Alibaba Group
阿里巴巴集团

Sigma Scheduler

Hongchao Deng
邓洪超
Software Engineer, Alibaba

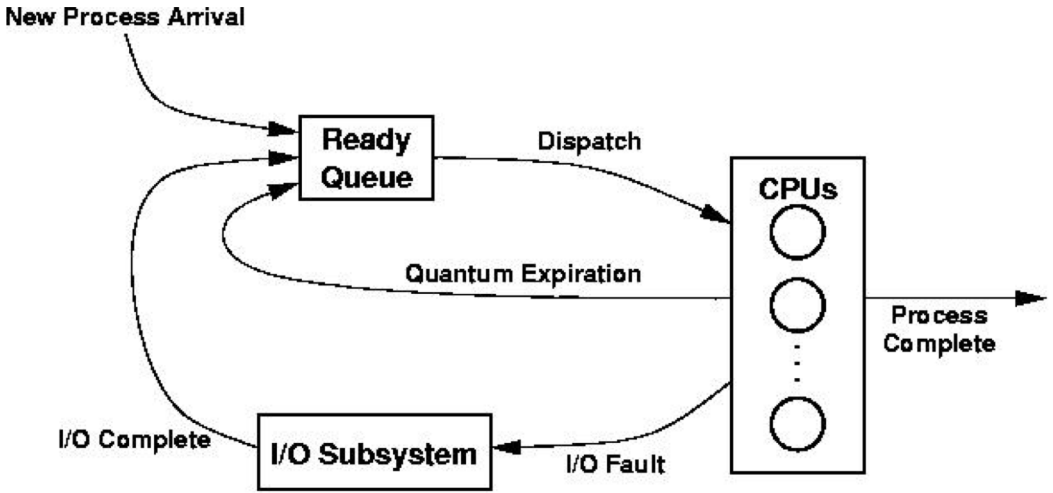
Alibaba Group 阿里巴巴集团

Agenda

- Scheduler overview
- Sigma features
- New challenges

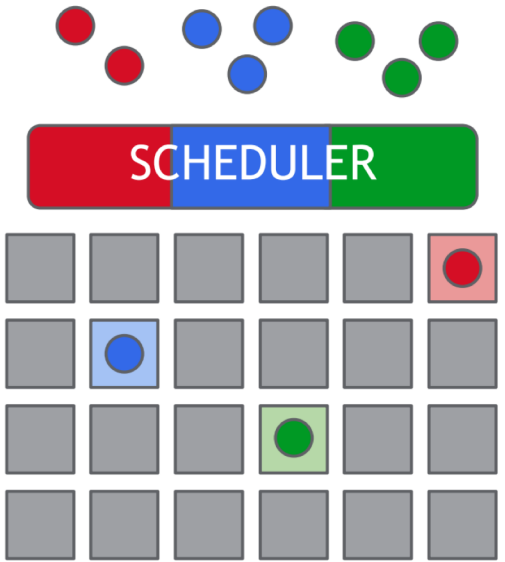
OS scheduler

- Pick a ready ***process*** to run on an available ***CPU***
  - **Enabling time-sharing**
  - Maximizing throughput
  - Minimizing wait time
  - Maximizing fairness
  - Meet deadlines



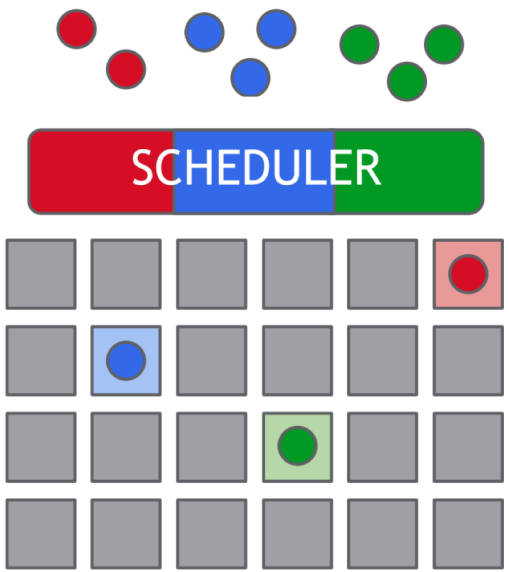
Cluster scheduler

- Find a **node** that has enough resources to run a **workload**
  - **Enable cluster sharing**
  - Maximizing machine utilization
  - Minimizing batch job runtime
  - Improving load balancing / fault tolerance
  - Predictable application performance

