JoySim: Simulating Kubernetes Clusters at Scale

Yuan Chen, Qichao Lu, Jiangang Fan, Guang Zhou, Haifeng Liu

JD.com

About JD.com

China's largest online and overall retailer and biggest Internet company by revenue

- 300 million+ active users
- 2018 revenue: $67.2 billion

China's largest e-commerce logistics infrastructure and fulfillment network

- 550+ warehouses
- Covering 99% of population
- Standard same-and next day delivery

First Chinese internet company to make the Fortune Global 500

Strategic partnerships Tencent Walmart Google

京东

JD Retail Technical Infrastructure Group

Provide and manage hyperscale containerized infrastructures and platforms for all JD services

- One of the earliest adopters of Kubernetes
- Run large scale Kubernetes clusters in production
- CNCF Platinum Member
- 2018 CNCF End User Award

Support 2019 Singles Days Sales

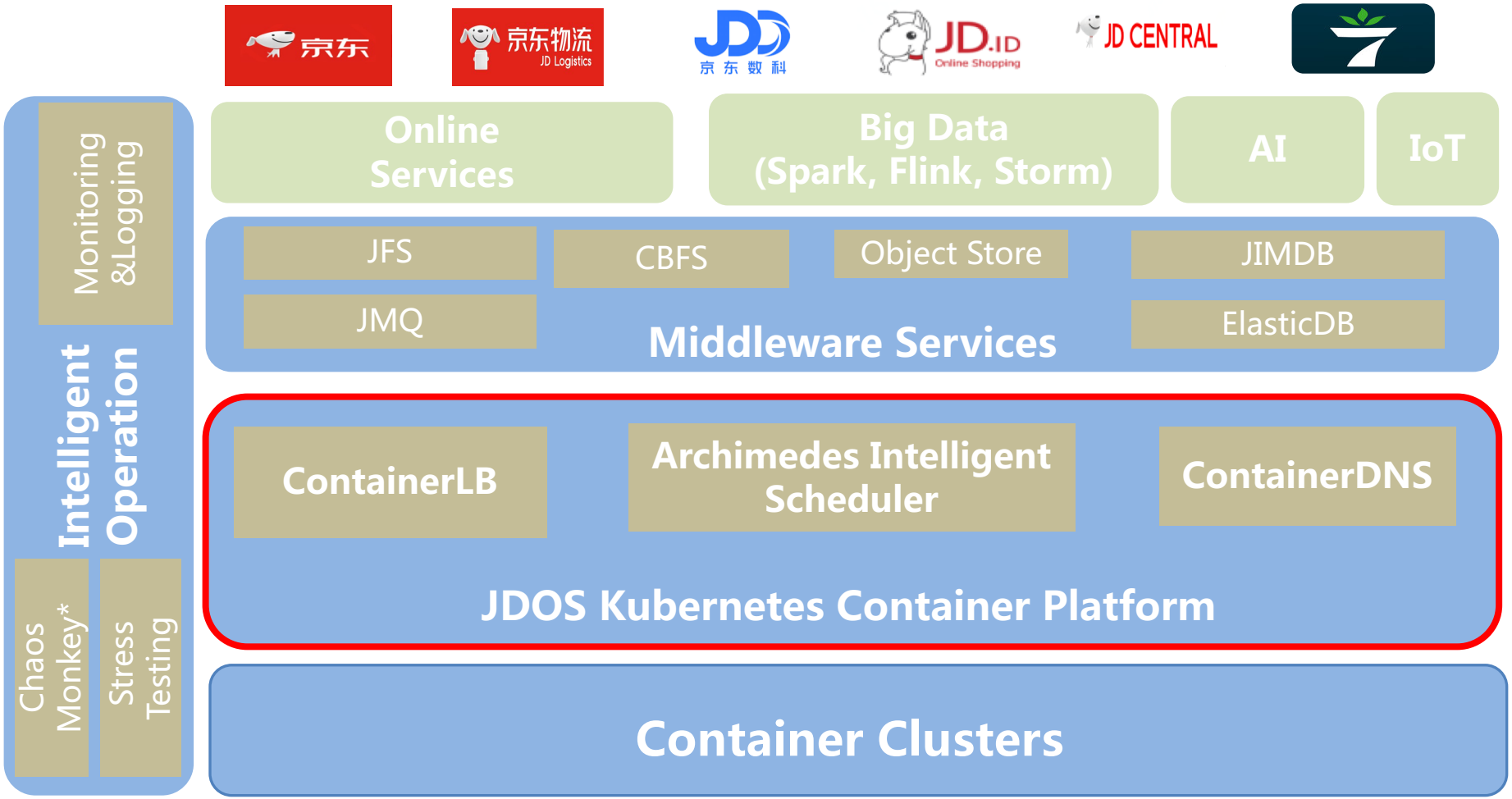
- ￥204.4 billion (about $29 billion) !

"We are thrilled to have JD... By sharing their Kubernetes experiences and investing directly in the project, JD.com is helping to spread cloud native computing throughout China", -Dan Kohn, Executive Director of the CNCF

http://tig.jd.com/en

京东

JD Container Platform



京东

Why Do We Need A Simulator?

- Evaluate new configurations and features before deploying in a production Kubernetes cluster.
  - **Scheduling optimization performance and quality**
  - Kubernetes Master (e.g., APIServer, ETCD) performance and scalability
- Time consuming and costly to perform such evaluations at scale.
- Simulation is a useful technique and tool!

