

NodeFinder: Scalable Search over Highly Dynamic Geo-distributed State



Azzam Alsudais
Eric Keller



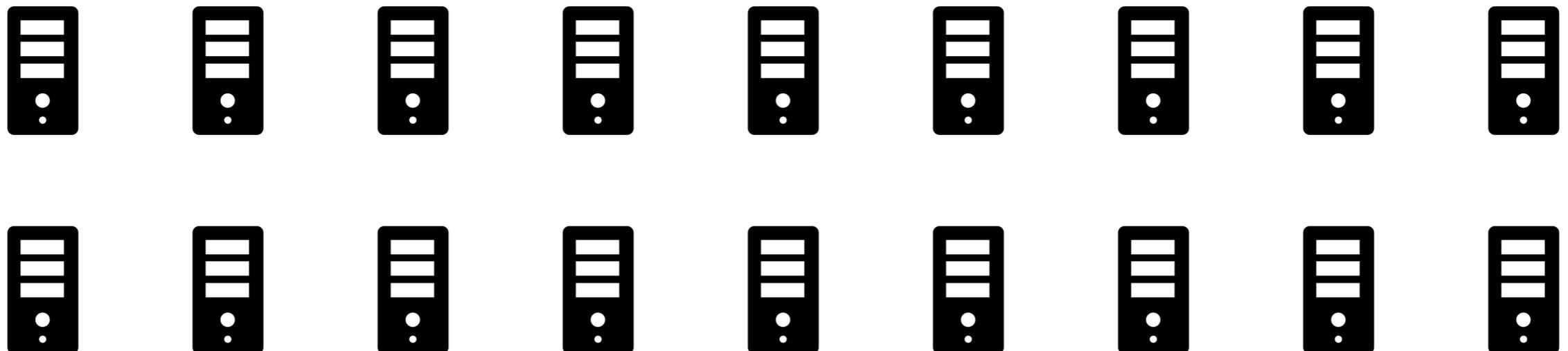
Zhe Huang
Bharath Balasubramanian
Shankaranarayanan Puzhavakath Narayanan
Kaustubh Joshi

USENIX HotCloud'18



Query Server

Nodes



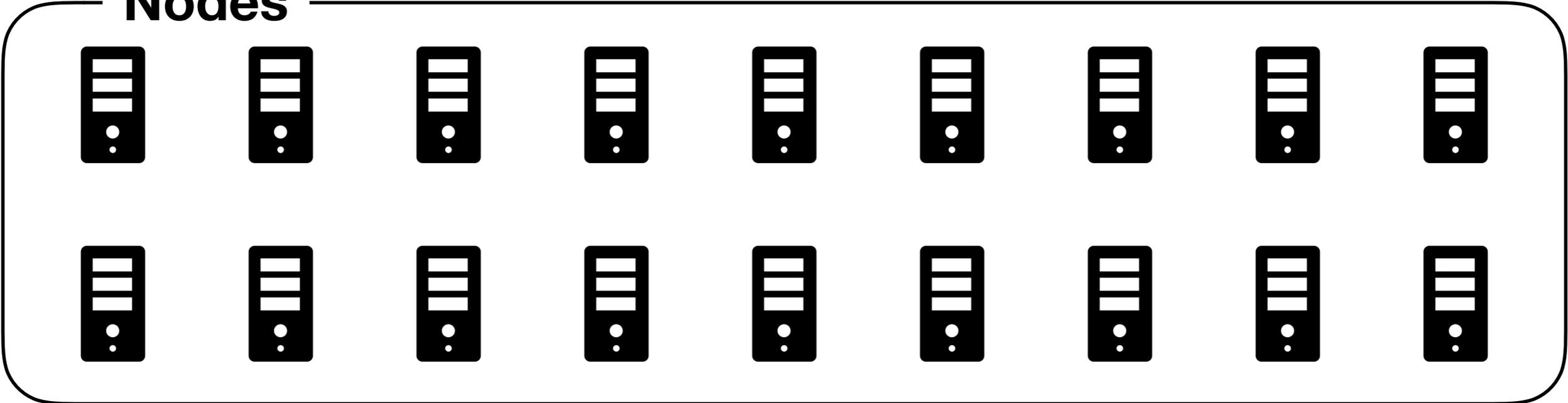


Find nodes with
4GB of free RAM



Query Server

Nodes

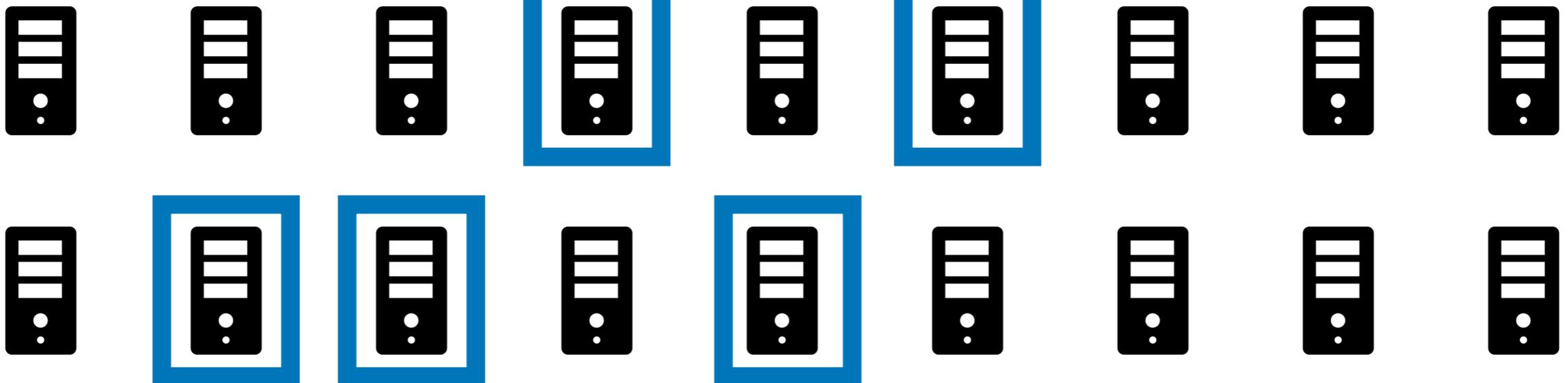




Find nodes with
4GB of free RAM

Query Server

Nodes



Use Cases

Use Cases

Cloud Management

Use Cases

Cloud Management

- ➔ VM placement and migration

Use Cases

Cloud Management

- ➔ VM placement and migration
- ➔ Host monitoring

Use Cases

Cloud Management

- ➔ VM placement and migration
- ➔ Host monitoring
- ➔ Resource utilization

Use Cases

Cloud Management

- ➔ VM placement and migration
- ➔ Host monitoring
- ➔ Resource utilization

NFV Automation

Use Cases

Cloud Management

- ➔ VM placement and migration
- ➔ Host monitoring
- ➔ Resource utilization

NFV Automation

- ➔ VNF homing on edge clouds

Use Cases

Cloud Management

- ➔ VM placement and migration
- ➔ Host monitoring
- ➔ Resource utilization

NFV Automation

- ➔ VNF homing on edge clouds
 - Multi-site VNF deployment

Use Cases

Cloud Management

- ➔ VM placement and migration
- ➔ Host monitoring
- ➔ Resource utilization

NFV Automation

- ➔ VNF homing on edge clouds
 - Multi-site VNF deployment
 - Different data sources

Use Cases

Cloud Management

- ➔ VM placement and migration
- ➔ Host monitoring
- ➔ Resource utilization

NFV Automation

- ➔ VNF homing on edge clouds
 - Multi-site VNF deployment
 - Different data sources
 - Simultaneous query processing

Challenges

Challenges

- Highly dynamic state

Challenges

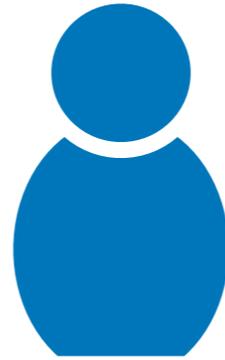
- Highly dynamic state
- Scale and geo-distribution