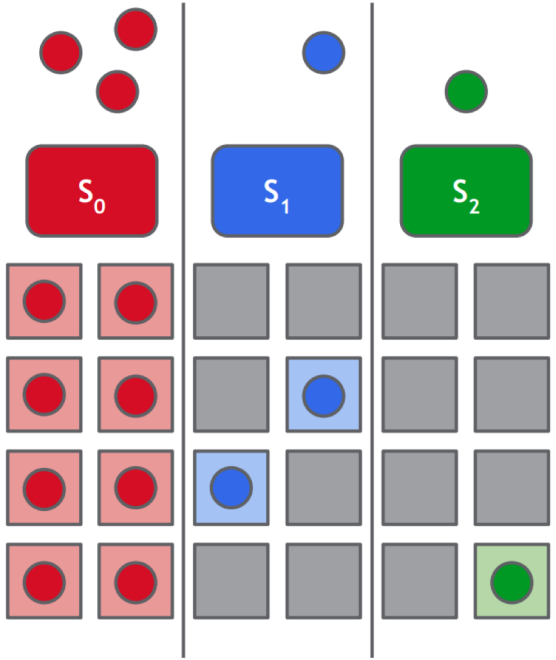


Architecture

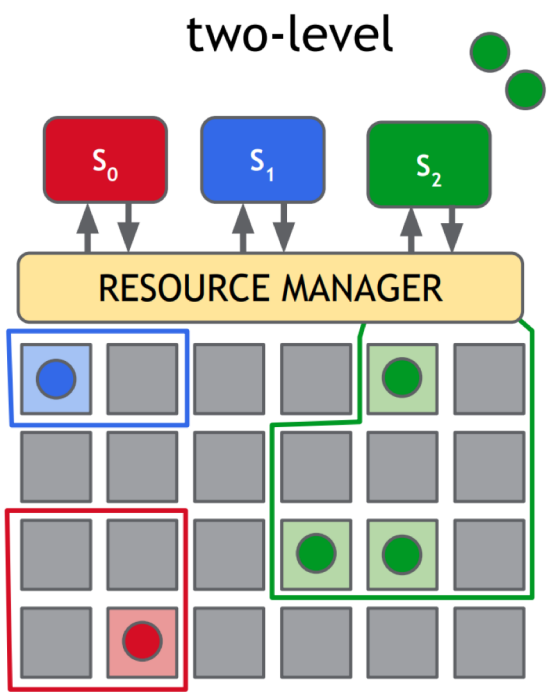
monolithic scheduler

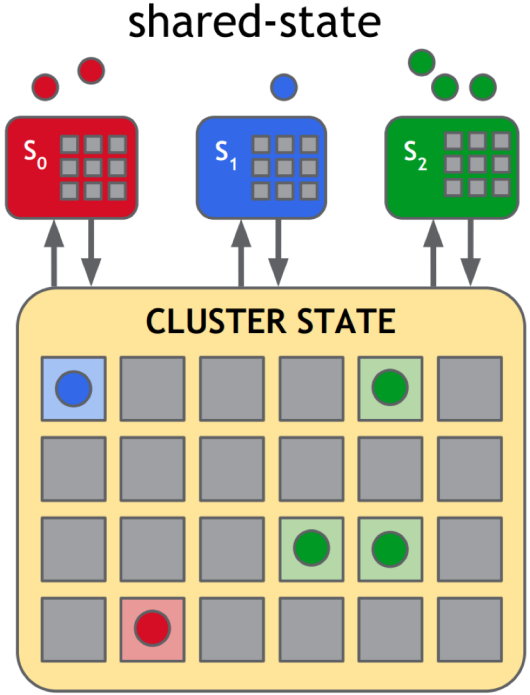


static partitioning



Architecture





Algorithm

- Queue based approach

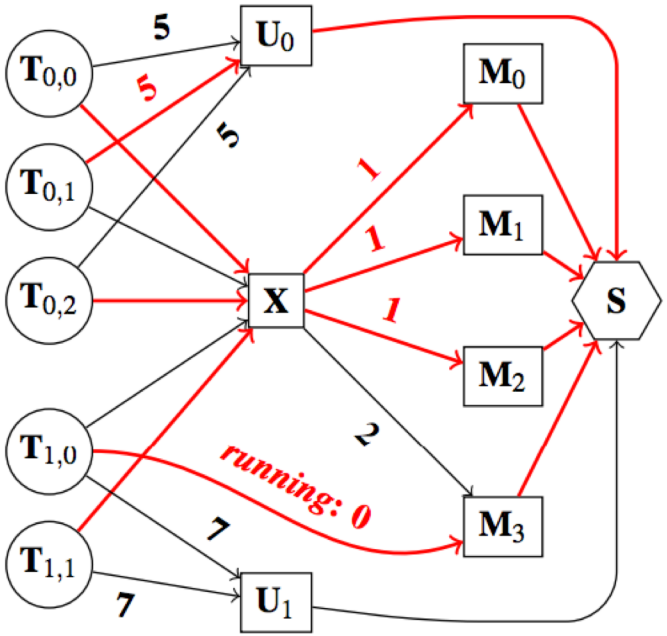
Algorithm

- Queue based approach

```
for each app:
    possibleMachines = filter(allMachines)
    sortedMachines = weight(possibleMachines)
    for each pod in app.Pods:
        assign(pod, sortedMachines)
```