京东

Cluster Simulation Tools

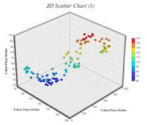
	Complex Scenarios	Performance	Scheduling Quality	Visualization
Kubernetes				
Mesos				
Yarn				

京东

Key Requirements

Simulate large-scale K8s clusters with a small amount of resources

Mimic real behaviors with a reasonable amount of confidence



Enable to evaluate scalability, performance and quality

Support realistic and complex scenarios

Comprehensive monitoring and visualization

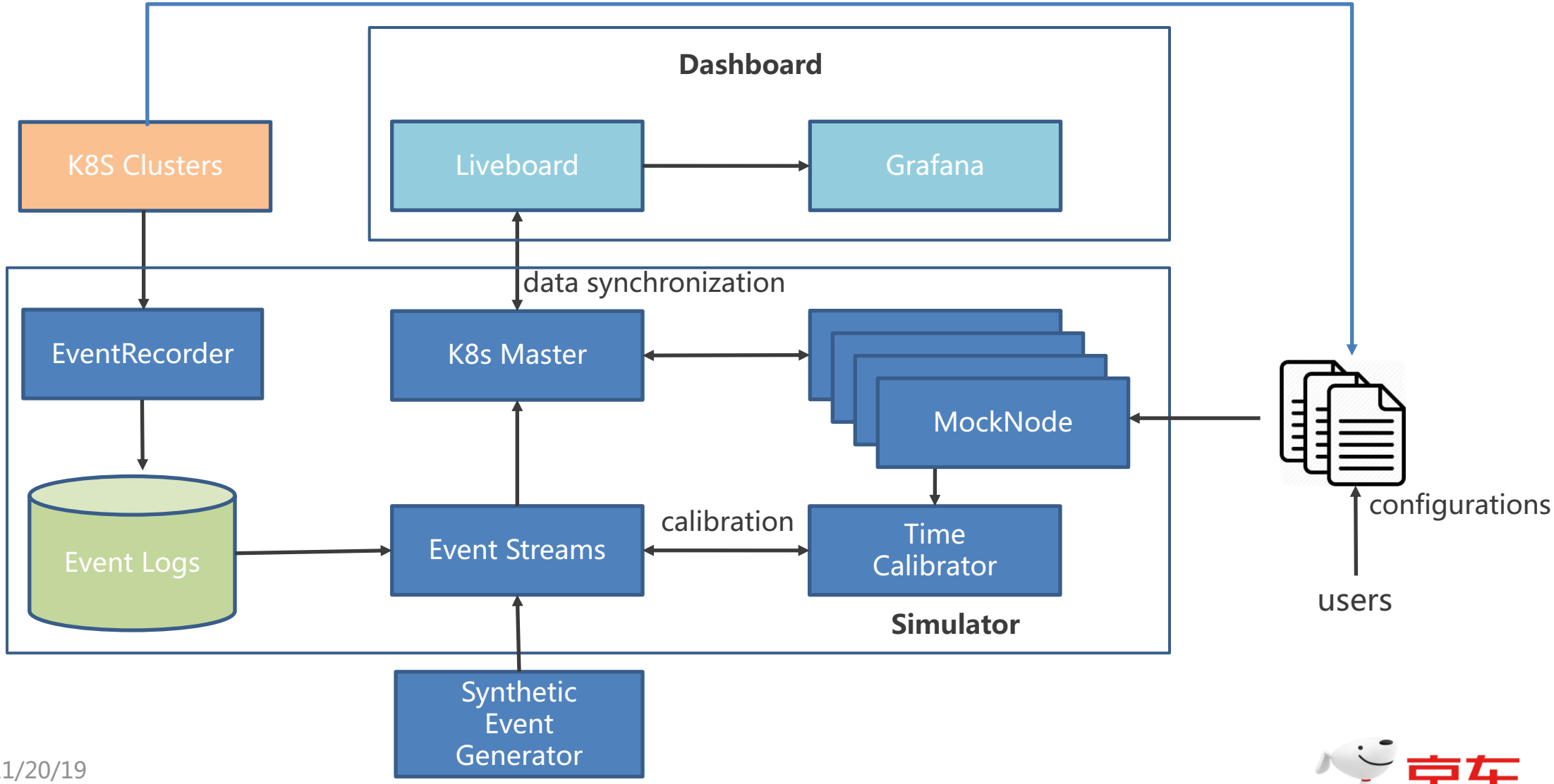
Outline

- Background and Introduction
- JoySim Design and Implementation
- Use Cases
- Conclusion
- Demo

JoySim: A Kubernetes Cluster Simulator

- Run real K8s masters: APIServer, Scheduler, ETCD, ControllerManager.
- Simulate nodes (Kublet) using lightweight MockNodes.
- Configuration information from real Kubernetes clusters or configuration files.
- Replay events from real K8S clusters or customized scenarios.
- Comprehensive monitoring: resource utilization, scheduling traces, performance.
- Easy to deploy, configure, manage and scale using K8s.

