DSDS: DATA STORE DRIVEN APPLICATION SCHEDULING

Frezewd Lemma Tena, ***Christof Fetzer***
TU Dresden, Germany

TECHNISCHE UNIVERSITÄT DRESDEN

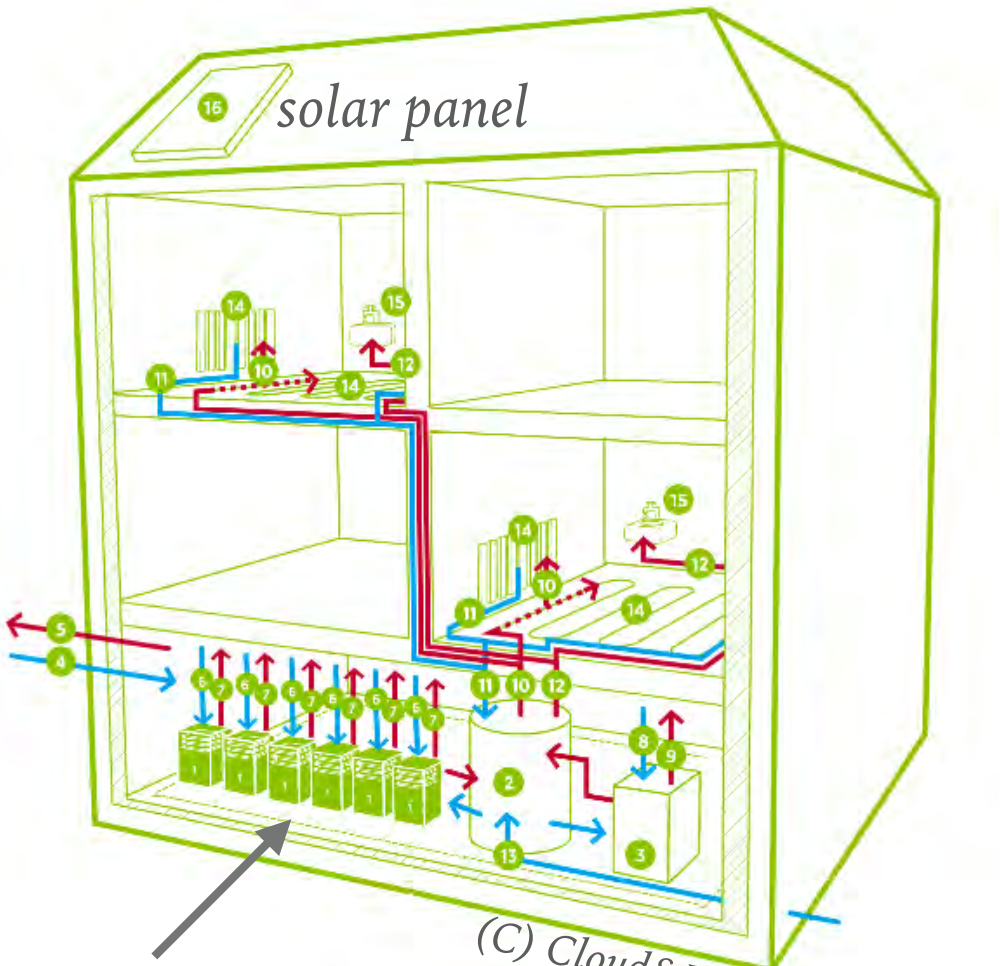
MOTIVATION

- **Context: reduce end user perceived latency**
  - move computing closer to end user
  - **how to build an edge cloud?**

- **Problem**: cost of building and operating an edge cloud

- Objective: **Reduce TCO of an edge cloud**
  - electricity costs
  - cost of hosting and maintaining computing infrastructure

SYSTEM MODEL



micro-cloud consisting of compute racks or containers

- **Distributed edge cloud**
  - connected to heating system
  - each micro-cloud provides compute & storage resources
- **Cost of computing depends on**
  - need for heat / hot water (of building)
  - local electricity cost:
    - local solar power

OBSERVATION 1: WE NEED TO INCREASE UTILIZATION

- Infrastructure permits to
  - reduce user perceived latency
- To reduce TCO, micro-clouds need to support more app domains:
  - compute heavy jobs (protein folding, …)
  - store backups
  - store replicas of data
  - data mining jobs (accessing one of the replicas)
  - …

OBSERVATION 2: CUT DOWN POWER COSTS

- **To reduce the electricity costs, we can**
  - use lower-cost solar power
  - sell the „waste heat“ of the computers
  - computers hibernate to reduce power consumption