Algorithm

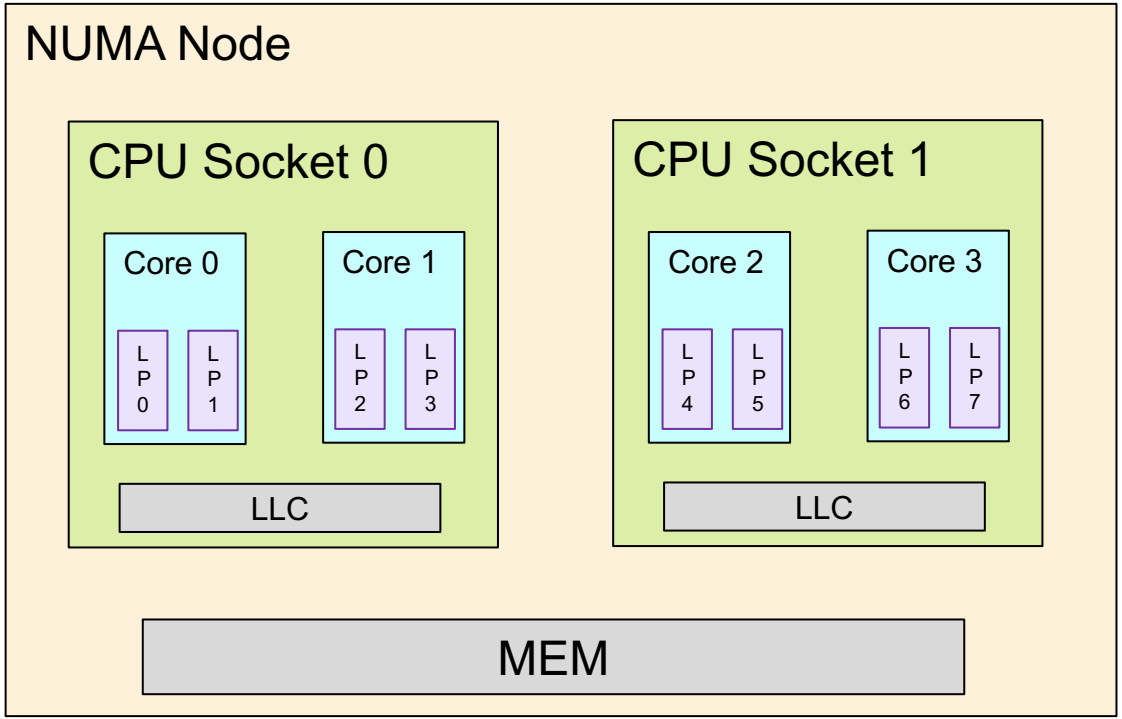
- Graph based approach



Sigma scheduler

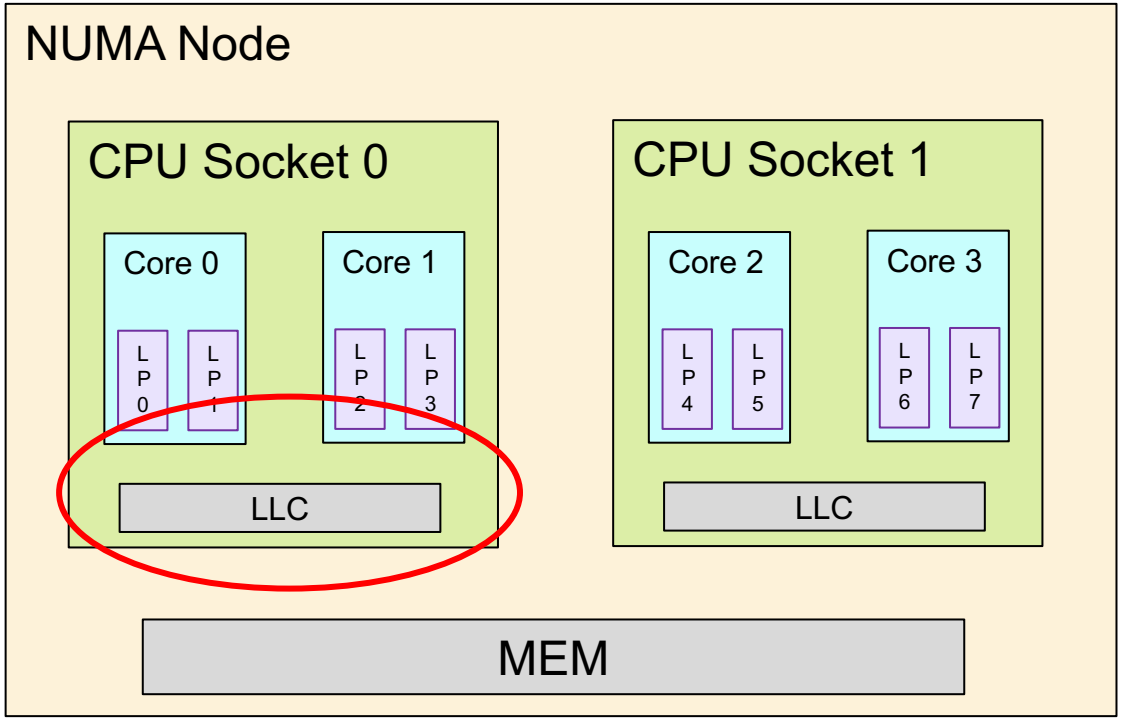
- A shared-state scheduler mainly focuses on long-running services
  - Predictable performance
  - Fault tolerance and load balancing

