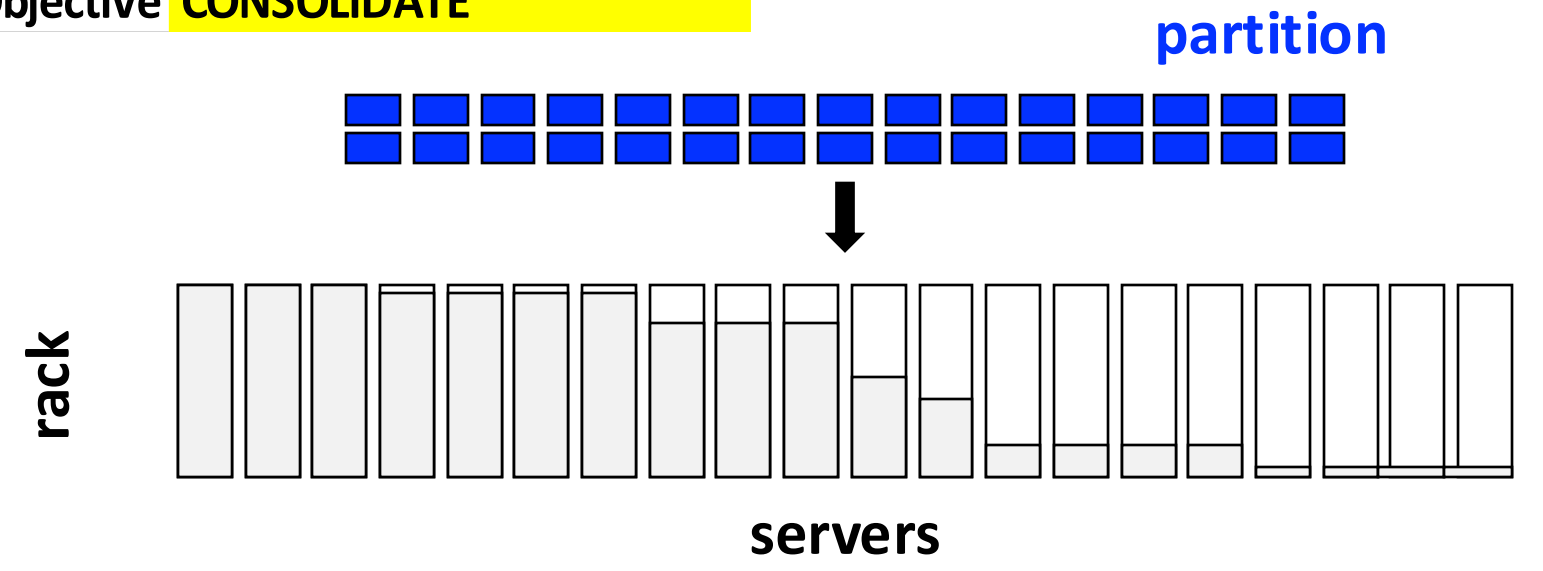
policyObjective **LOAD_BALANCE**



partition of size 30
desired 2 per server

rack of 20 servers
evenly allocated

policyObjective **CONSOLIDATE**



rack of 20 servers
un-evenly allocated