Architecture



京东

Events

- Collection from real Kubernetes production environments
  - Pods creation, deletion, preemption, rescheduling …
  - Collection agent: **Telegraf**
  - Storage: **InfluxDB**
- Customized scenarios generated by a synthesizer
  - Cluster and node template: size and resource capcities
  - Pod template: request type and size
  - Event template: request arrival rate, inter-arrival time (e.g., deterministic, random, exponential distribution)


```
{
    "type":"ADDED",
    "object":{
        "kind":"Configmap",
        "metadata":{
            "name":"newjdoslog-ht03",
            "namespace":"group-manager",
            "selfLink":"/api/v1/namespaces/group-manager/configmaps/newjdoslog-ht03",
            "uid":"41ec6f68-54fc-11e8-9dec-f898efe7ce8c",
            "resourceVersion":"445984",
            "creationTimestamp":"2018-05-11T09:18:24Z",
            "labels":{
                "app":"newjdoslog",
                "group":"ht03",
                "system":"logsystem"
            }
        }
    },
    "timestamp":"2018-05-11T09:18:24Z"
}
```

```
{
    "type":"ADDED",
    "object":{
        "kind":"Pod",
        "metadata":{
            "name":"6e7d5fbb-2e8d-4340-8437-23c72f7bcb8d",
            "namespace":"capjdos",
            "selfLink":"/api/v1/namespaces/capjdos/pods/6e7d5fbb-2e8d-4340-8437-23c72f7bcb8d",
            "uid":"44ecaf13-571f-11e8-9dec-f898efe7ce8c",
            "resourceVersion":"2905653097",
            "creationTimestamp":"2018-05-14T02:34:04Z",
            "labels":{
                "app":"3056",
                "app_id":"3056",
                "env":"HT_CMO",
                "group":"ht03",
                "novaname":"3056-1526265234352-2003843108-f09ac252",
                "novauuid":"6e7d5fbb-2e8d-4340-8437-23c72f7bcb8d",
                "region":"ht03",
                "res.lvm":"true",
                "system":"capjdos"
            },
            "annotations":{
                "containerEnvHash":"e047303e",
                "containerResourceHash":"47ada25c",
                "containerVolumeMountsHash":"7853f5cb",
                "kubernetes.io/config.seen":"2018-08-22T20:25:10.883601119+08:00",
                "kubernetes.io/config.source":"api",
                "novaflavor":"c24caed4-5b06-41a8-9fdb-88f0a08ccaae",
                "novaimage":"19954b7f-dc09-44bf-a643-82ffb5d14b0c",
                "novazone":"HT_CMO"
            }
        },
        "spec":{
            "containers":[
                {
                    "name":"container",
                    "image":"is.ht1.n.jd.local/cap2/base:search_bd_work_jdk6_20160519.171214",
                    "resources":{
                        "limits":{
                            "cpu":"8",
                            "memory":"16Gi"
                        },
                        "requests":{
                            "cpu":"2",
                            "memory":"10752Mi"
                        }
                    },
                    "terminationMessagePath":"/dev/termination-log",
                    "terminationMessagePolicy":"File",
                    "imagePullPolicy":"IfNotPresent",
                    "stdin":true,
                    "tty":true
                }
            ],
```

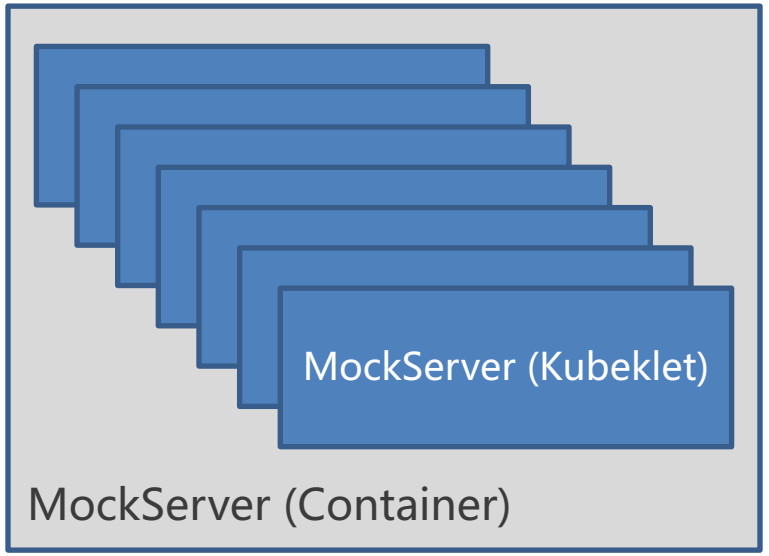
Node Simulation

MockNode

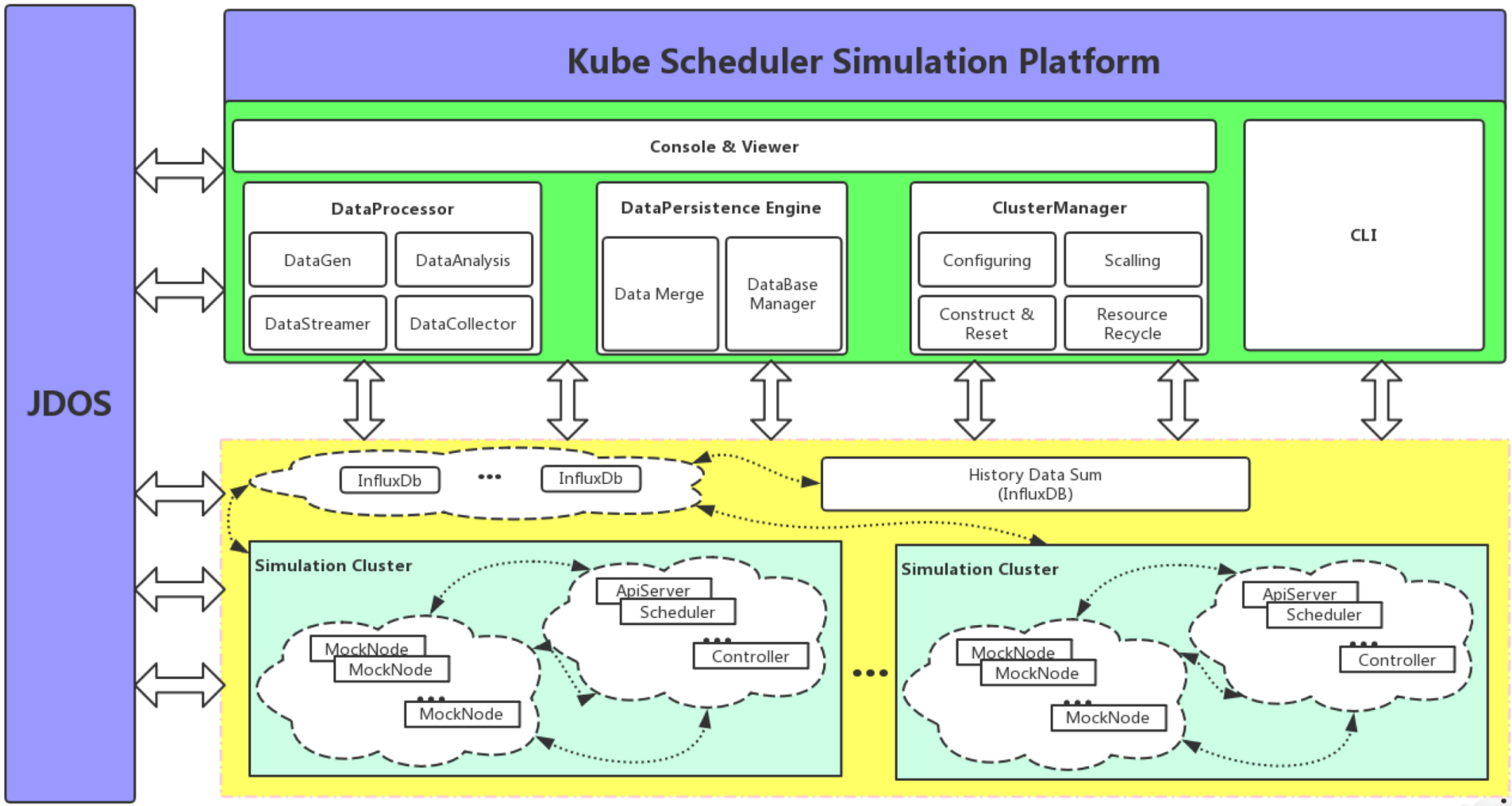
- Simulate Kubelet in K8s
- Resource template: customized memory, CPU disk…
- Watch pod scheduling events and update statuses
- Report resource usage