CPU topology awareness scheduling



Alibaba Group 阿里巴巴集团

CPU topology awareness scheduling



Alibaba Group 阿里巴巴集团

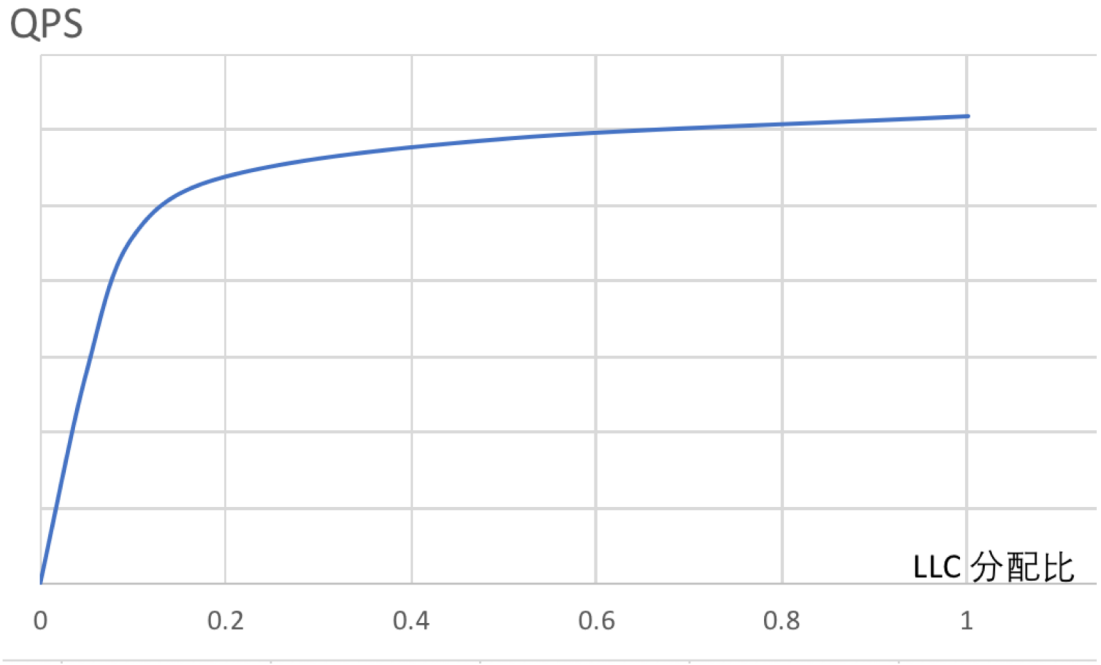
CPU topology awareness scheduling

memkeyval

	5%	10%	15%	20%	25%	30%	35%	40%	45%	50%	55%	60%	65%	70%	75%	80%	85%	90%	95%
LLC (small)	115%	88%	88%	91%	99%	101%	79%	91%	97%	101%	135%	138%	148%	140%	134%	150%	114%	78%	70%
LLC (med)	209%	148%	159%	107%	207%	119%	96%	108%	117%	138%	170%	230%	182%	181%	167%	162%	144%	100%	104%
LLC (big)	>300%	>300%	>300%	>300%	>300%	>300%	>300%	>300%	>300%	>300%	>300%	>300%	>300%	280%	225%	222%	170%	79%	85%