Challenges

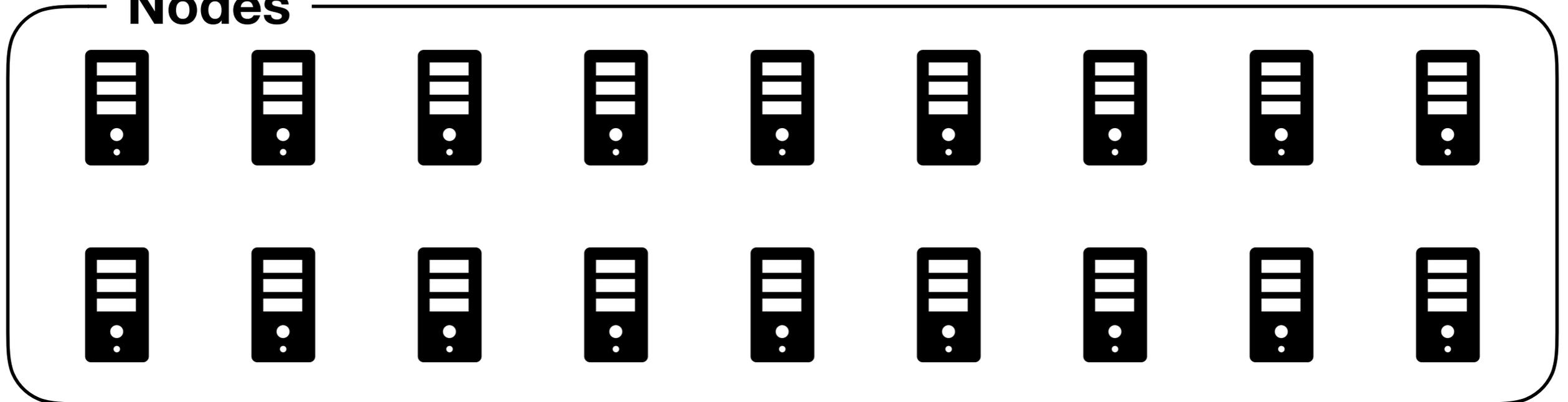
- Highly dynamic state
- Scale and geo-distribution
- Complex queries spanning different data sources