MockServer

- Run in containers
- Deployed and managed by K8s
- **Simulate 100+ MockNodes per MockServer (8 cores and 16GB memory)**



Implementation and Deployment



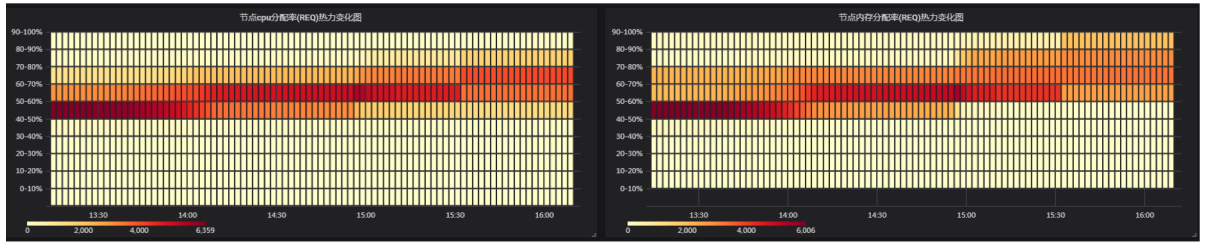
Monitoring and Visualization

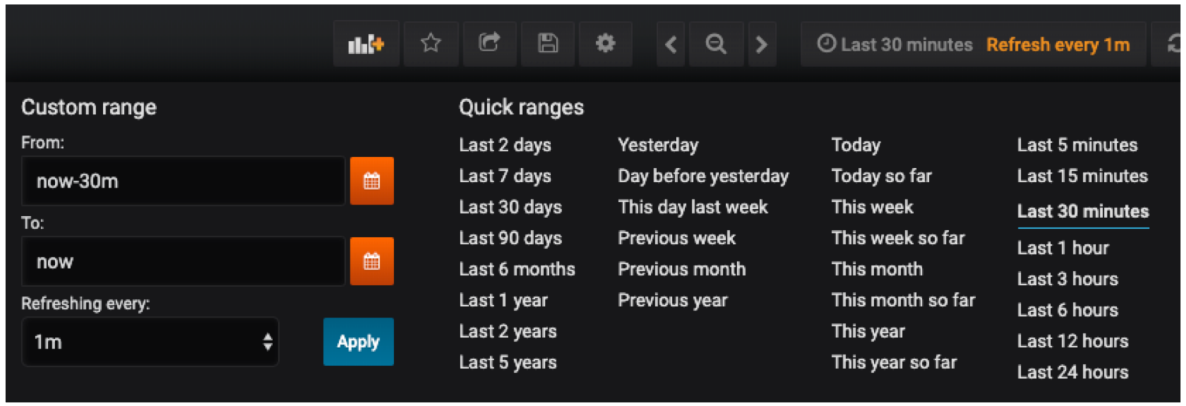
Metrics

- APIServer performance
- Scheduling performance and quality: node CPU, memory, disk allocation rate