Alibaba Group 阿里巴巴集团

CPU topology awareness scheduling

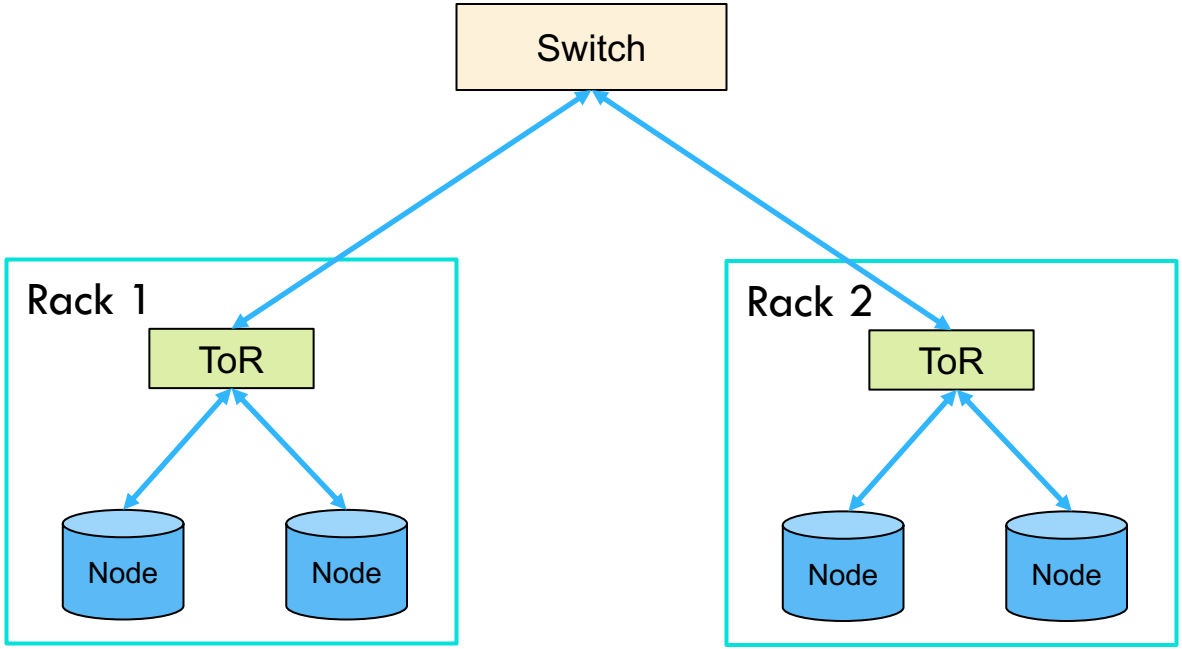


Alibaba Group 阿里巴巴集团

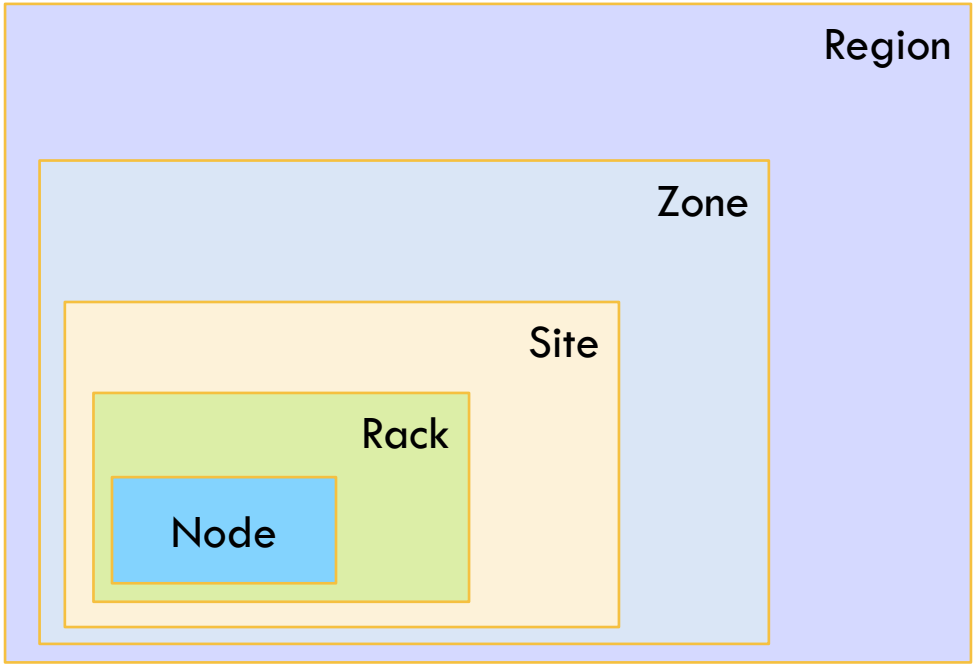
CPU topology awareness scheduling

- Spreading
  - Physical core
  - Socket
- Exclusive
  - Physical core

Cluster topology awareness



Cluster topology awareness



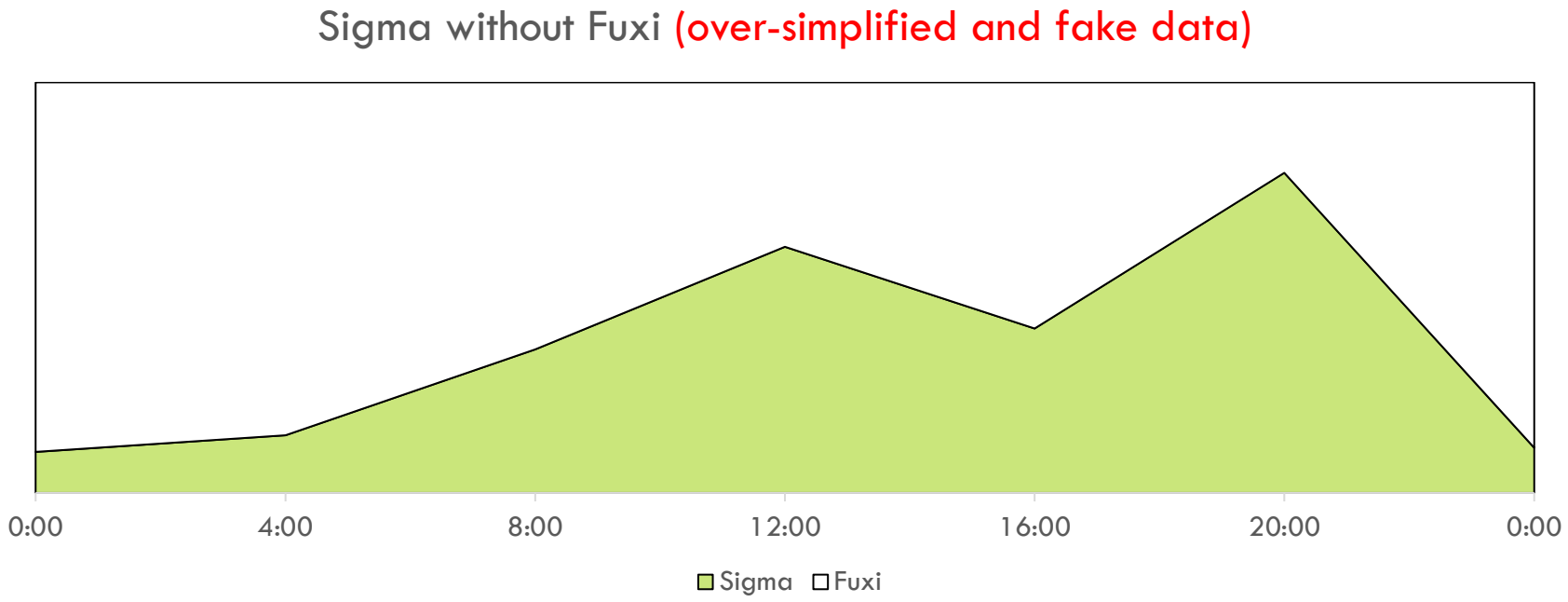
Cluster topology awareness

- Anti-affinity
  - Spreading deployments to different machines, racks, zones, etc
- Affinity
  - Data locality
  - Network locality