Node Finding Approaches

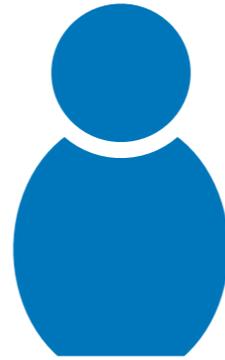
Frequent Push



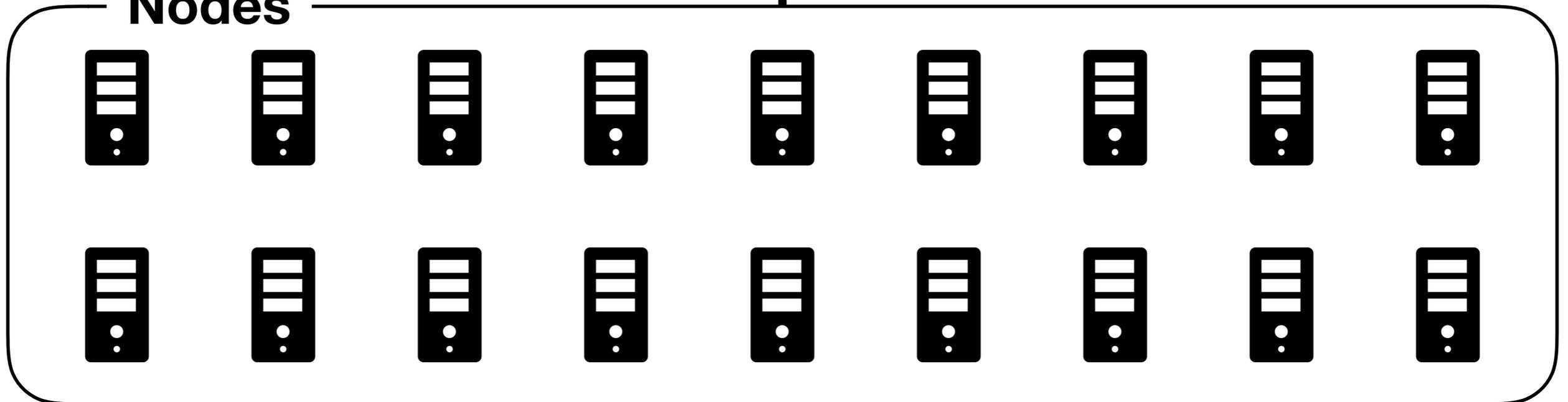
Nodes



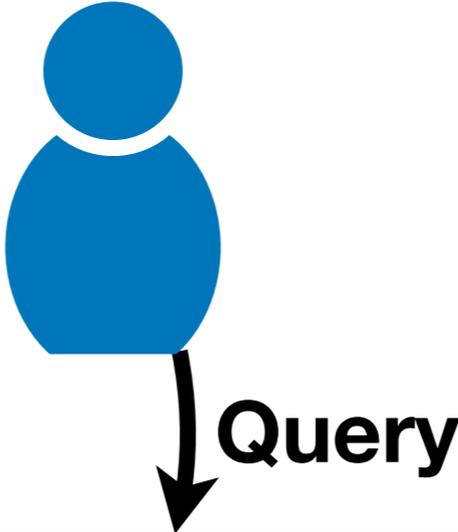
Frequent Push



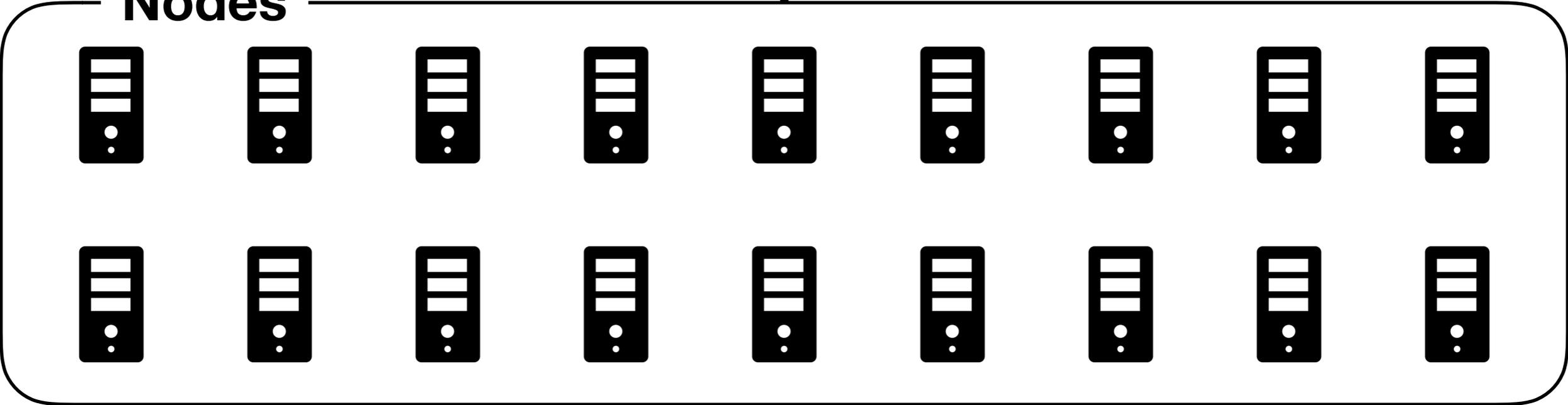
Nodes



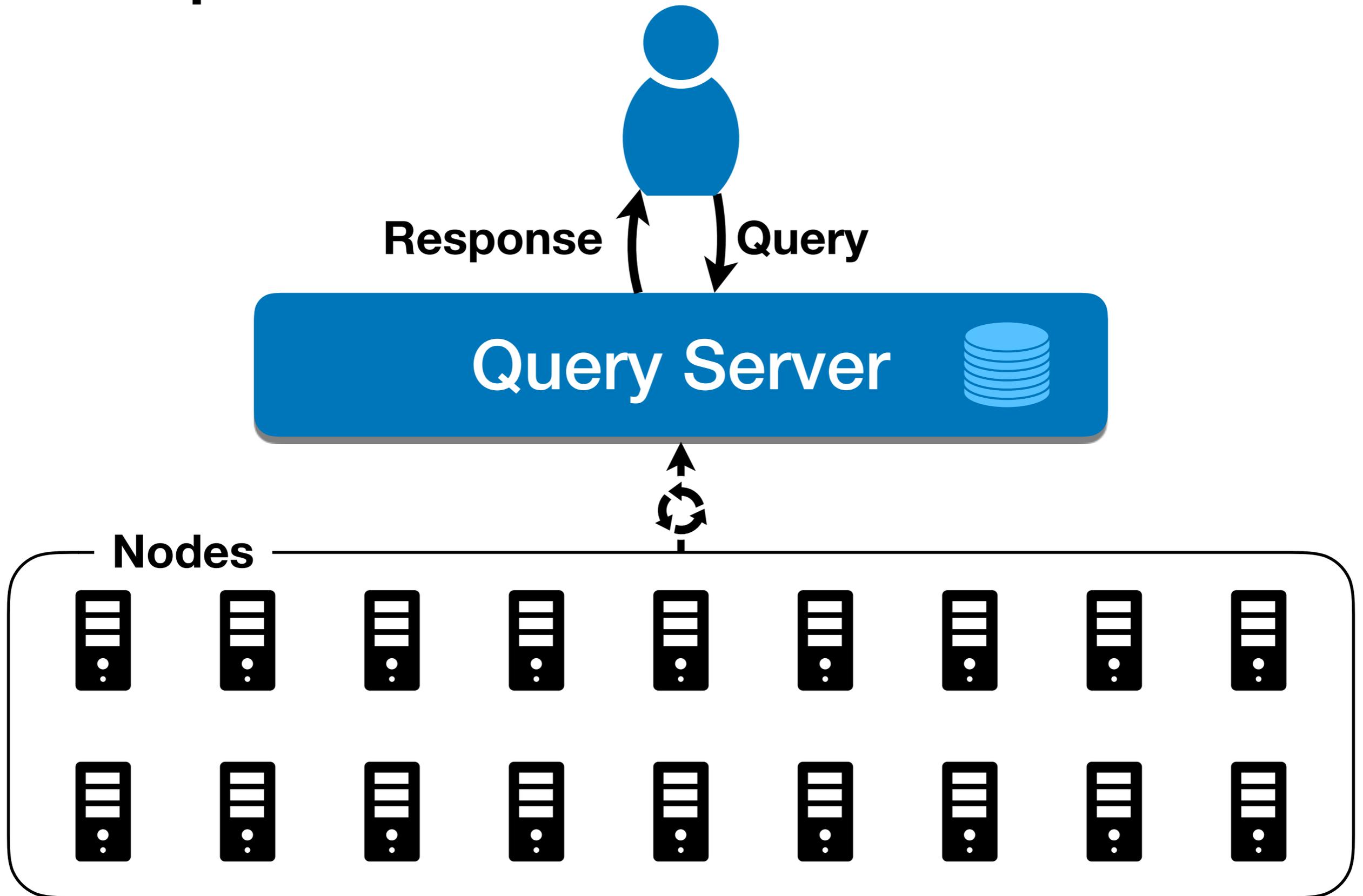
Frequent Push



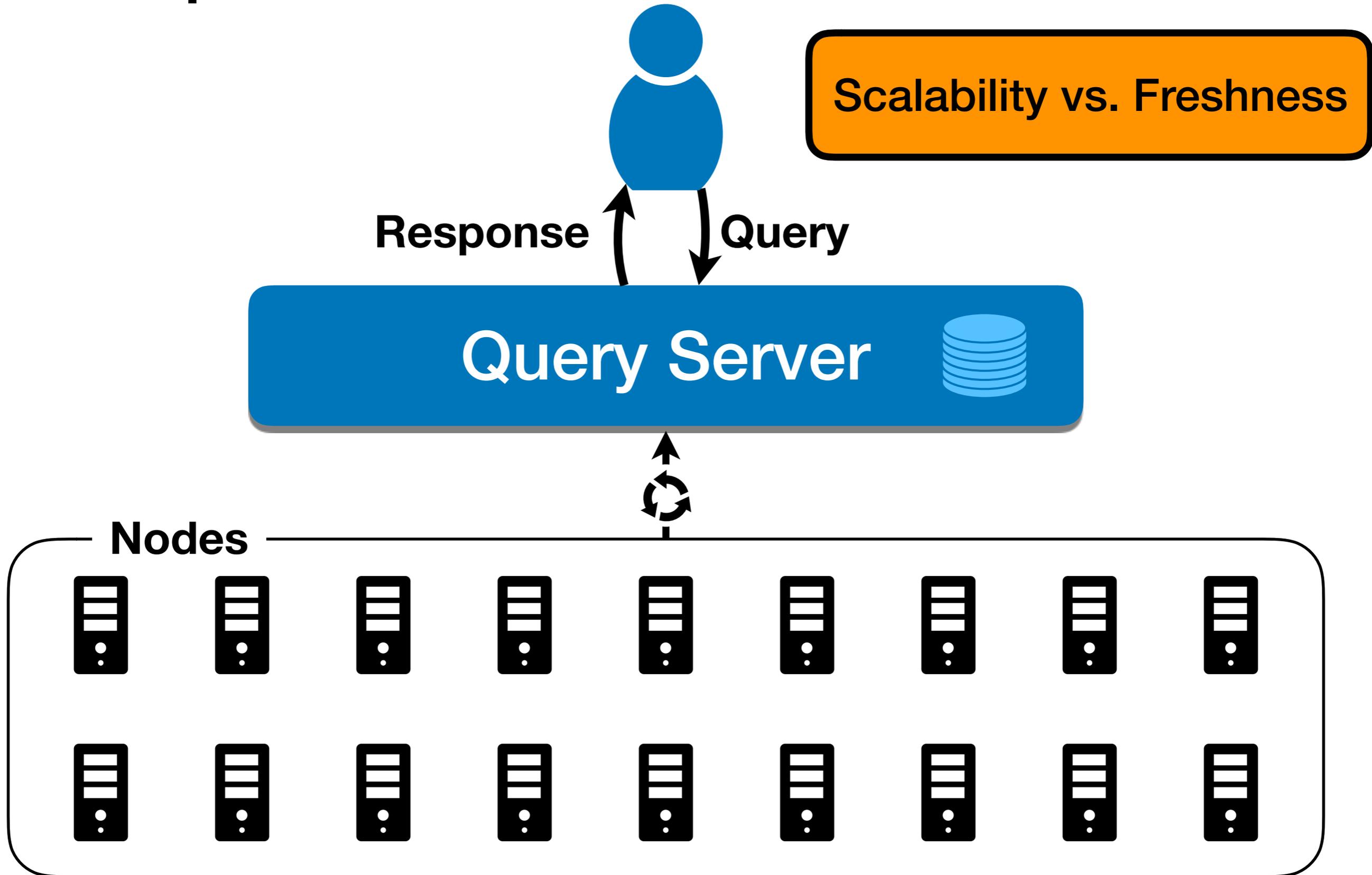