Reporting and visualization

- Heat map, TS charts, statistics at multiple granularities
- Import and export





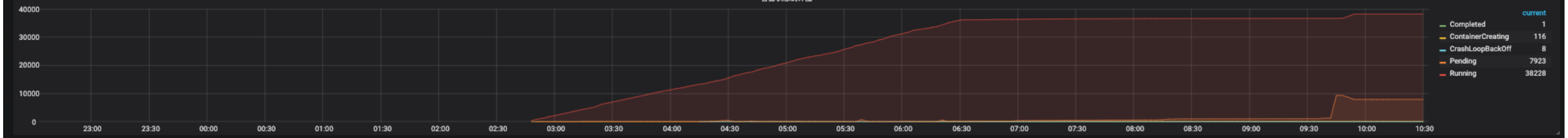
仿真平台 调度质量监控

Last 12 hours Refresh every 1m

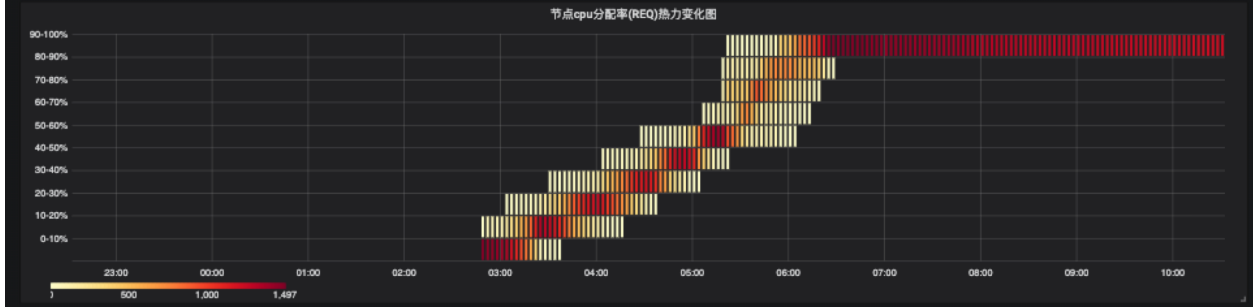
集群概况

region	物理机器数目	容器数目	总CPU(核)	总内存	已分配CPU(核)	已分配内存	CPU分配率(Limit)	内存分配率(Limit)	CPU分配率(REQ)	内存分配率(REQ)
simulation0	1500	38353	90000	366 TiB	89436	324 TiB	263.61%	126.99%	99.37%	88.35%

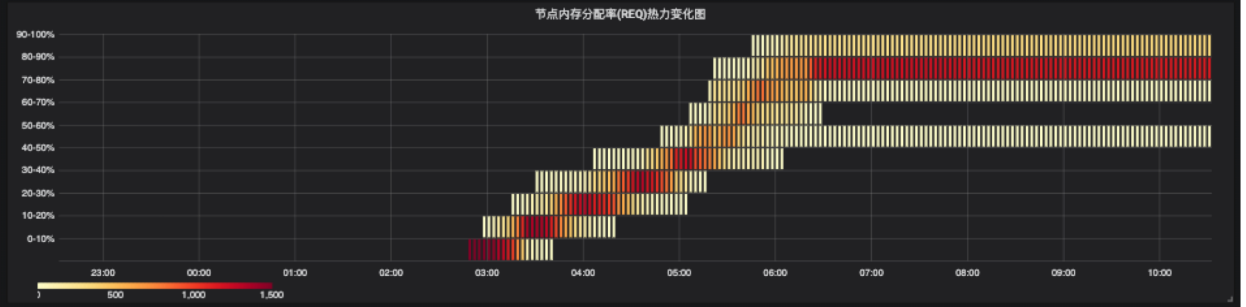
容器状态统计图



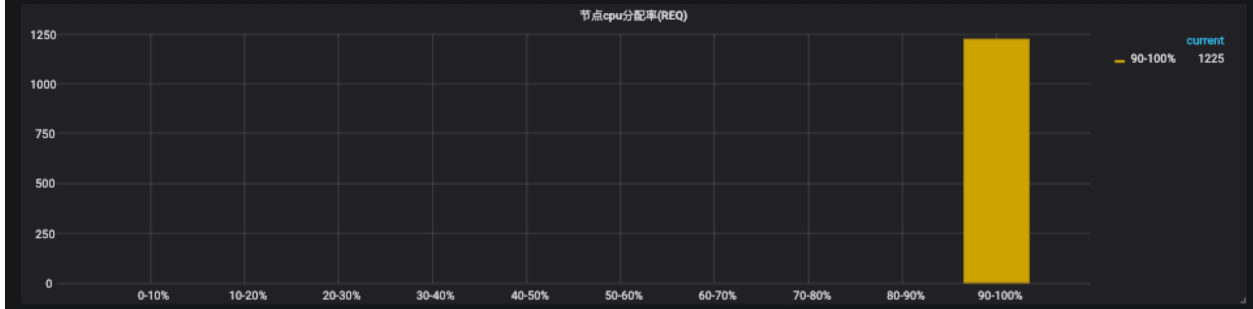
节点cpu分配率(REQ)热力变化图



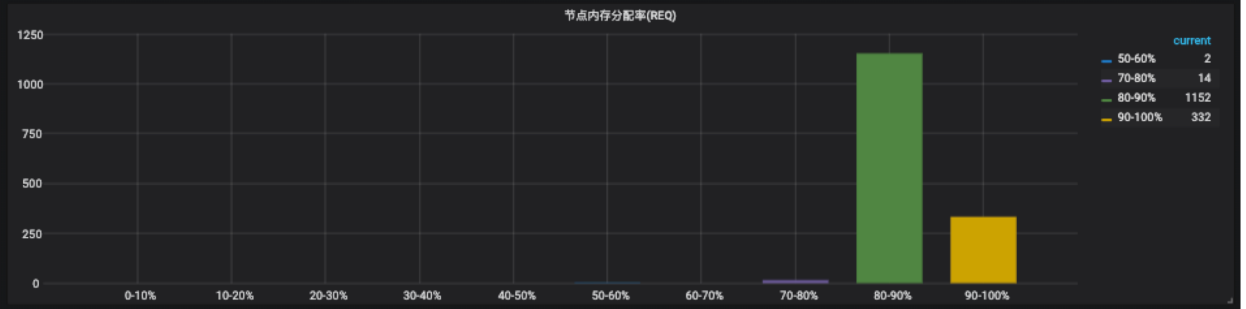
节点内存分配率(REQ)热力变化图



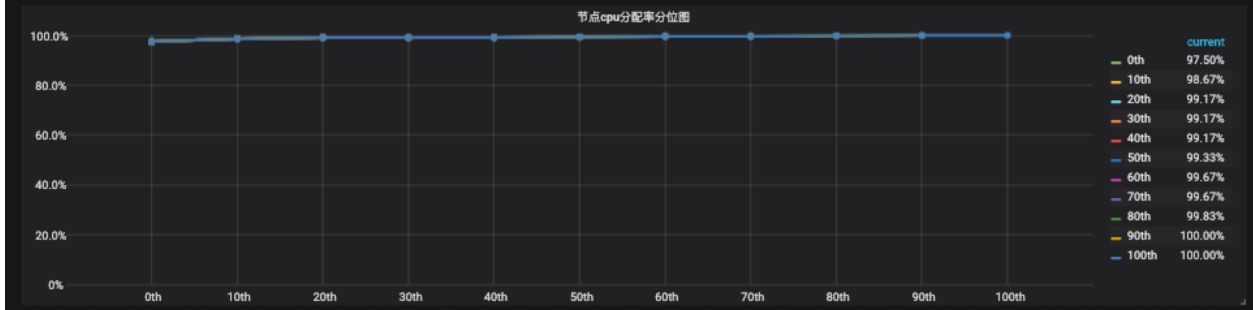
节点cpu分配率(REQ)