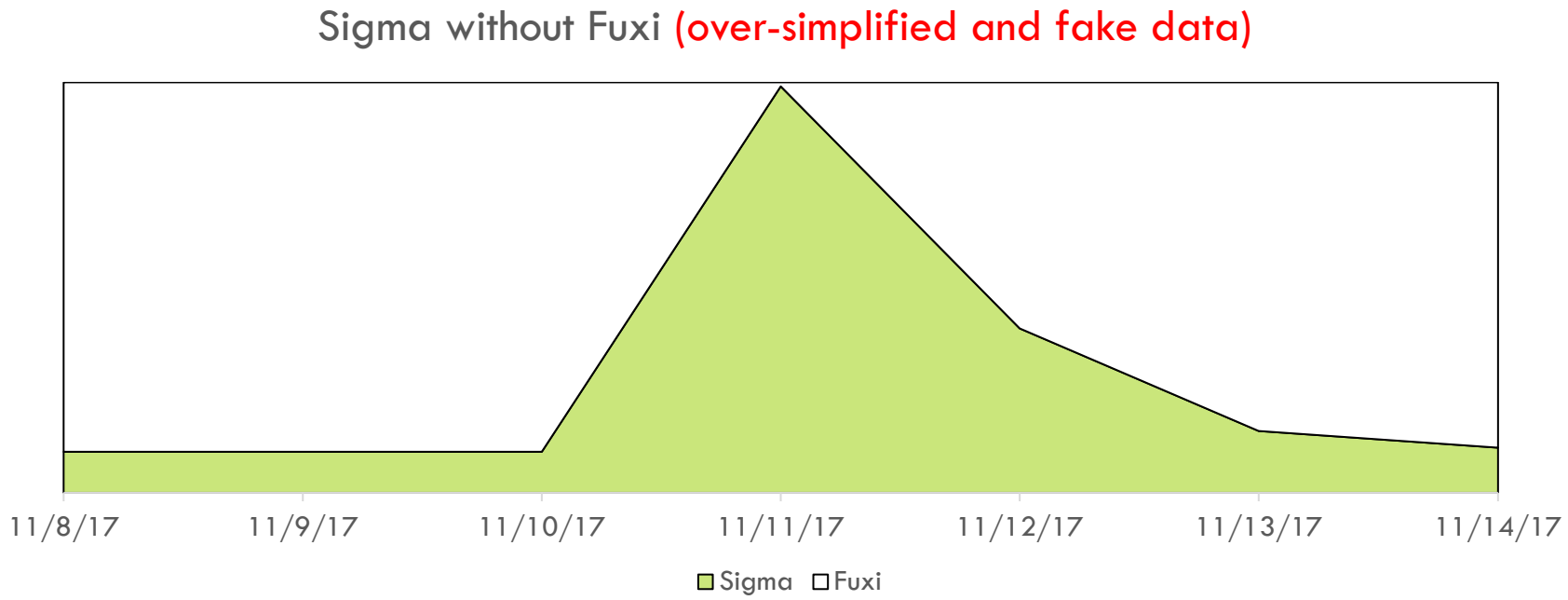
Corporative scheduling

- Sigma
  - Container scheduling system with a focus on production services
  - Over-provision for peak workload
- Fuxi [VLDB Sept. 2014]
  - Job scheduling system
  - Flexible resource usage pattern