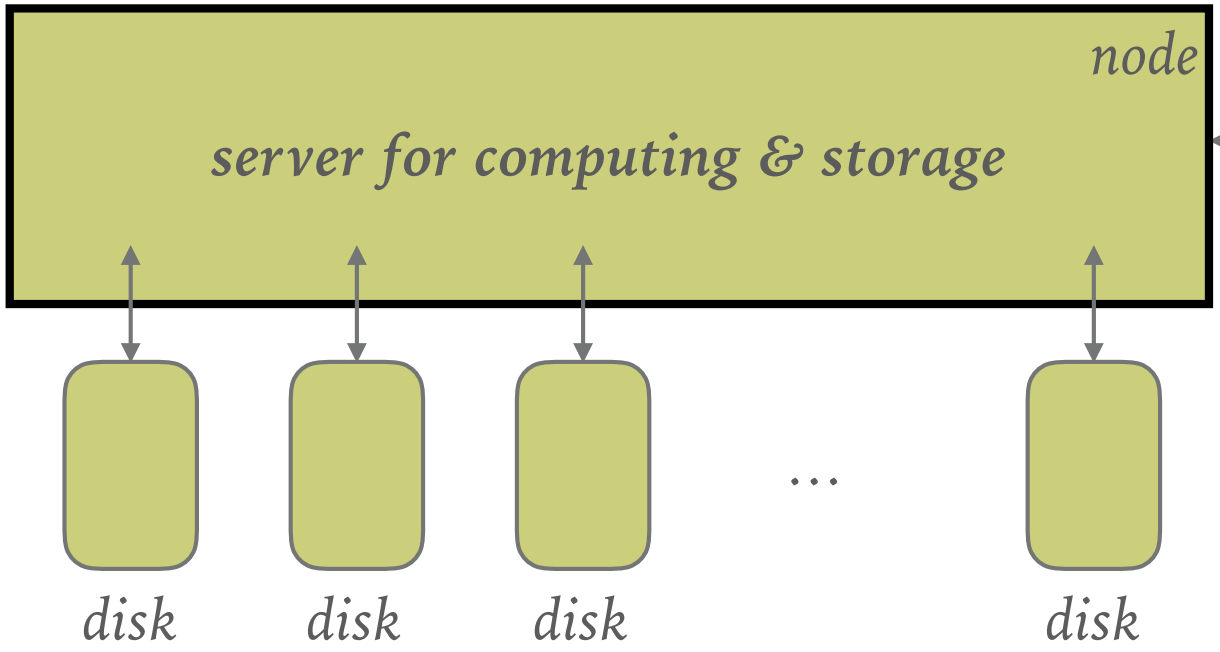
- **Difficult scheduling problem!**

PROBLEM ADDRESSED

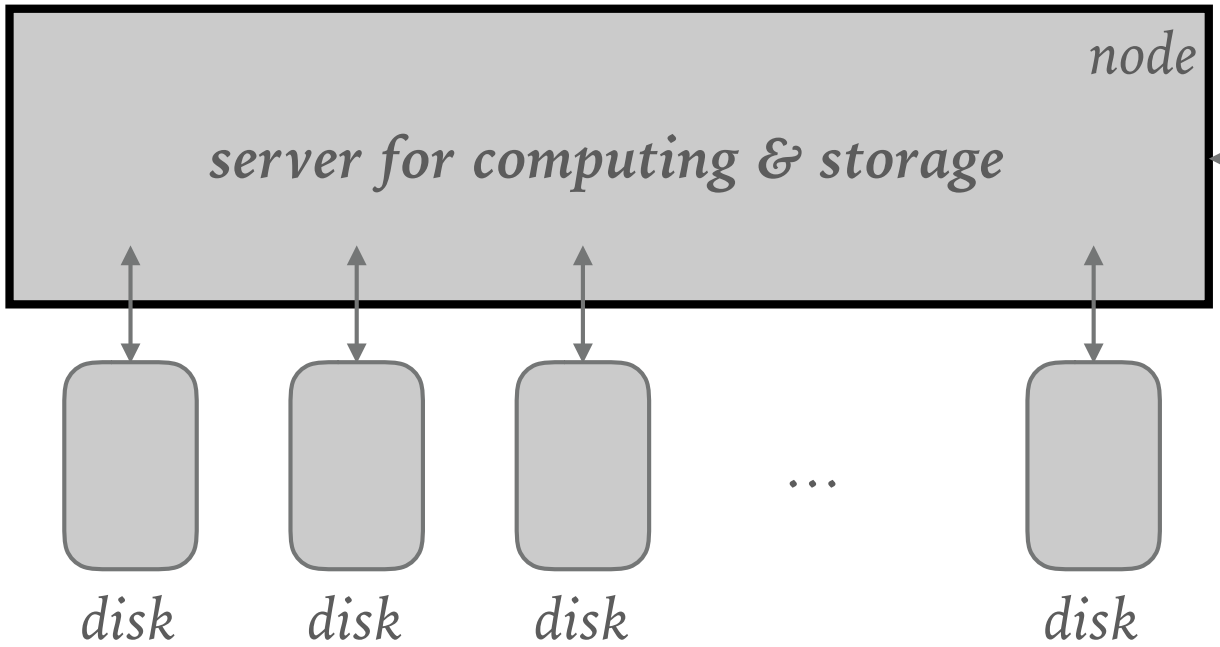
- In which microcloud should we run a compute job?
  - e.g., data mining jobs access

- **Naive approach**:
  - at microcloud that has the lowest effective electricity costs
- **Problem**:
  - data too large to move to another microcloud before running compute job

NODE ARCHITECTURE (COST-EFFECTIVE PLATFORM)