Nodes



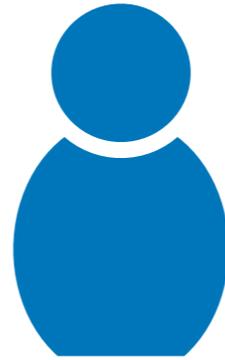
Frequent Push



Frequent Push

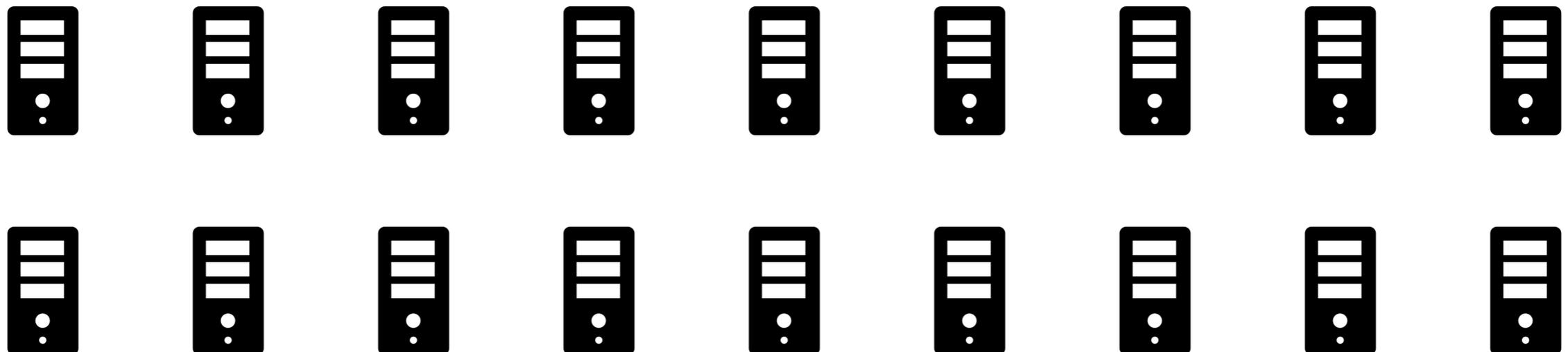


On-demand Pull

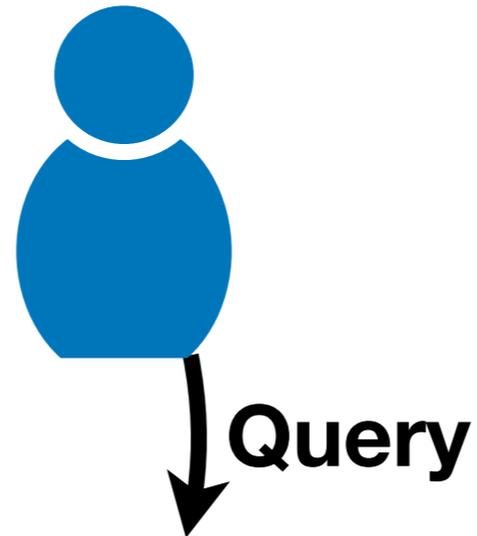


Query Server

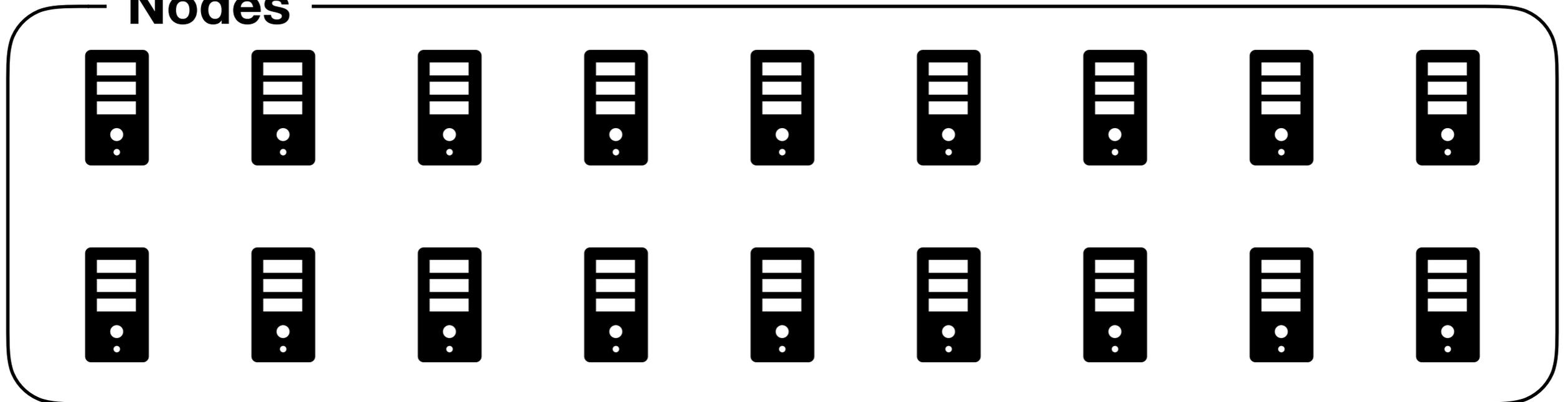
Nodes



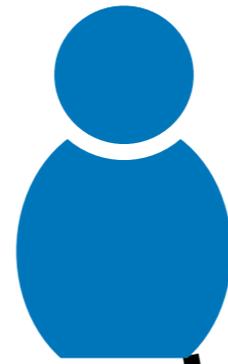
On-demand Pull