Corporative scheduling

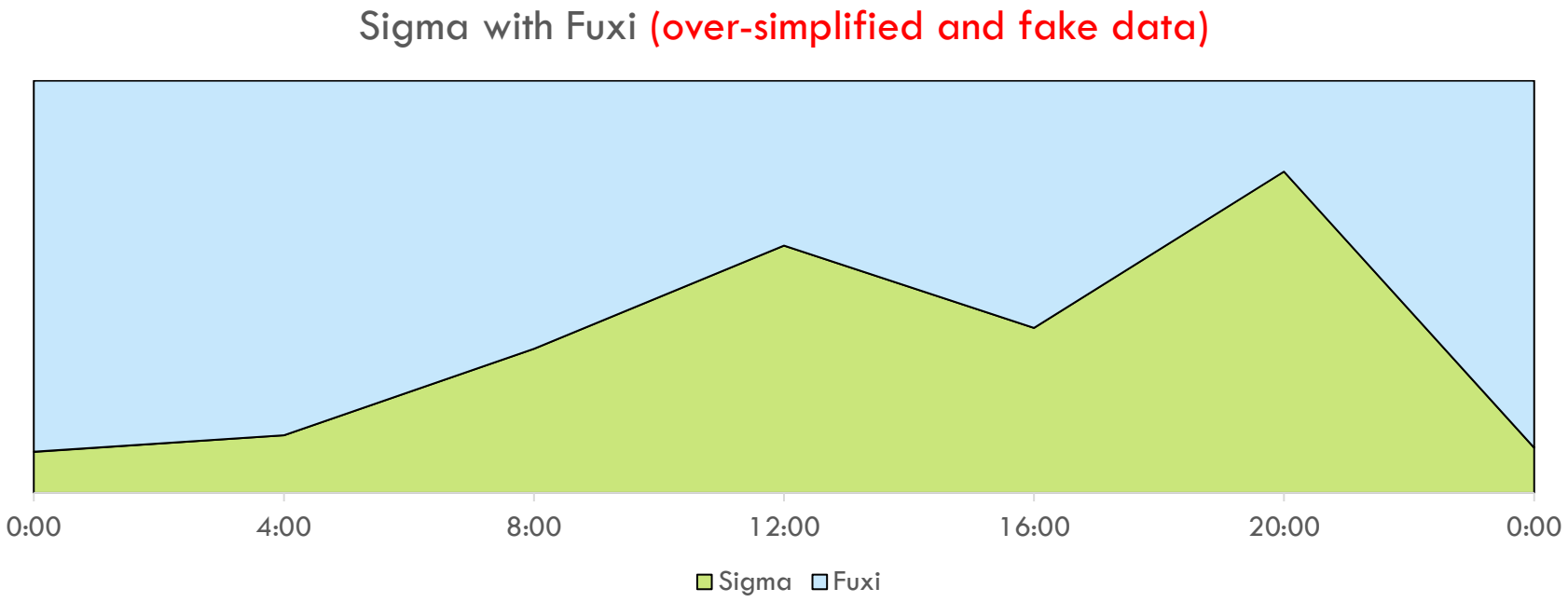


Corporative scheduling



Alibaba Group 阿里巴巴集团

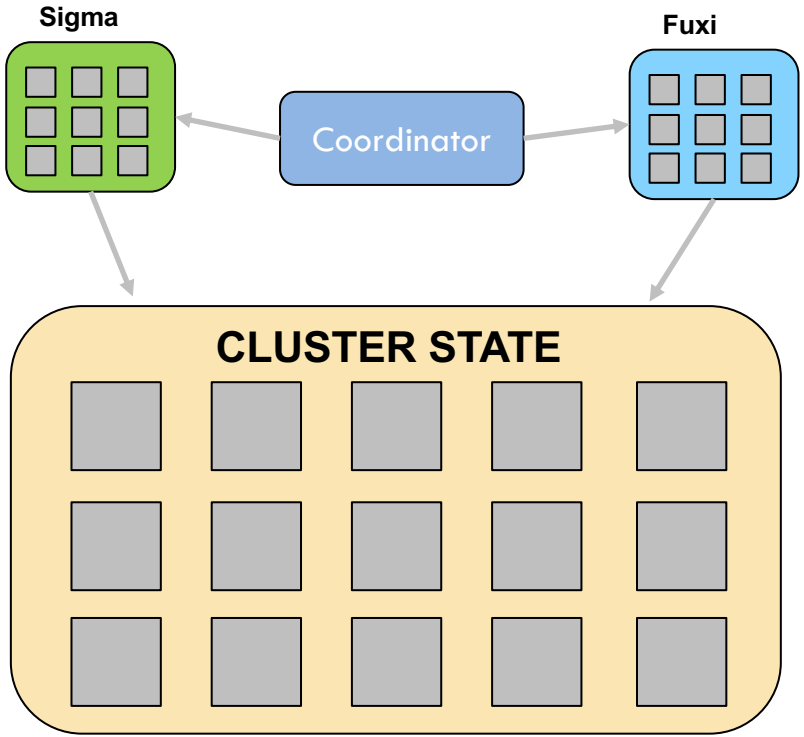
Corporative scheduling



Corporative scheduling

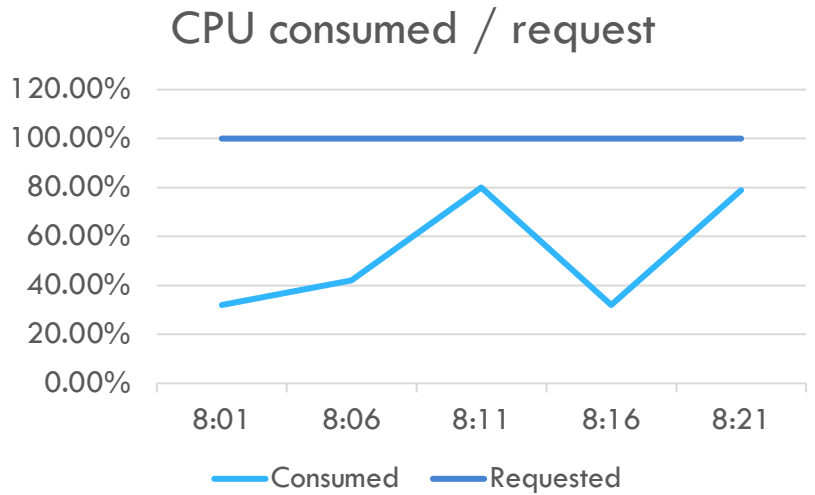
- Preemption
  - Tiered priority between different schedulers
  - Gold, Sliver, Copper