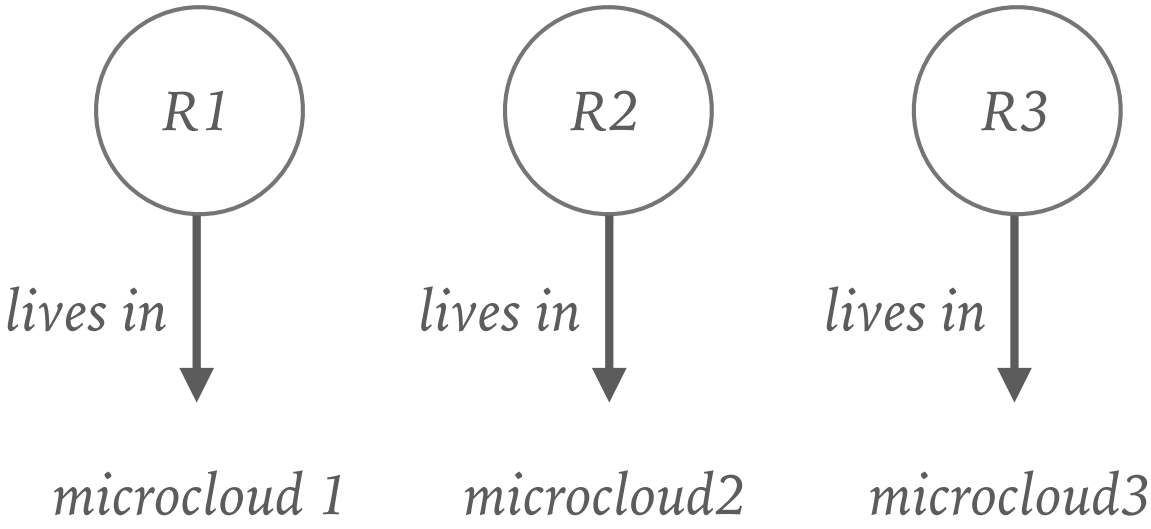


not energy-proportional



***Example**: access to one disk requires server to be in „active state“*

REPLICATION OF DATA



typically, we keep

3 replicas

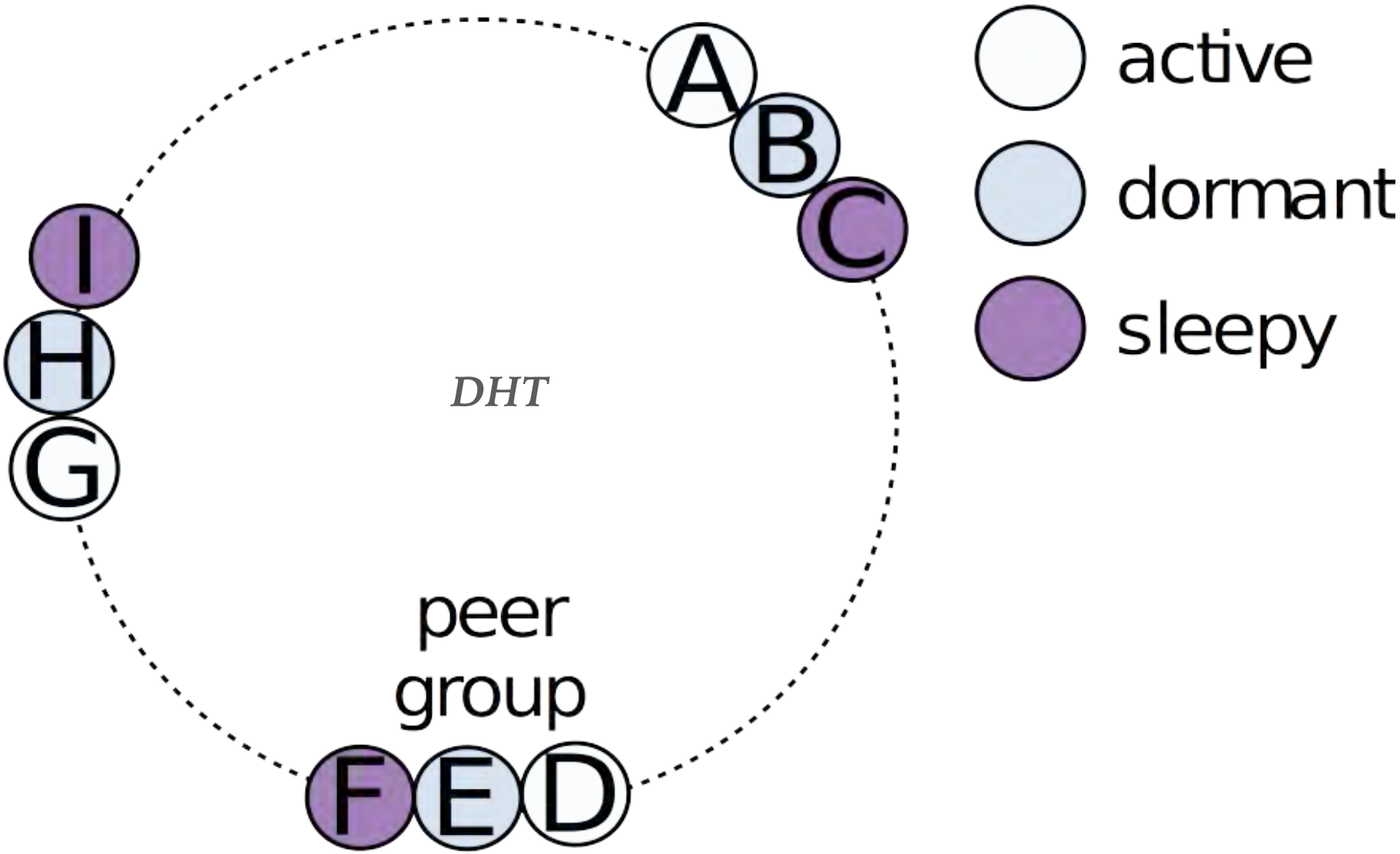
Write(W): 3
Read(R): 1
satisfies: $R + W > N$

For writing: all three disks/servers need to be active

For reading: one disk/server needs to be active

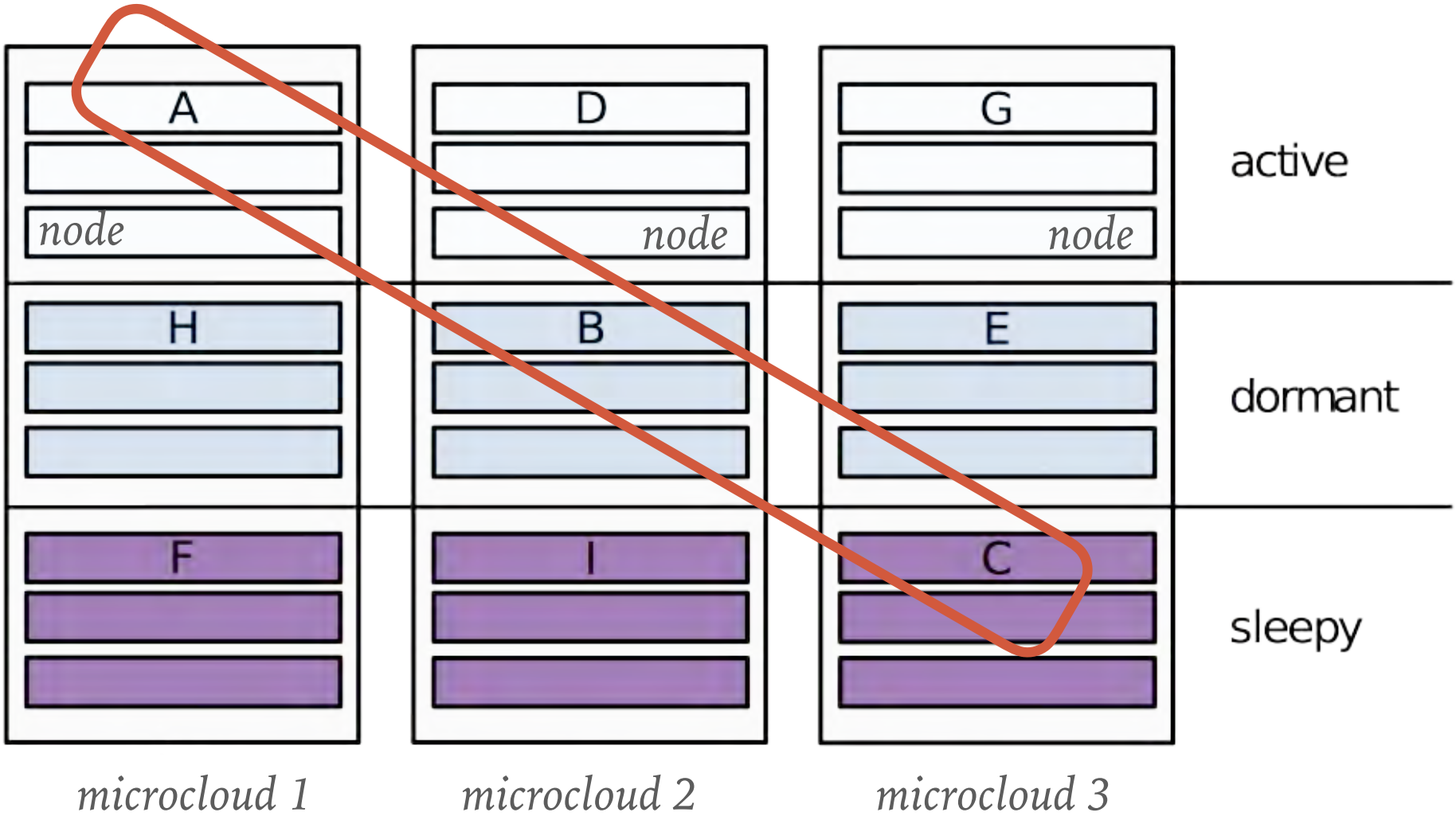
***Problem**: this might require to keep all servers & disks in „active state“*