Nodes



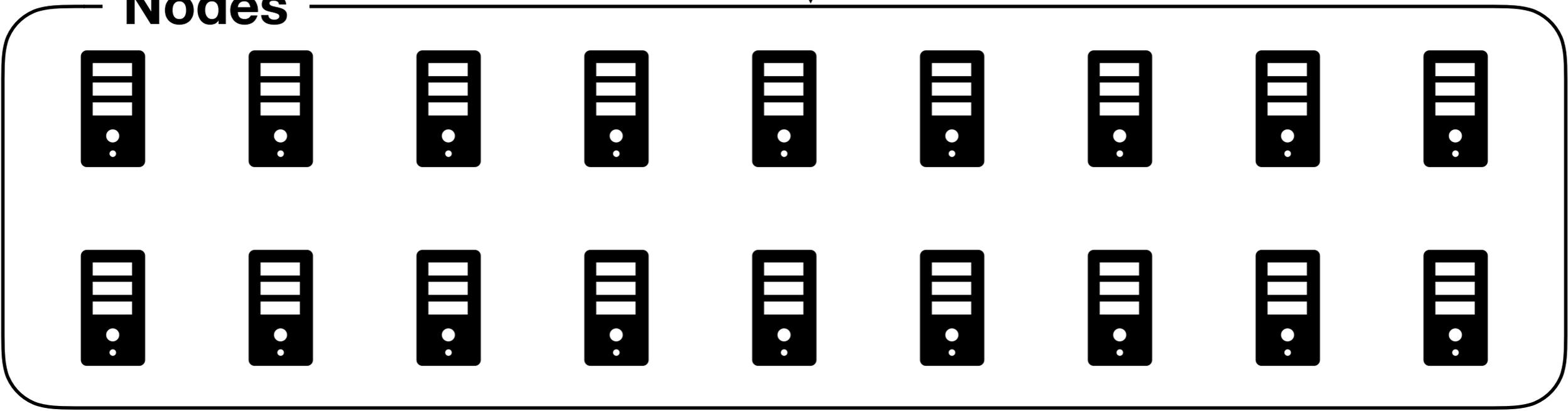
On-demand Pull



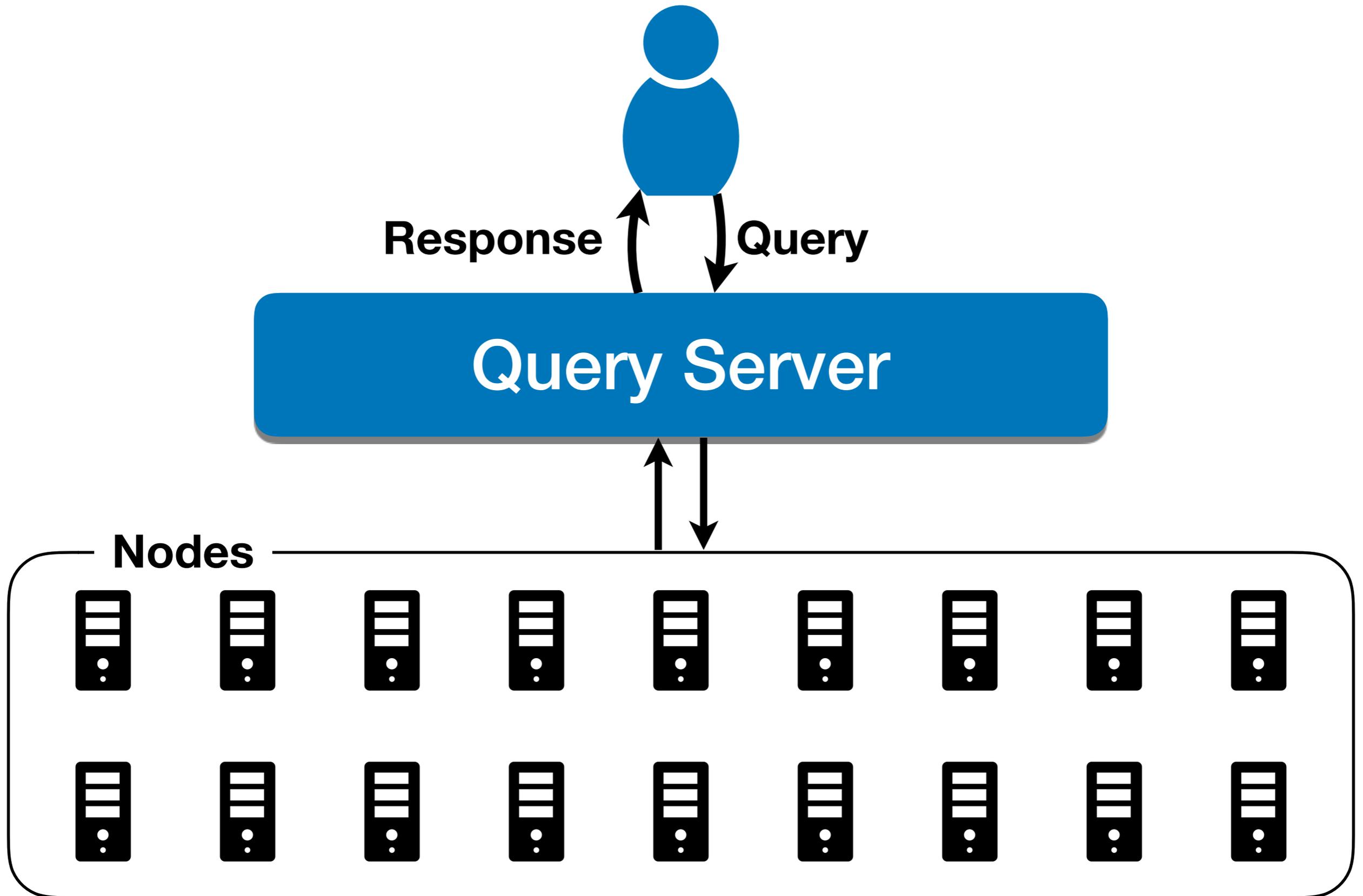
Query



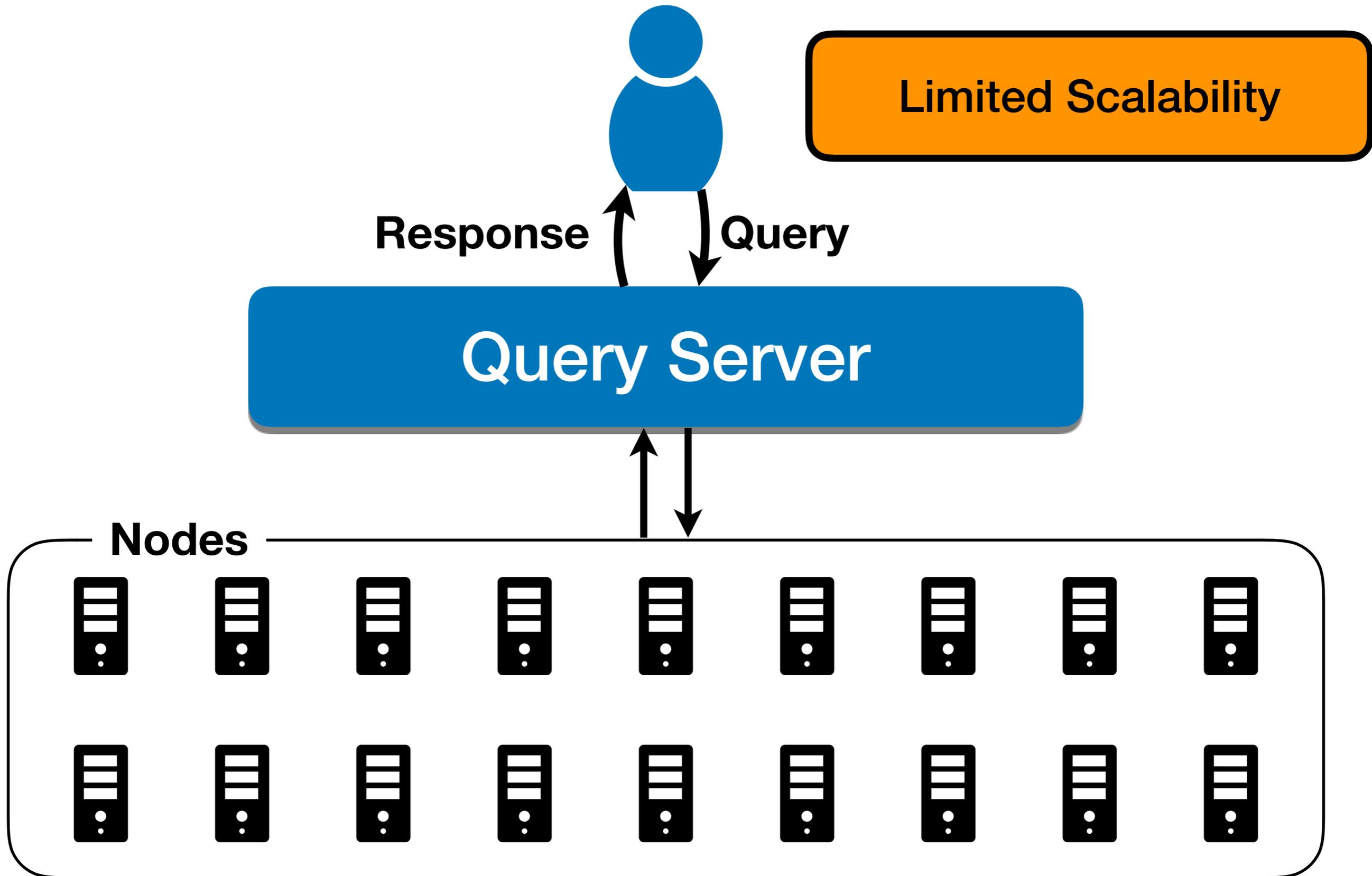
Nodes



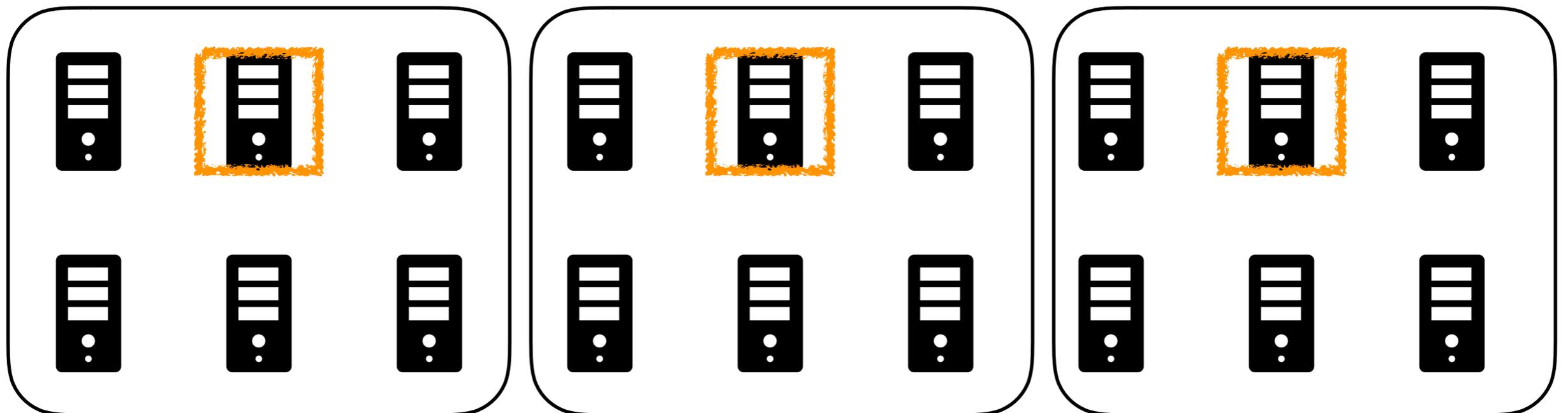
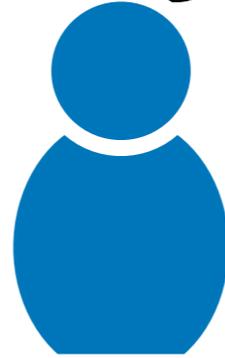
On-demand Pull



On-demand Pull



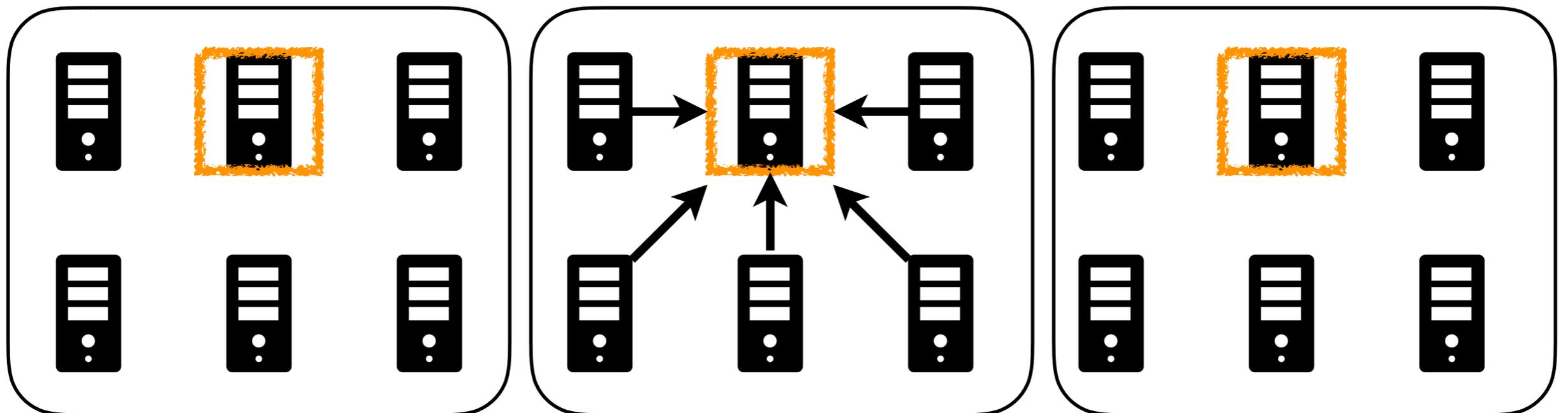