Corporative scheduling



Resource usage prediction

```
JobA {
    CPU: 2.00 Cores,
    Memory: 2.00 GB,
}
```

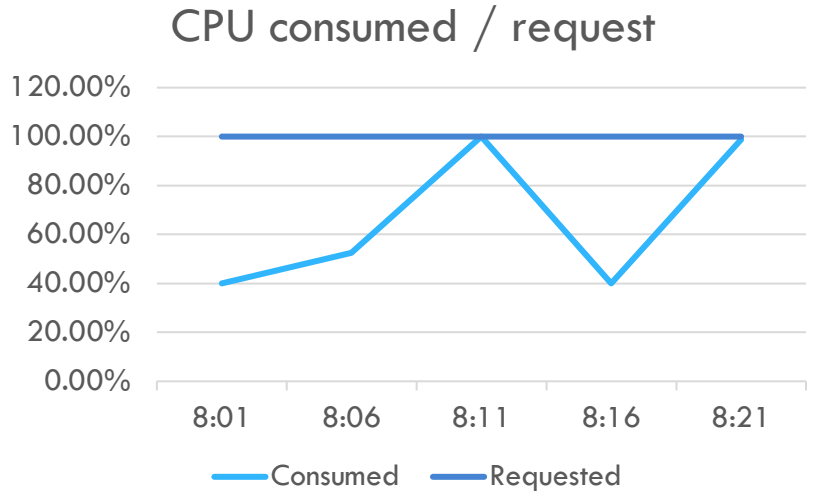


Resource usage prediction

- It is hard for users to estimate the usage
  - Users tend to over provision to ensure SLA
  - Users tend to just forget about the task resource setting if it runs well
  - Users tend to reuse resource setting even if there is a new release
- Statistic approach works well enough without affecting SLA
  - P99 over the last 7 days for the same task built

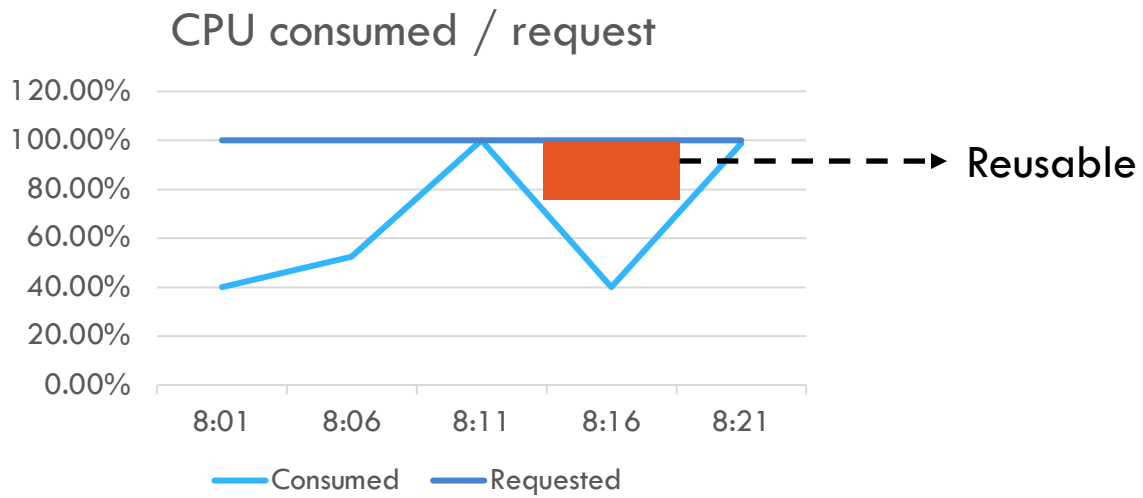
Resource usage prediction

```
JobA {
    CPU: cP99(JobA),
    Memory: mP99(JobB),
}
```



Alibaba Group 阿里巴巴集团

Over-subscription



Over-subscription

- Allocate more resources than allocatable
  - Best effort batch jobs
  - Preempt low-priority best effort jobs to reclaim resources
- Benefits of over-subscription
  - Apps have diurnal patterns
  - Save budget by reusing resources

Over-subscription

- Resource usage forecast
  - Look into the next hours
    - Most of batch job finish in hours
  - Simple ML algorithms work reasonably well
    - *Query-based Workload Forecasting for Self-Driving Database Management system*
    - *Resource Central: Understanding and Predicting Workloads for Improved Resource Management in Large Cloud Platforms*

Smart scheduling

- Dynamic adjustment based on historical data
  - Image fetch time
  - Performance
  - Stability

Rescheduling

- Node level rescheduling
  - Reduce CPU cores fragmentation
- Cluster level rescheduling
  - Reduce CPU, Memory resources fragmentation
  - Increase policy enforcement level
    - Rebalance workload to help with soft affinity and soft anti-affinity
  - Rebalance workload onto newly added machines


Alibaba Group
阿里巴巴集团
Thanks!
We are hiring!
hongchao.deng@alibaba-inc.com