POWERCASS ARCHITECTURE



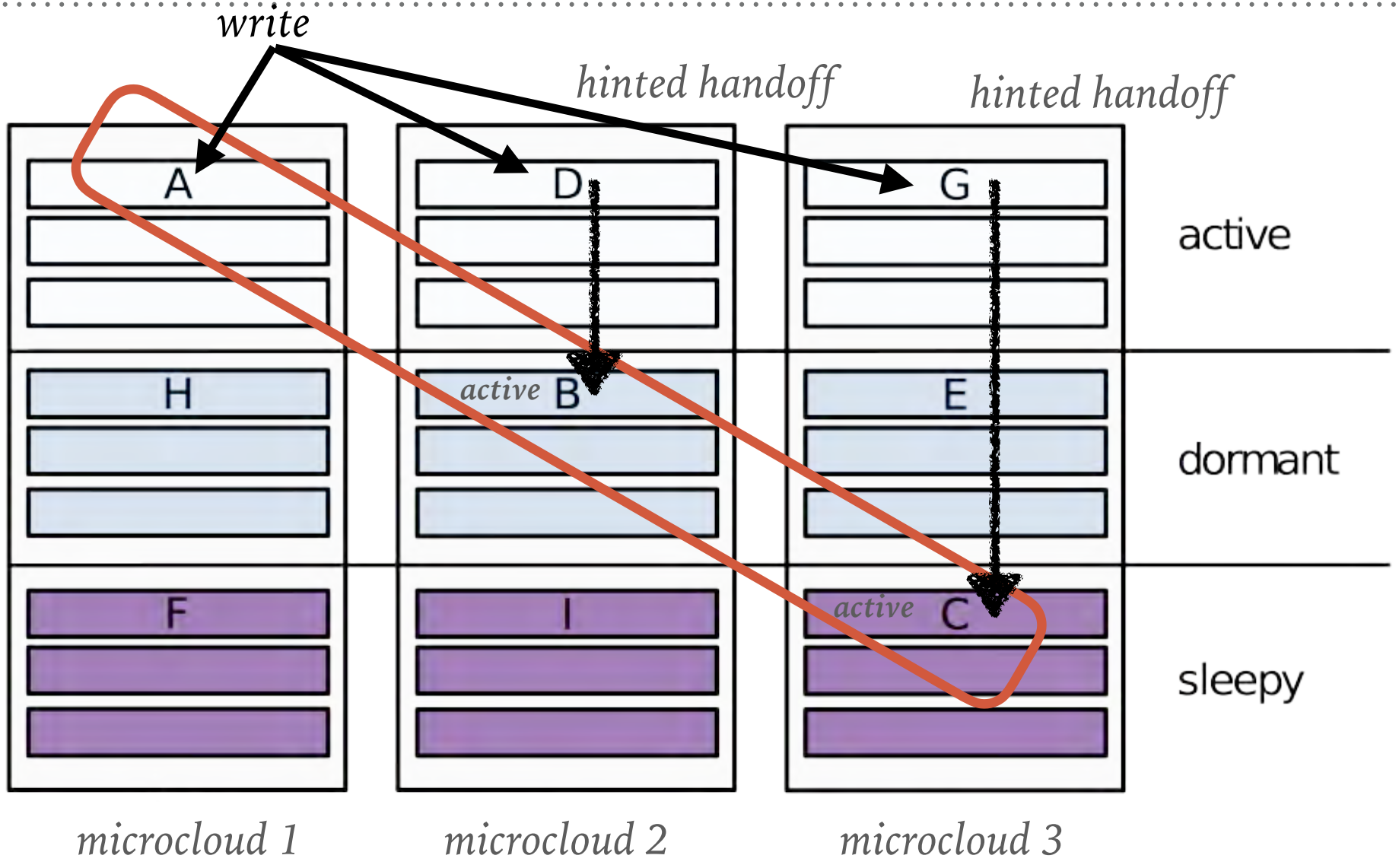
Approach: dormant and sleep peers can go into „hibernation mode“

REPLICATION ACROSS MICRO CLOUDS



We can always read data from active node

WRITING TO SWITCHED-OFF NODES

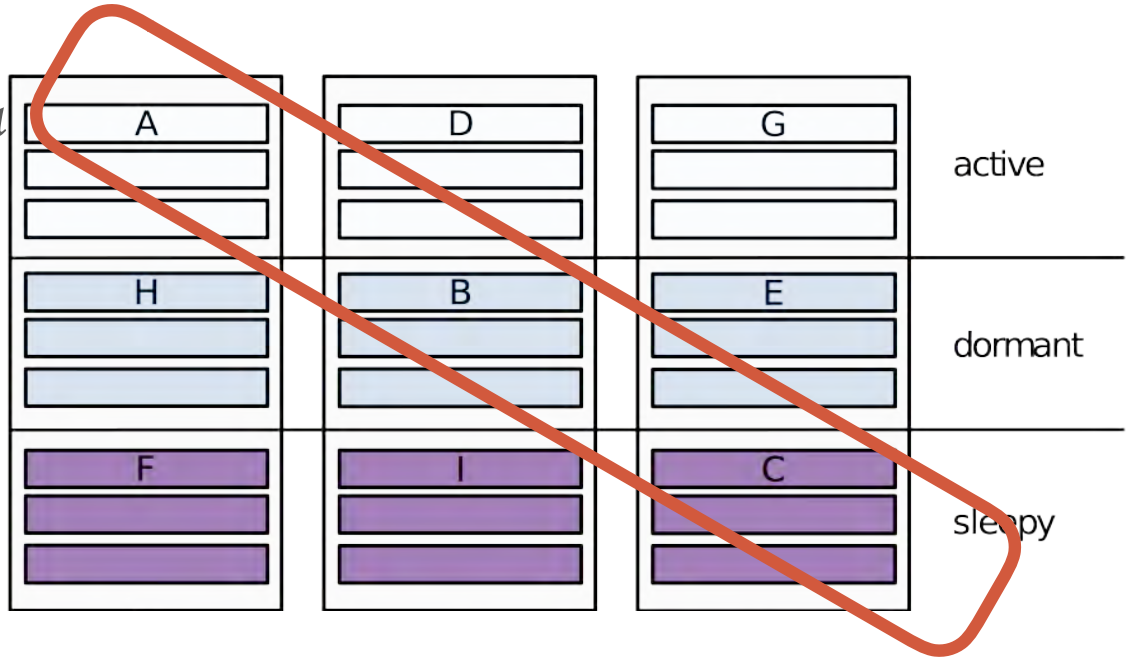


Can always write: hinted-handoff to using active nodes

APPLICATION ASSUMPTIONS

- We assume that we
  - know what data will be accessed by an application
  - know if a job is „short“ or „long“ running