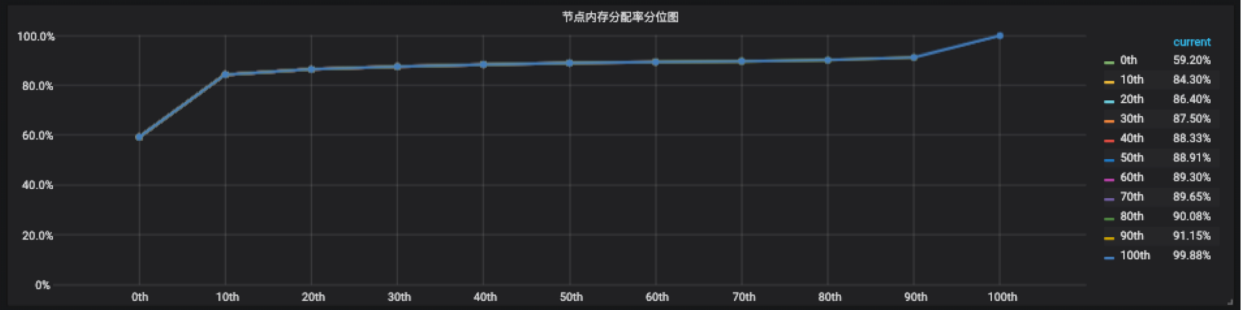
节点内存分配率(REQ)