Hierarchical (Aggregation)



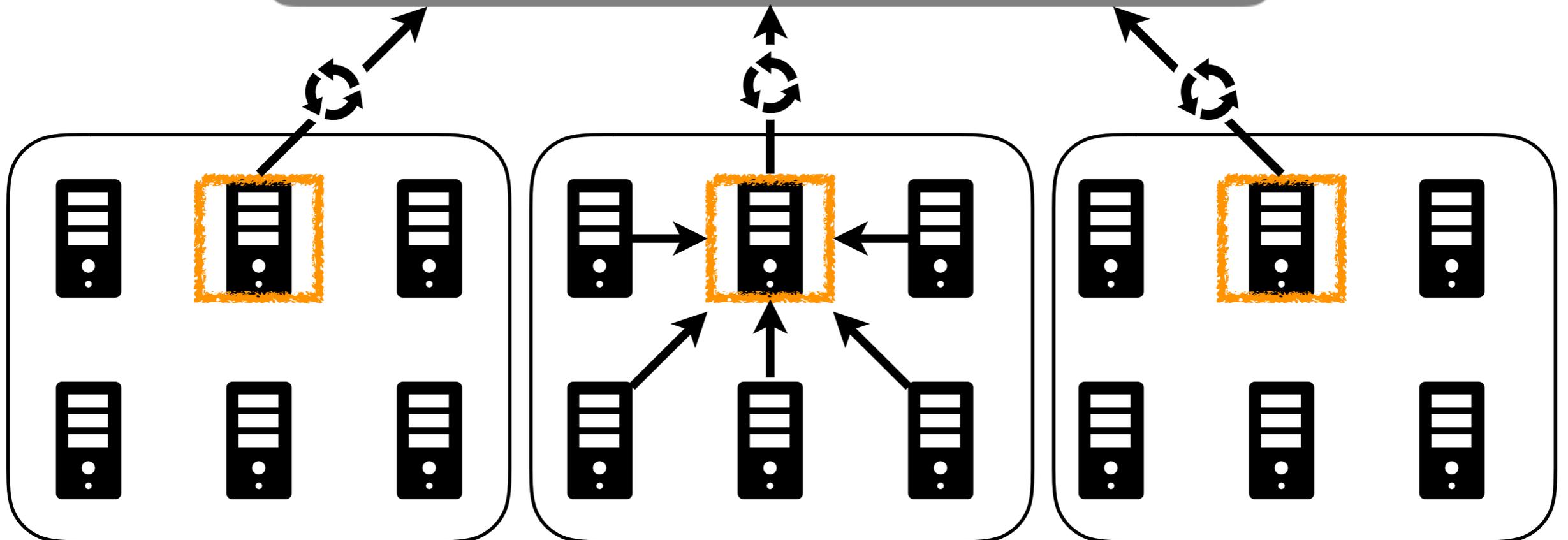
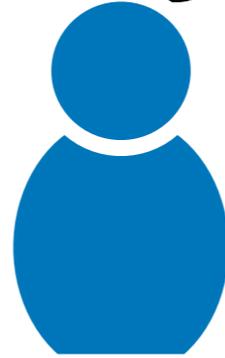
Hierarchical (Aggregation)



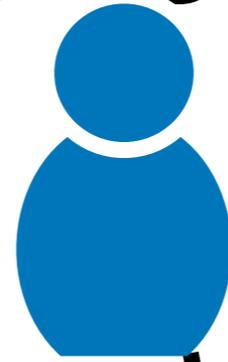
Query Server 

A blue rounded rectangular box containing the text "Query Server" in white, followed by a blue icon of a database stack.