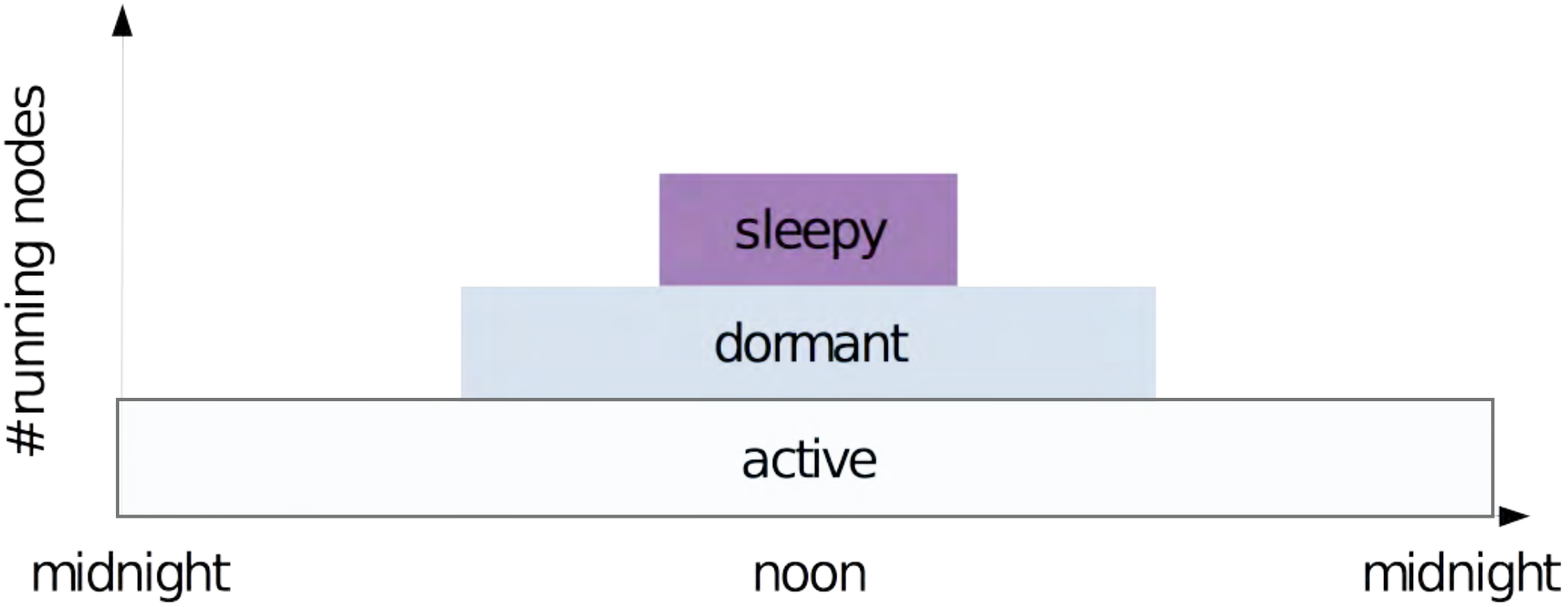
application App's data



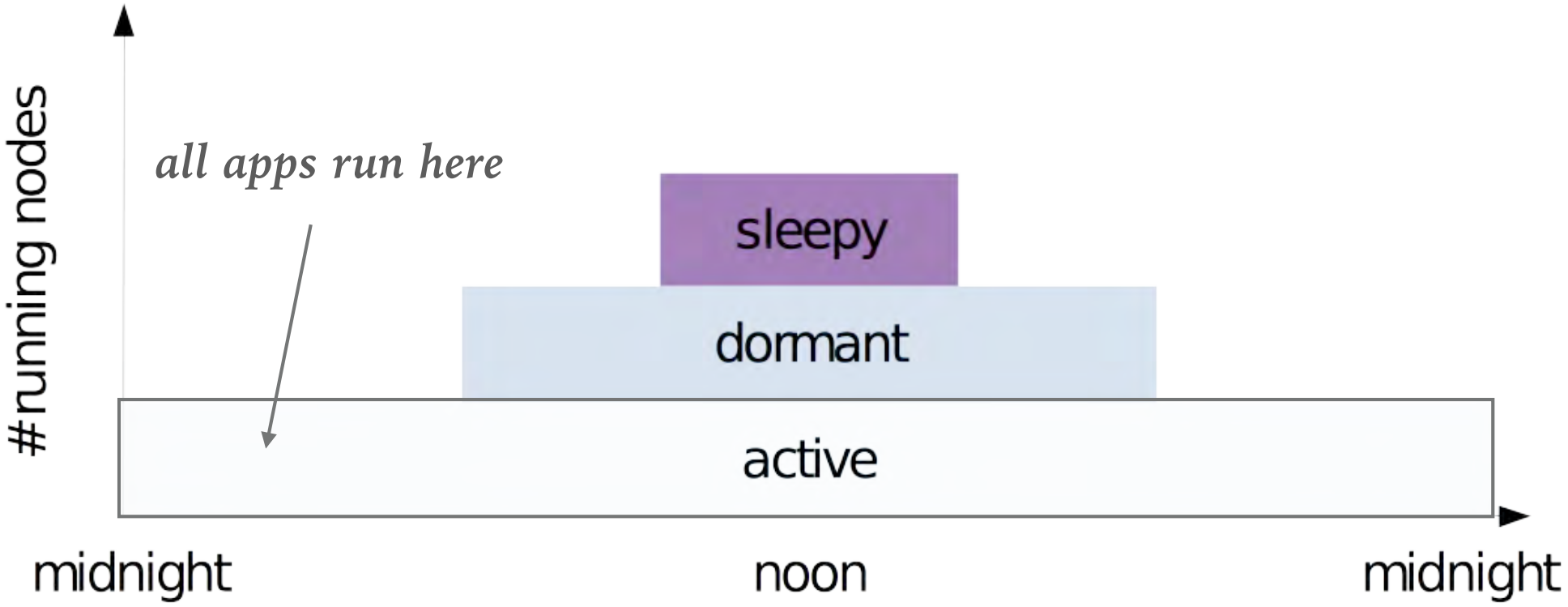
Where should we execute App?