节点cpu分配率分位图

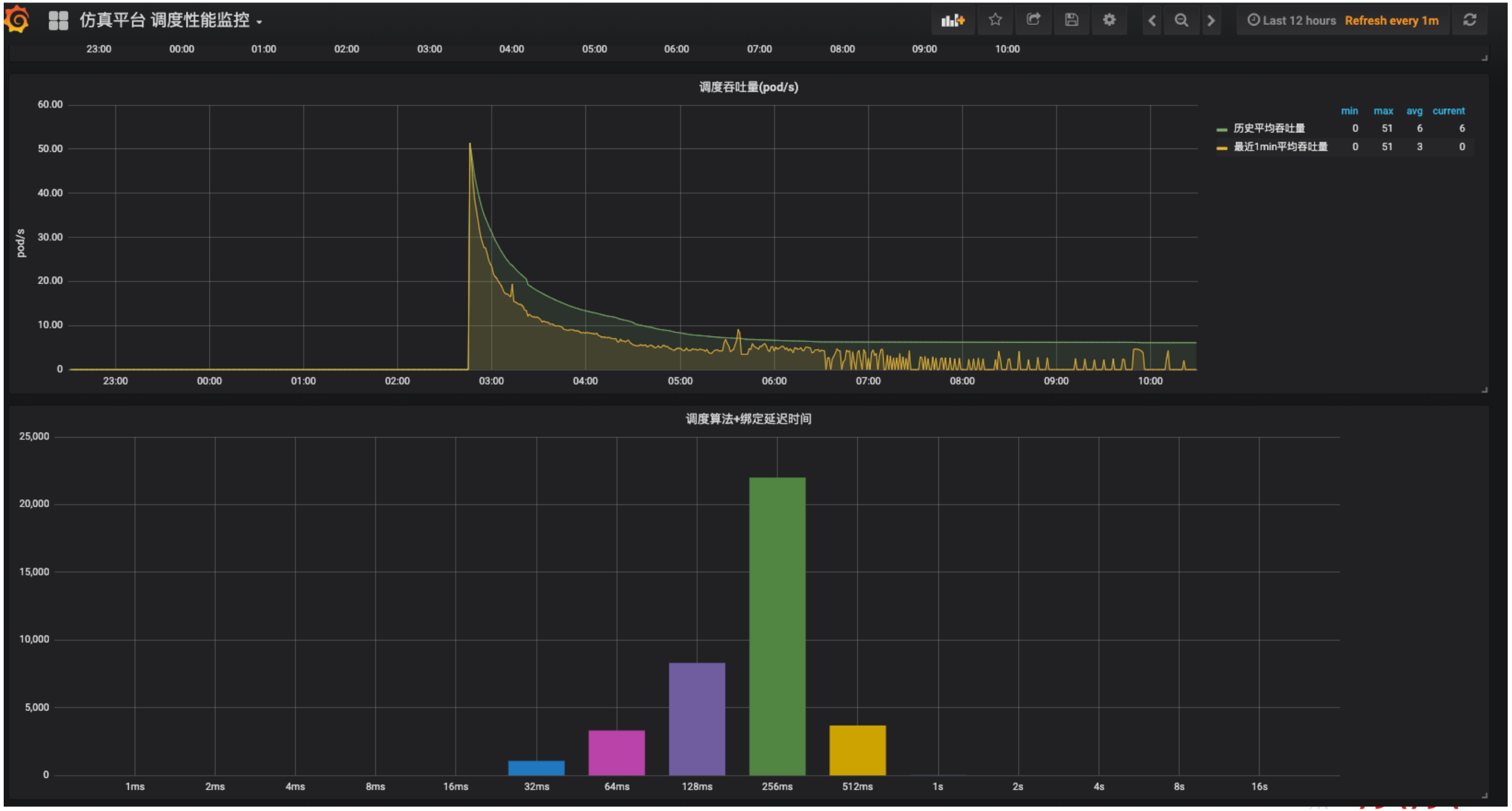


节点内存分配率分位图




仿真平台 调度性能监控
Last 12 hours
Refresh every 1m
predicate延迟时间
17,500
15,000
12,500
10,000
7,500
5,000
2,500
0
1ms
2ms
4ms
8ms
16ms
32ms
64ms
128ms
256ms
512ms
1s
2s
4s
8s
16s
current
1ms 0
2ms 818
4ms 6,021
8ms 2,202
16ms 605
32ms 2,210
64ms 6,520
128ms 15,251
256ms 4,575
512ms 150
1s 1
2s 0
4s 0
8s 0
16s 0
priority延迟时间
12,000
10,000
8,000
6,000
4,000
2,000
0
1ms
2ms
4ms
8ms
16ms
32ms
64ms
128ms
256ms
512ms
1s
2s
4s
8s
16s
current
1ms 3
2ms 399
4ms 630
8ms 1,471
16ms 2,358
32ms 4,893
64ms 5,407
128ms 11,455
256ms 11,728
512ms 9
1s 0
2s 0
4s 0
8s 0
16s 0


仿真平台 调度性能监控
Last 12 hours
Refresh every 1m
调度算法延迟时间
25,000
20,000
15,000
10,000
5,000
0
1ms
2ms
4ms
8ms
16ms
32ms
64ms
128ms
256ms
512ms
1s
2s
4s
8s
16s
current
1ms 0
2ms 0
4ms 0
8ms 0
16ms 73
32ms 1,368
64ms 3,308
128ms 8,384
256ms 21,886
512ms 3,319
1s 15
2s 0
4s 0
8s 0
16s 0
绑定binding延迟时间
40,000
35,000
30,000
25,000
20,000
15,000
10,000
5,000
0
1ms
2ms
4ms
8ms
16ms
32ms
64ms
128ms
256ms
512ms
1s
2s
4s
8s
16s
current
1ms 0
2ms 0
4ms 34,433
8ms 3,456
16ms 257
32ms 49
64ms 55
128ms 67
256ms 33
512ms 3
1s 0
2s 0
4s 0
8s 0
16s 0


仿真平台 调度性能监控
Last 12 hours
Refresh every 1m
23:00
00:00
01:00
02:00
03:00
04:00
05:00
06:00
07:00
08:00
09:00
10:00
调度吞吐量(pod/s)
pod/s
60.00
50.00
40.00
30.00
20.00
10.00
0
min
max
avg
current
历史平均吞吐量
0
51
6
6
最近1min平均吞吐量
0
51
3
0
调度算法+绑定延迟时间
25,000
20,000
15,000
10,000
5,000
0
1ms
2ms
4ms
8ms
16ms
32ms
64ms
128ms
256ms
512ms
1s
2s
4s
8s
16s


仿真平台 调度性能监控
Last 12 hours
Refresh every 1m
调度算法+绑定延迟时间热力图
16 s
8 s
4 s
2 s
1 s
512 ms
256 ms
128 ms
64 ms
32 ms
16 ms
8 ms
4 ms
2 ms
1 ms
23:00
00:00
01:00
02:00
03:00
04:00
05:00
06:00
07:00
08:00
09:00
10:00
.0
0.5
1.0
调度延迟时间
300 ms
250 ms
200 ms
150 ms
100 ms
50 ms
0 ns
min
max
avg
current
binding 50分位延时 3 ms 4 ms 4 ms 4 ms
binding 90分位延时 4 ms 4 ms 4 ms 4 ms
binding 99分位延时 4 ms 11 ms 9 ms 10 ms
binding 调度平均延时 3 ms 4 ms 4 ms 4 ms
predicate_evaluation 50分位延时 3 ms 92 ms 62 ms 4 ms
predicate_evaluation 90分位延时 15 ms 173 ms 122 ms 168 ms
predicate_evaluation 99分位延时 17 ms 278 ms 171 ms 271 ms
predicate_evaluation 调度平均延时 14 ms 78 ms 63 ms 67 ms
priority_evaluation 50分位延时 1 ms 177 ms 78 ms 177 ms
priority_evaluation 90分位延时 1 ms 216 ms 134 ms 216 ms
priority_evaluation 99分位延时 1 ms 242 ms 163 ms 242 ms
priority_evaluation 调度平均延时 1 ms 91 ms 63 ms 91 ms
调度吞吐量(pod/s)

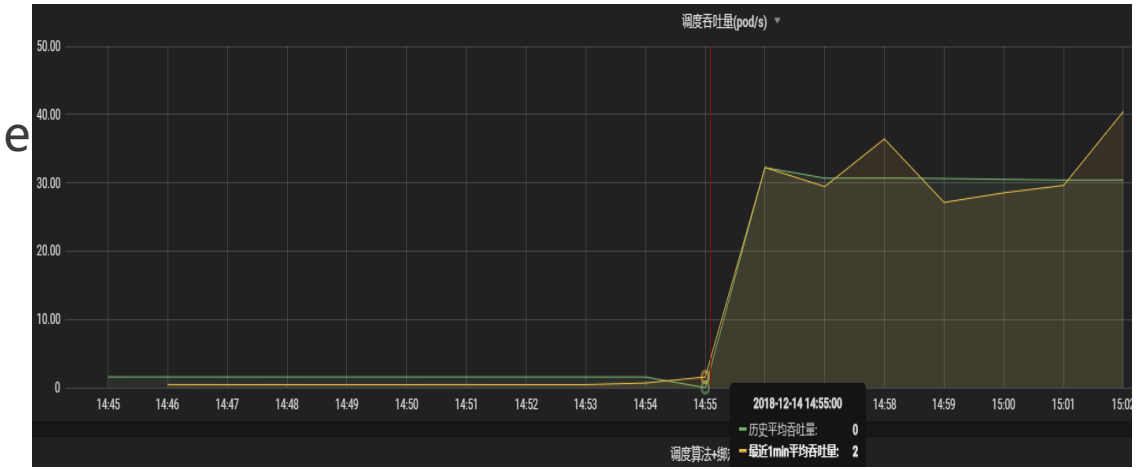
京东

Outline

- Background and Introduction
- JoySim Design and Implementation
- Use Cases
- Conclusion
- Demo

