# CLOUD COMPUTING CONCEPTS

with Indranil Gupta (Indy)

## SCHEDULING

Lecture C

## DOMINANT-RESOURCE FAIR SCHEDULING

# Challenge

- What about scheduling VMs in a cloud (cluster)?
- Jobs may have multi-resource requirements
  - Job 1's tasks: 2 CPUs, 8 GB
  - Job 2's tasks: 6 CPUs, 2 GB
- How do you schedule these jobs in a "fair" manner?
- That is, how many tasks of each job do you allow the system to run concurrently?
- What does fairness even mean?

# Dominant Resource Fairness (DRF)

- Proposed by researchers from U. California Berkeley
- Proposes notion of fairness across jobs with multi-resource requirements
- They showed that DRF is
  - Fair for multi-tenant systems
  - Strategy-proof: tenant can’t benefit by lying
  - Envy-free: tenant can’t envy another tenant’s allocations

# Where is DRF Useful?

- DRF is
  - Usable in scheduling VMs in a cluster
  - Usable in scheduling Hadoop in a cluster
- DRF used in Mesos, an OS intended for cloud environments
- DRF-like strategies also used some cloud computing company's distributed OS's

# How DRF Works

- Our example
  - Job 1's tasks: 2 CPUs, 8 GB

    => Job 1's *resource vector* = <2 CPUs, 8 GB>
  - Job 2's tasks: 6 CPUs, 2 GB

    => Job 2's *resource vector* = <6 CPUs, 2 GB>
- Consider a cloud with <18 CPUs, 36 GB RAM>

# How DRF Works (2)

- Our example
  - Job 1's tasks: 2 CPUs, 8 GB

    => Job 1's *resource vector* = <2 CPUs, 8 GB>
  - Job 2's tasks: 6 CPUs, 2 GB

    => Job 2's *resource vector* = <6 CPUs, 2 GB>
- Consider a cloud with <18 CPUs, 36 GB RAM>
- Each Job 1's task consumes % of total CPUs = 2/18 = **1/9**
- Each Job 1's task consumes % of total RAM = 8/36 = **2/9**
- 1/9 < 2/9
  - => *Job 1's dominant resource is RAM*, i.e., Job 1 is more memory-intensive than it is CPU-intensive

# How DRF Works (3)

- Our example
  - Job 1's tasks: 2 CPUs, 8 GB

    => Job 1's *resource vector* = <2 CPUs, 8 GB>
  - Job 2's tasks: 6 CPUs, 2 GB

    => Job 2's *resource vector* = <6 CPUs, 2 GB>
- Consider a cloud with <18 CPUs, 36 GB RAM>
- Each Job 2's task consumes % of total CPUs = 6/18 = 6/18
- Each Job 2's task consumes % of total RAM = 2/36 = 1/18
- 6/18 > 1/18
  - => *Job 2's dominant resource is CPU*, i.e., Job 1 is more CPU-intensive than it is memory-intensive

# DRF Fairness

- For a given job, the % of its dominant resource type that it gets cluster-wide, is the same for all jobs
  - Job 1's % of RAM = Job 2's % of CPU
- Can be written as linear equations, and solved

# DRF Solution, For our Example

- DRF Ensures
  - Job 1's % of RAM = Job 2's % of CPU
- Solution for our example:
  - Job 1 gets 3 tasks each with <2 CPUs, 8 GB>
  - Job 2 gets 2 tasks each with <6 CPUs, 2 GB>
  - Job 1's % of RAM

= Number of tasks * RAM per task / Total cluster RAM

= 3*8/36 = 2/3

  - Job 2's % of CPU

= Number of tasks * CPU per task / Total cluster CPUs

= 2*6/18 = 2/3

# Other DRF Details

- DRF generalizes to multiple jobs
- DRF also generalizes to more than 2 resource types
  - CPU, RAM, Network, Disk, etc.
- DRF ensures that each job gets a fair share of that type of resource which the job desires the most
  - Hence fairness

# Summary: Scheduling

- Scheduling very important problem in cloud computing
  - Limited resources, lots of jobs requiring access to these jobs
- Single-processor scheduling
  - FIFO/FCFS, STF, Priority, Round-Robin
- Hadoop scheduling
  - Capacity scheduler, Fair scheduler
- Dominant-Resources Fairness