NODES

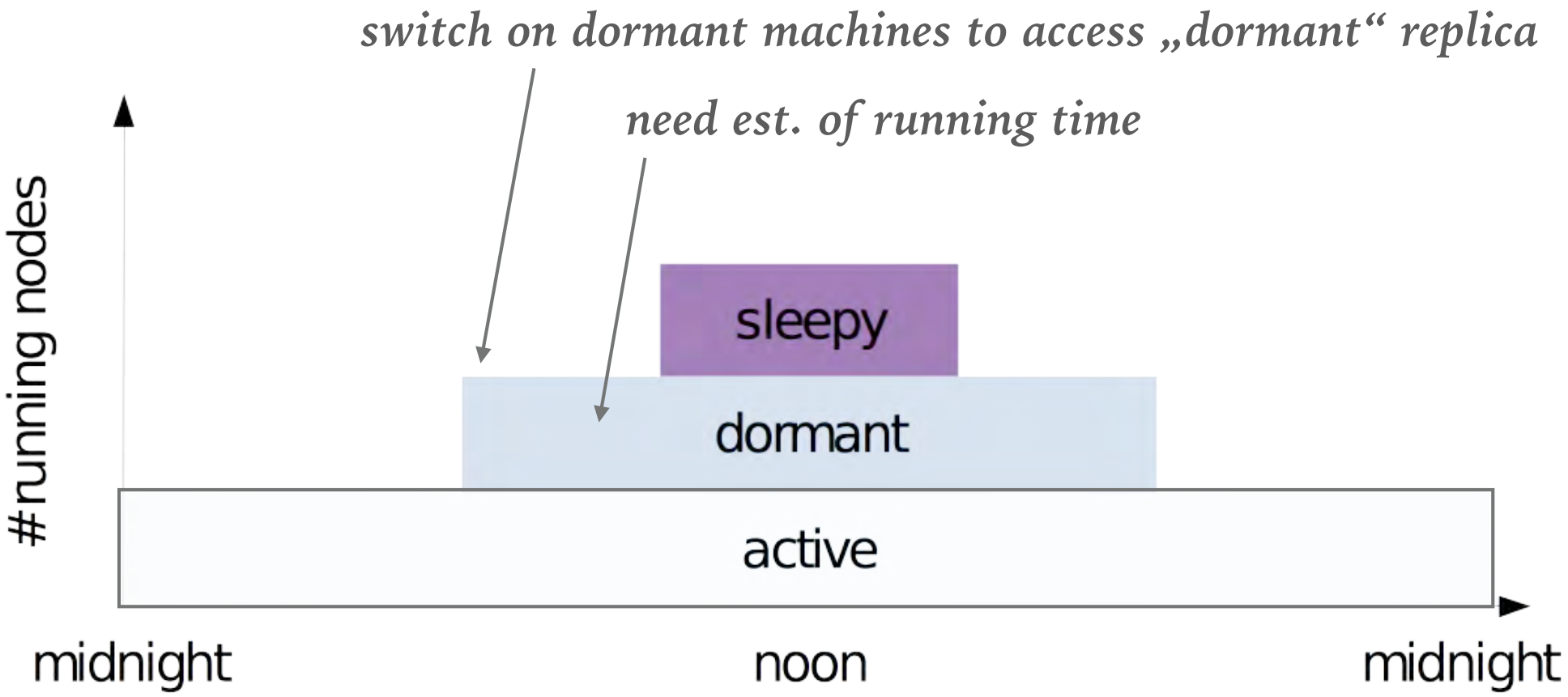
- daily load pattern



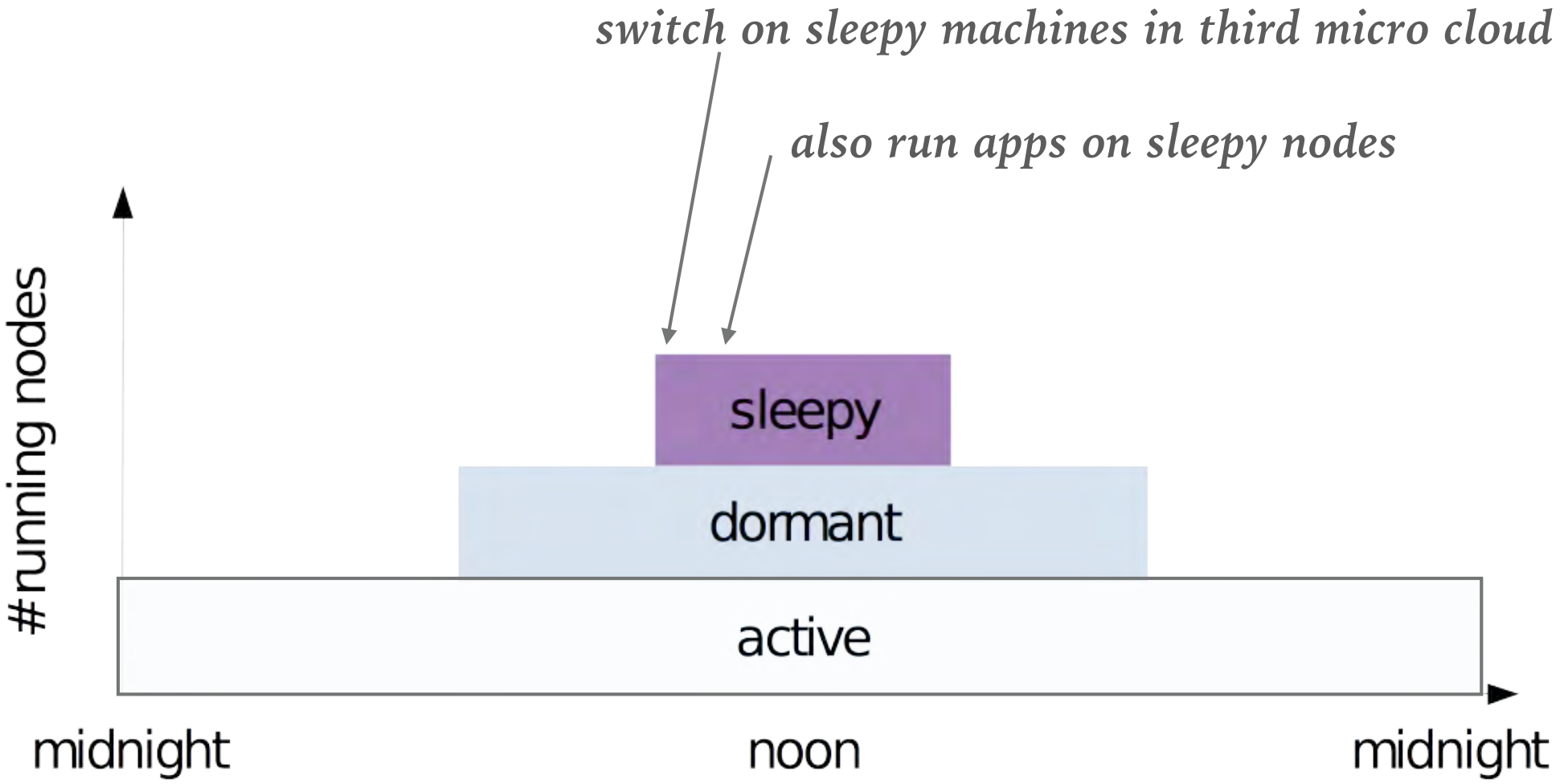
SCHEDULING IDEA: LOW LOAD



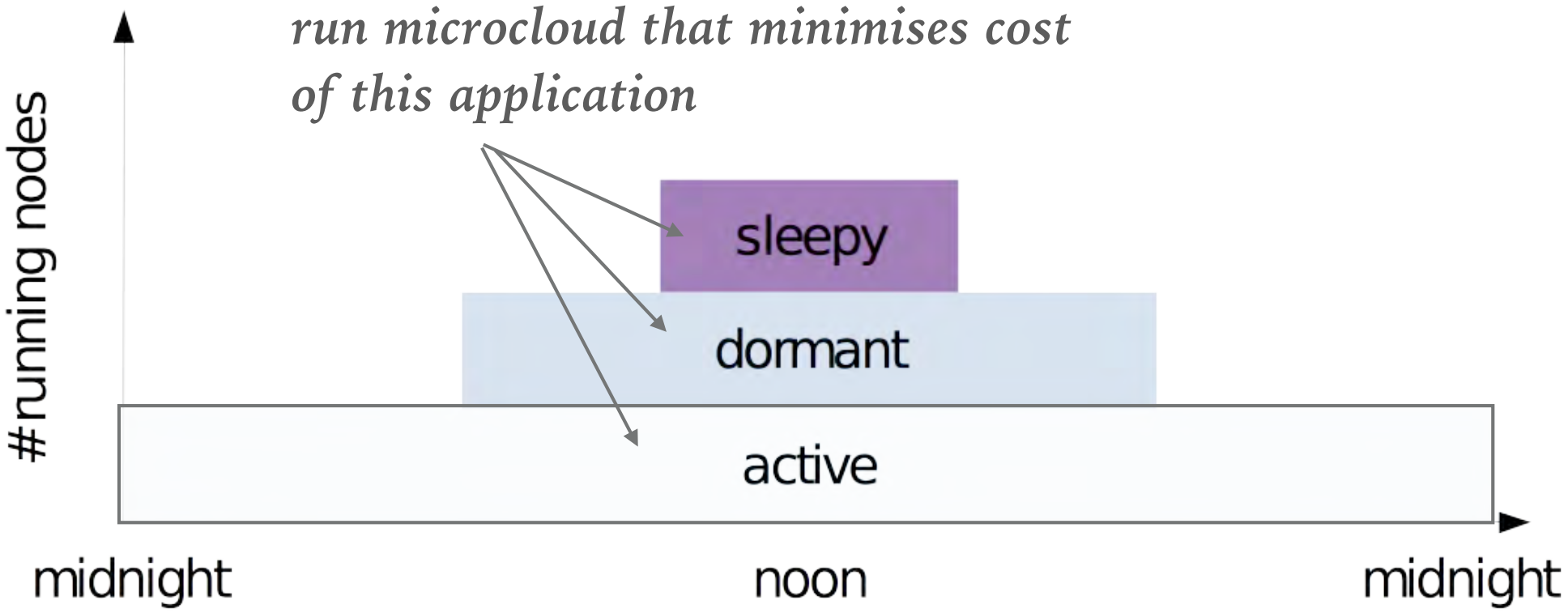
SCHEDULING IDEA: MEDIUM LOAD



SCHEDULING IDEA: HIGH LOAD



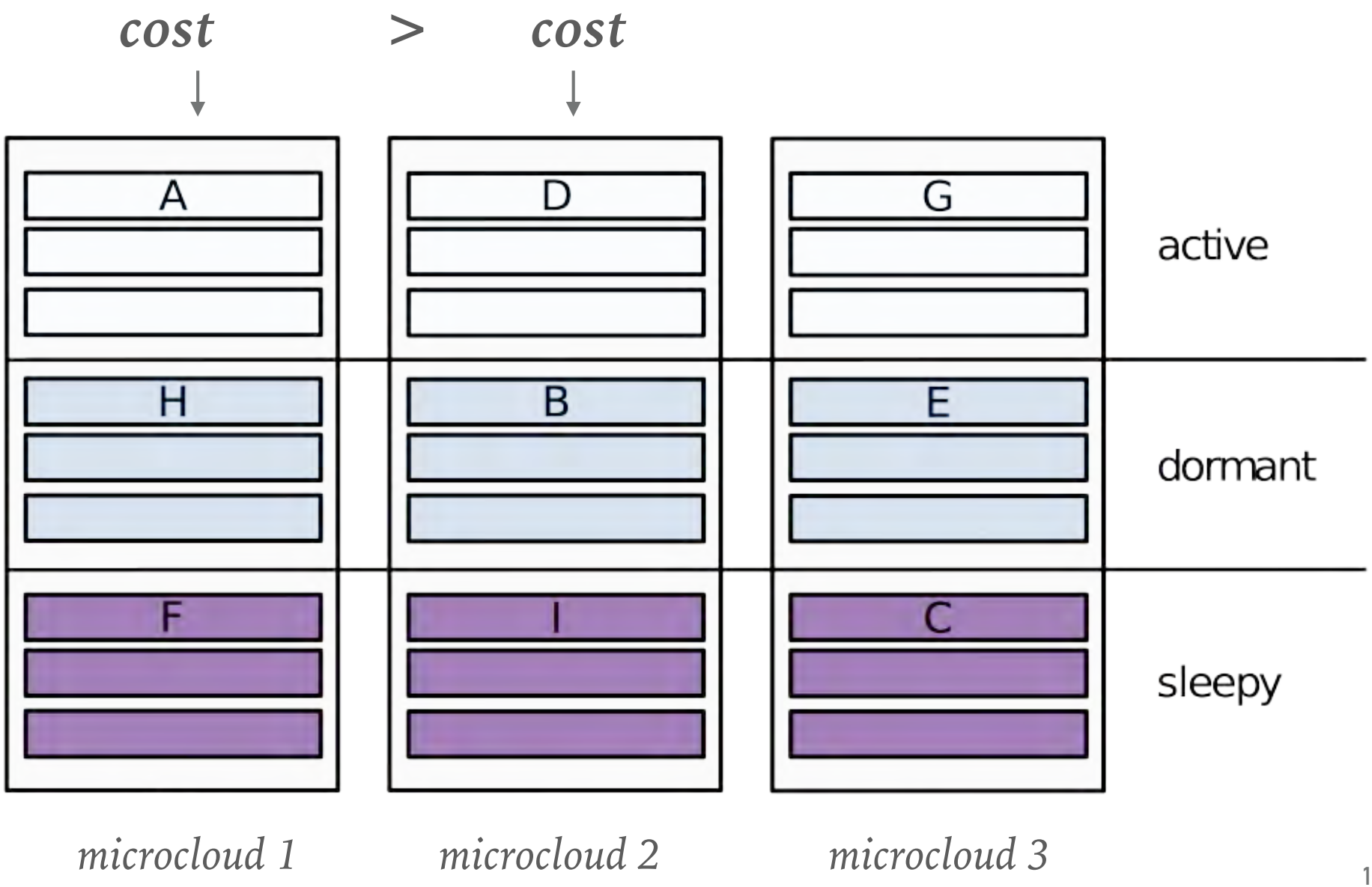
SCHEDULING IDEA: HIGH LOAD



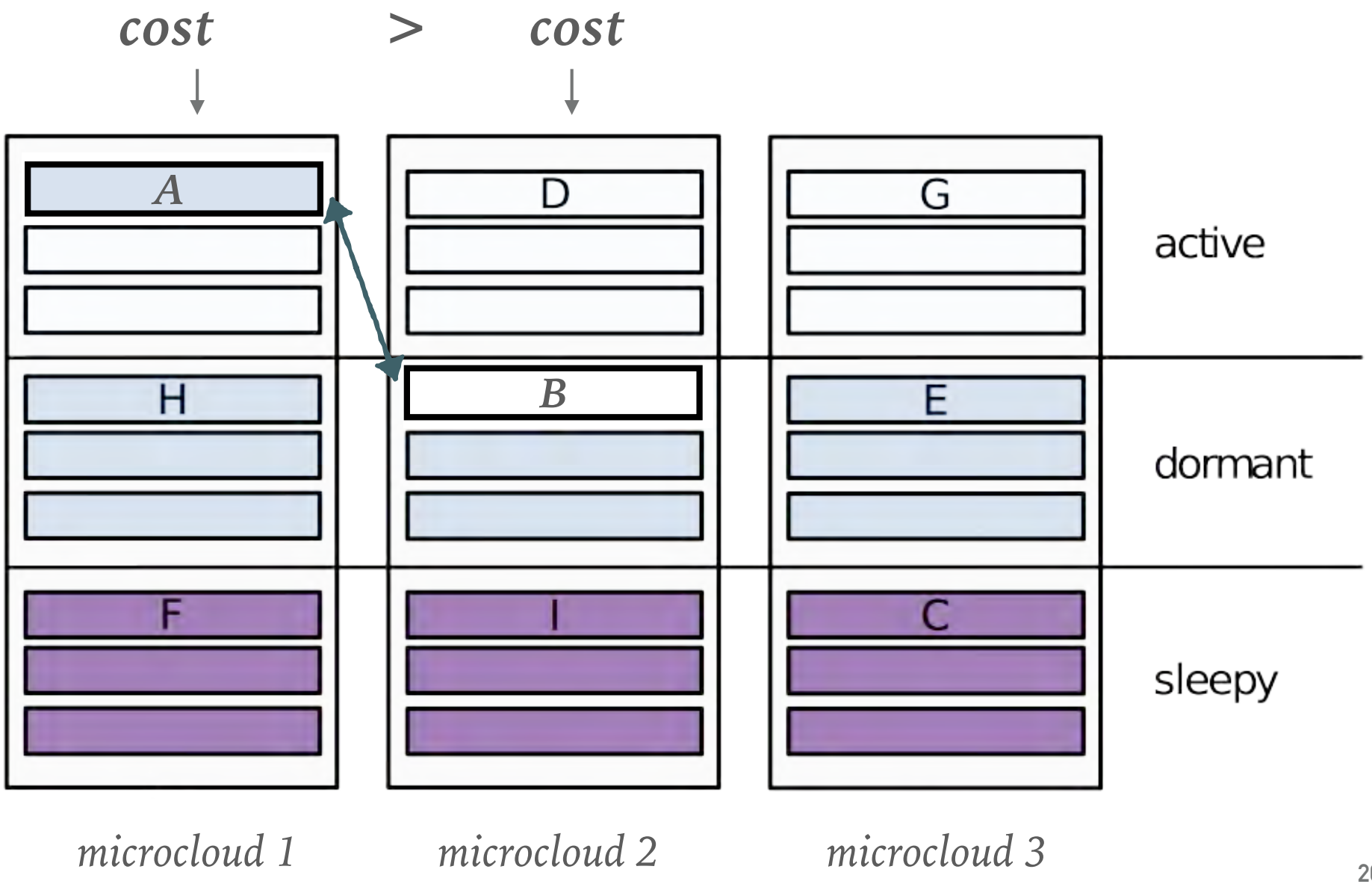
NEXT STEPS: SWITCH ROLES OF NODES

- **Problem:**
  - static classification in active / dormant / sleep not optimal
- **Approach:**
  - switch „roles“ of nodes to reduce cost of computation
- **Example:**
  - swap roles of sleepy and dormant nodes at different sites

EXAMPLE


cost
>
cost
A
D
G
active
H
B
E
dormant
F
I
C
sleepy
microcloud 1
microcloud 2
microcloud 3

SWITCH ROLE OF NODES: ACTIVE VS DORMANT



PROBLEMS

- What if nodes **A** and **B** do not store identical content?
  - we might not be able to simply change roles of A and B!

- **How to address this?**
  - keep nodes identical (bad for durability)
  - migrate data locally to different class of node
  - …

CURRENT WORK

- **Address security concerns** (due to limited physical security)
- **Motivation**:
  - we need to keep the data encrypted
  - data mining job needs encryption key - how to keep this secure?
- **Approach**: Docker-Compatible Secure Framework
  - provide secure computation based on Intel SGX (SCONE, OSDI 2016, SGXBounds, EuroSys 2017)

SUMMARY

- **We are working on an edge cloud that combines**
  - energy-efficiency, and
  - low-latency (edge cloud)
- We want to use this edge cloud to
  - store and process data
- Showed: smart scheduling can reduce the cost of computation
- **Current work**:
  - further improve energy-efficiency
  - address security issues