Case Study: APIServer Evaluation and Optimization

- **Simulation testbed**
  - Event traces from multiple K8s clusters
  - 1800 simulation containers, 6 Mocknodes (60 CPU cores, 300GB) per container
  - **Simulate 11,400 nodes, 200,000+ container**
- **APIServer scalability evaluation**
  - 6 APIServers to support a 10,000-node cluster
- **ETCD optimization**
  - Two etcd clusters: one for Pod resources and one for other resource objects
  - No read/write/storage bottlenecks
- **Scheduler optimization**
  - QPS: 2→ 40
  - Quality: utilization 2X



京东

Case Study: Scheduler Performance Optimization

Simulation testbed

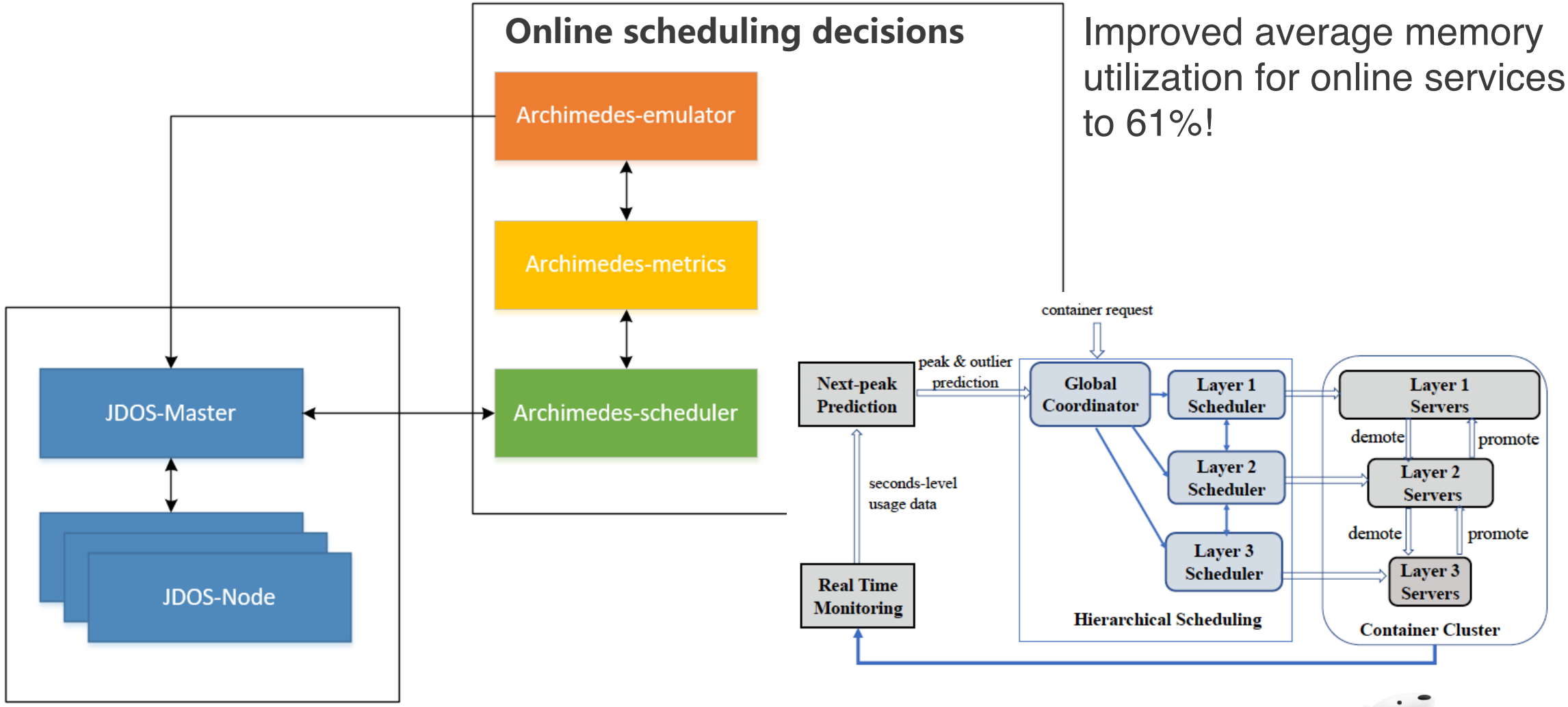
- 800 nodes，8124 running pods, schedule a job of 500 pods

Batch job scheduling

Scheduler	Execution Time	Scheduling Mechanisms
K8s Scheduler	16 sec	Per pod scheduling and binding
Volcano Batch Scheduler	4.72 sec	Per pod scheduling and group binding
JDOS Batch Scheduler v1	0.27 sec	Group scheduling and binding
JDOS Batch Scheduler v2	0.15 sec	Adaptive group scheduling and binding

京东

Case Study: Online Scheduling



Improved average memory utilization for online services to 61%!

京东

Summary

- **JoySim**: a simulator for large scale Kubernetes cluster simulation.
- Applications for scalability and performance evaluation and optimization of large scale Kubernetes cluster at JD.com.
- Planning to work with CNCF SIG-Scheduling for an open source release.

Thank you!

京东

Contact

Qichao Lu luqichao@jd.com

京东