Distributed Global Scheduling in Datacenters

UNIVERSITY OF CAMBRIDGE

Systems Research Group (SRG)
Department of Computer Science
University of Cambridge

Smita Vijayakumar
First Year Ph.D. Student
sv440@cst.cam.ac.uk

Evangelia Kalyvianaki
Ph.D. Supervisor
ek264@cst.cam.ac.uk

Anil Madhavapeddy
Ph.D. Supervisor
avsm2@cst.cam.ac.uk

State of the Art Scheduling

Underutilised Datacenter Resources

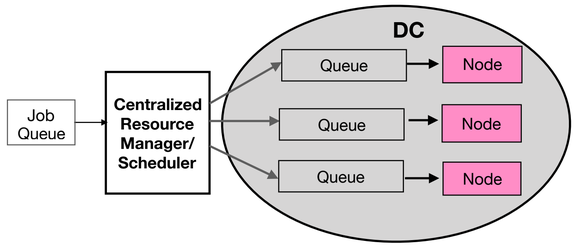
Azure[1]
- 60% VMs have <= **20%** CPU usage!

Alibaba[2]
- Average server CPU **50%**
- Memory <= **60%**

Underutilisation is expensive![3]

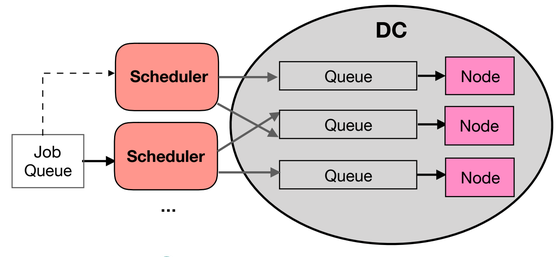
[1] *[Resource Central, SOSP,'17]*
[2] *[https://github.com/alibaba/clusterdata]*
[3] *[Scalable system scheduling for HPC and big data, JPDC,17]*

Centralised



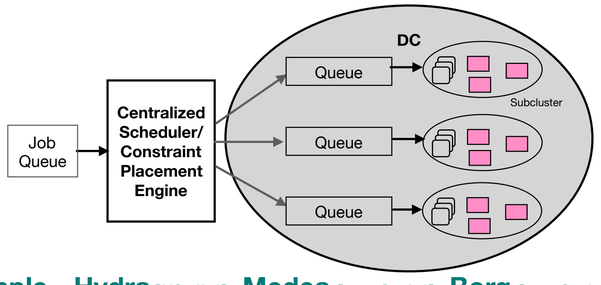
Examples - Mesos *[NSDI'11]*, **Yarn, Apollo** *[OSDI'14]*
- ☑ **Global resource view**
- ✖ **Scheduler can be a bottleneck**
- ✖ **Delayed, high volumes of resource updates**

Decentralised



Example - Sparrow *[NSDI'14]*
- ☑ **Fast and simple**
- ✖ **Unsuitable for long running jobs**
- ✖ **Not globally optimal**

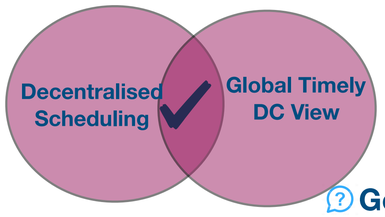
Hybrid



Example - Hydra *[NSDI'19]*, **Medea** *[EuroSys'18]*, **Borg** [EuroSys'15]
- ☑ **Policy-driven job/task placement**
- ☑ **Less node information traffic**
- ✖ **Centralised or decentralised components**

Proposed Direction

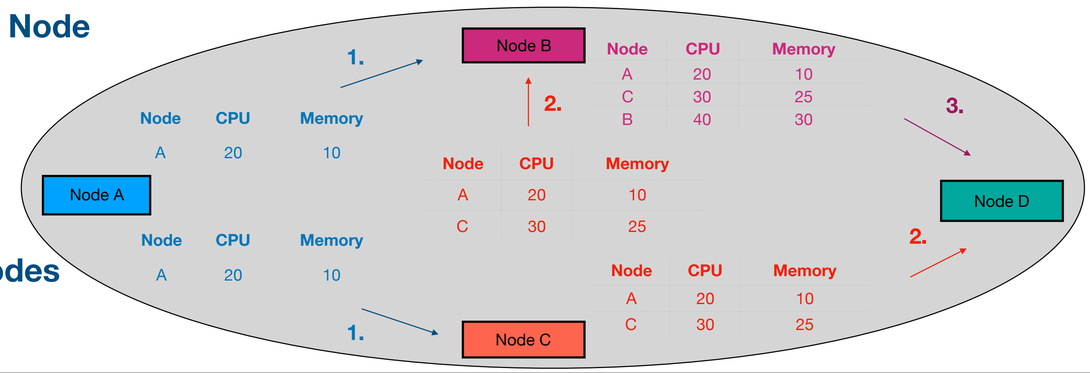
Global Scheduling at Node Level



Challenges
- ? **Good for long and short jobs**
- ? **Volume and frequency of updates**
- ? **Time from local to global view**

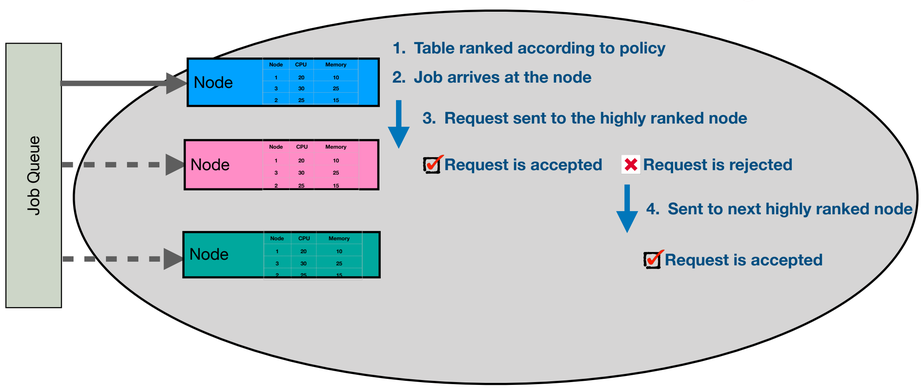
Up-to-Date Global View at each Node

Inspired by routing protocols
- ☑ **BGP, OSPF, ...**
- ☑ **Resource data propagation**
- ☑ **Global view convergence**
- ☑ **Same ranking policy across nodes**



Scheduling Using "Up-to-Date" Global View

Intra-DC Load Balanced Scheduling



Challenges
- ? **Collision avoidance**
- ? **Minimise inter-DC traffic**
- ? **Minimise scheduling time**

Inter-DC Load Balanced Scheduling