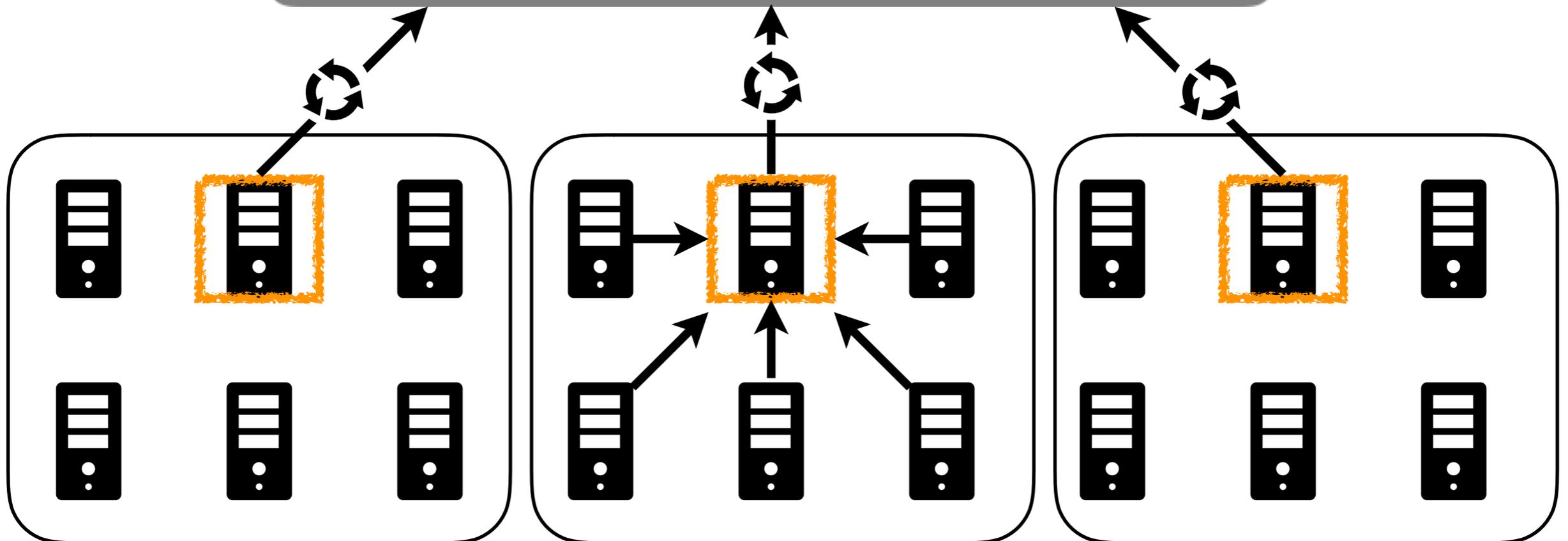
Hierarchical (Aggregation)



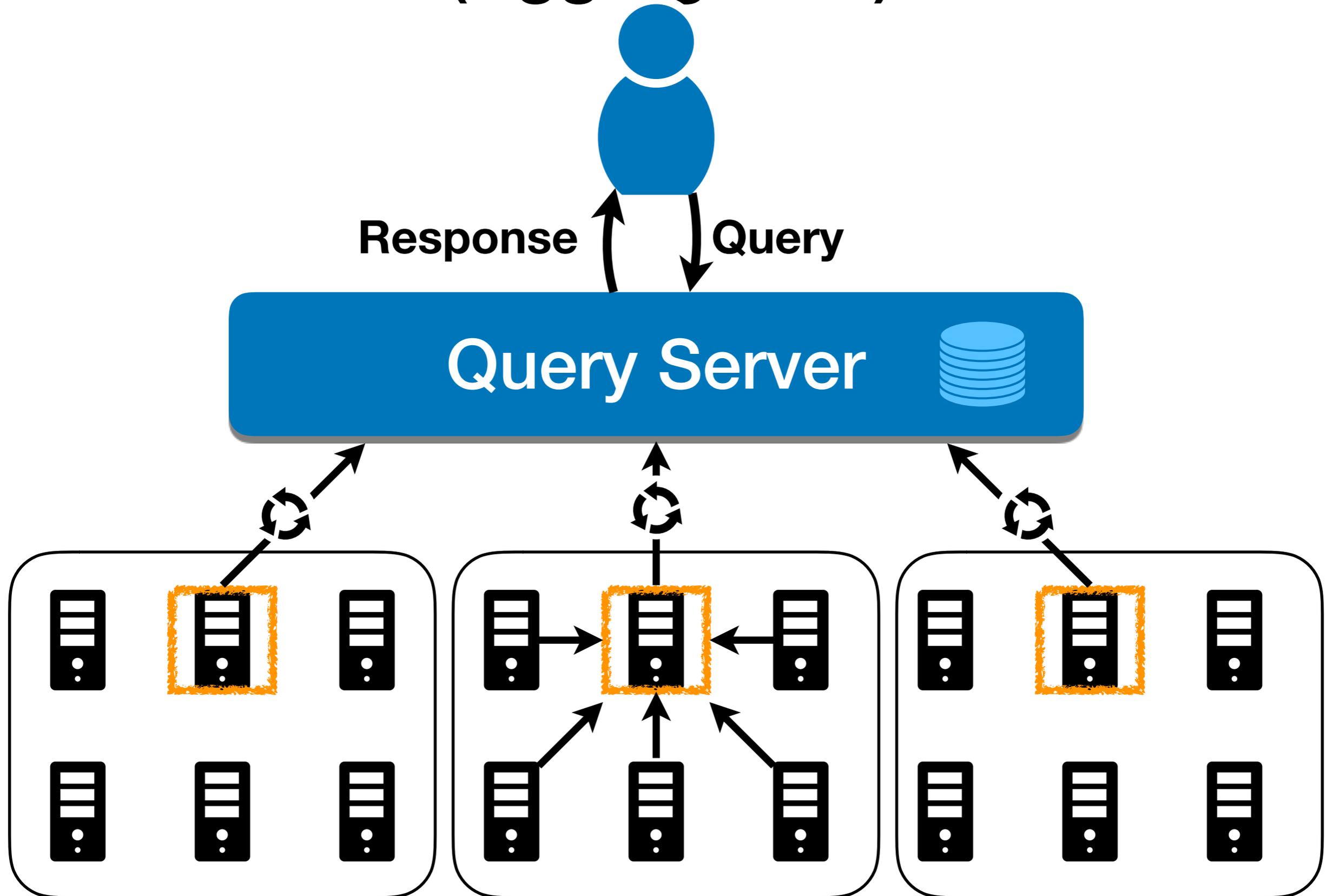
Hierarchical (Aggregation)



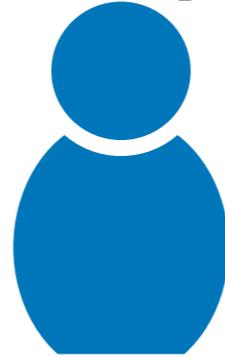
Query



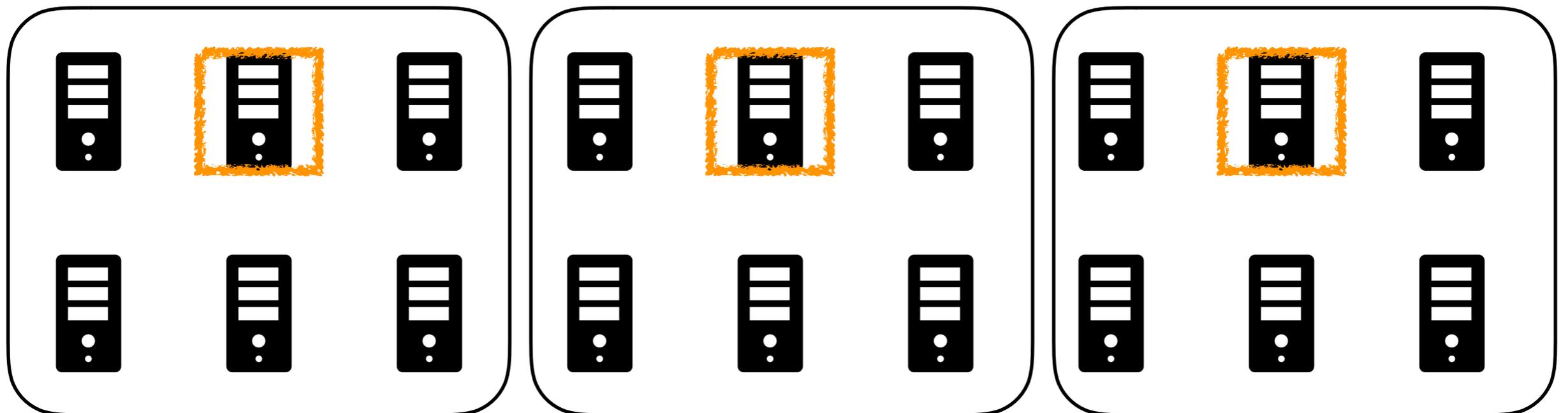
Hierarchical (Aggregation)



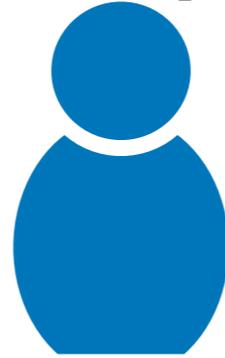
Hierarchical (Hybrid)



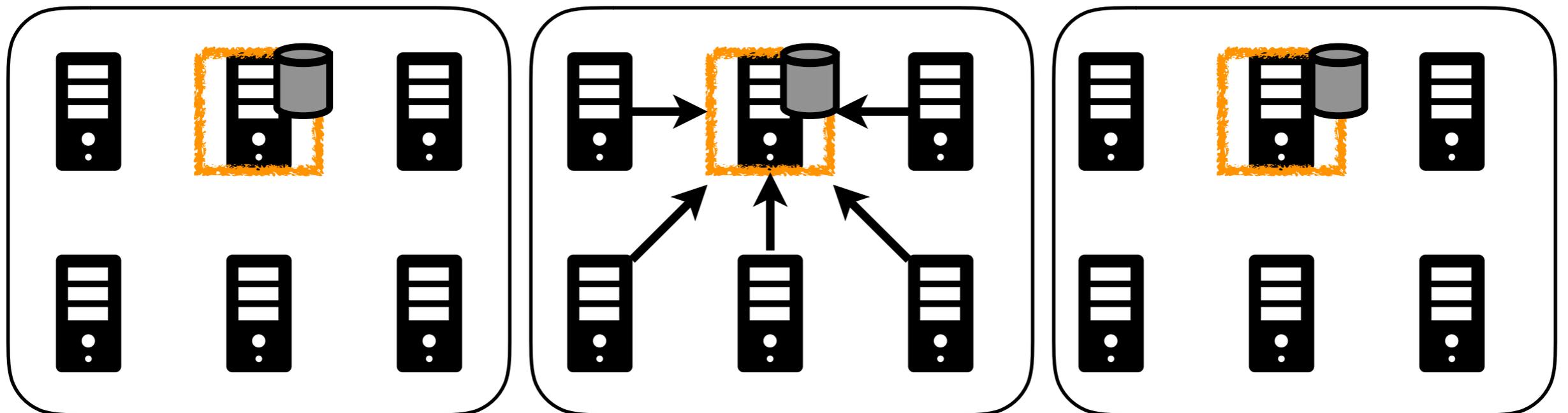
Query Server



Hierarchical (Hybrid)



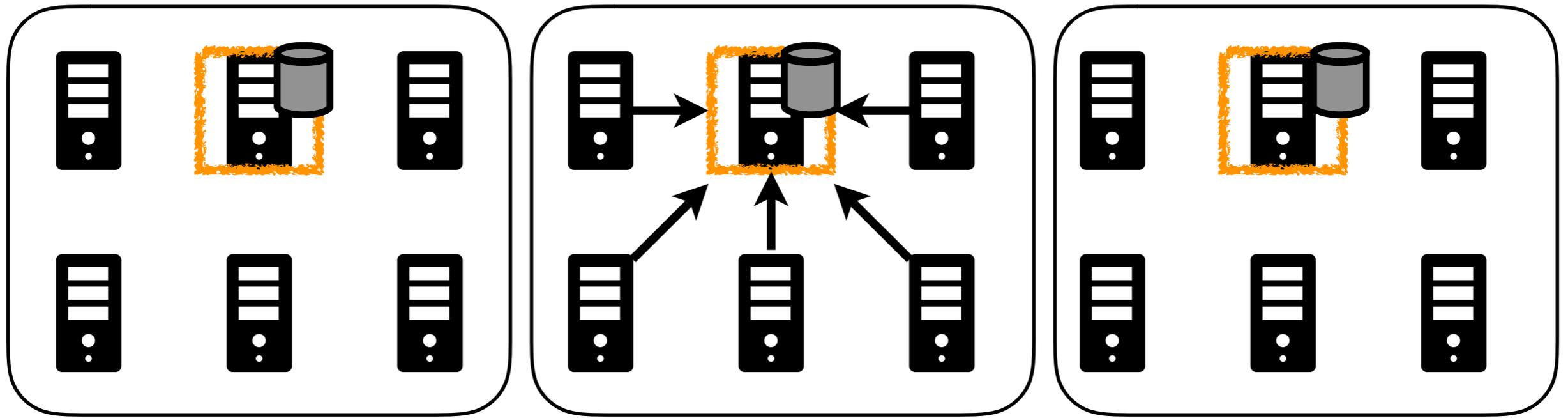
Query Server



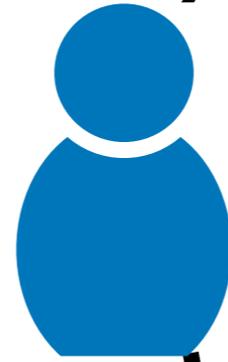
Hierarchical (Hybrid)



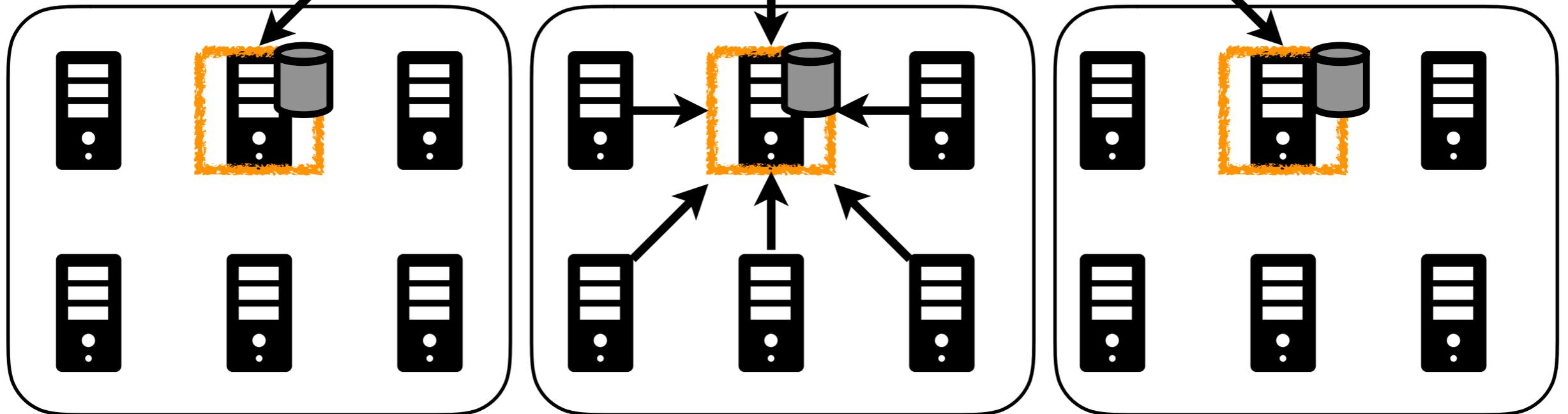
Query



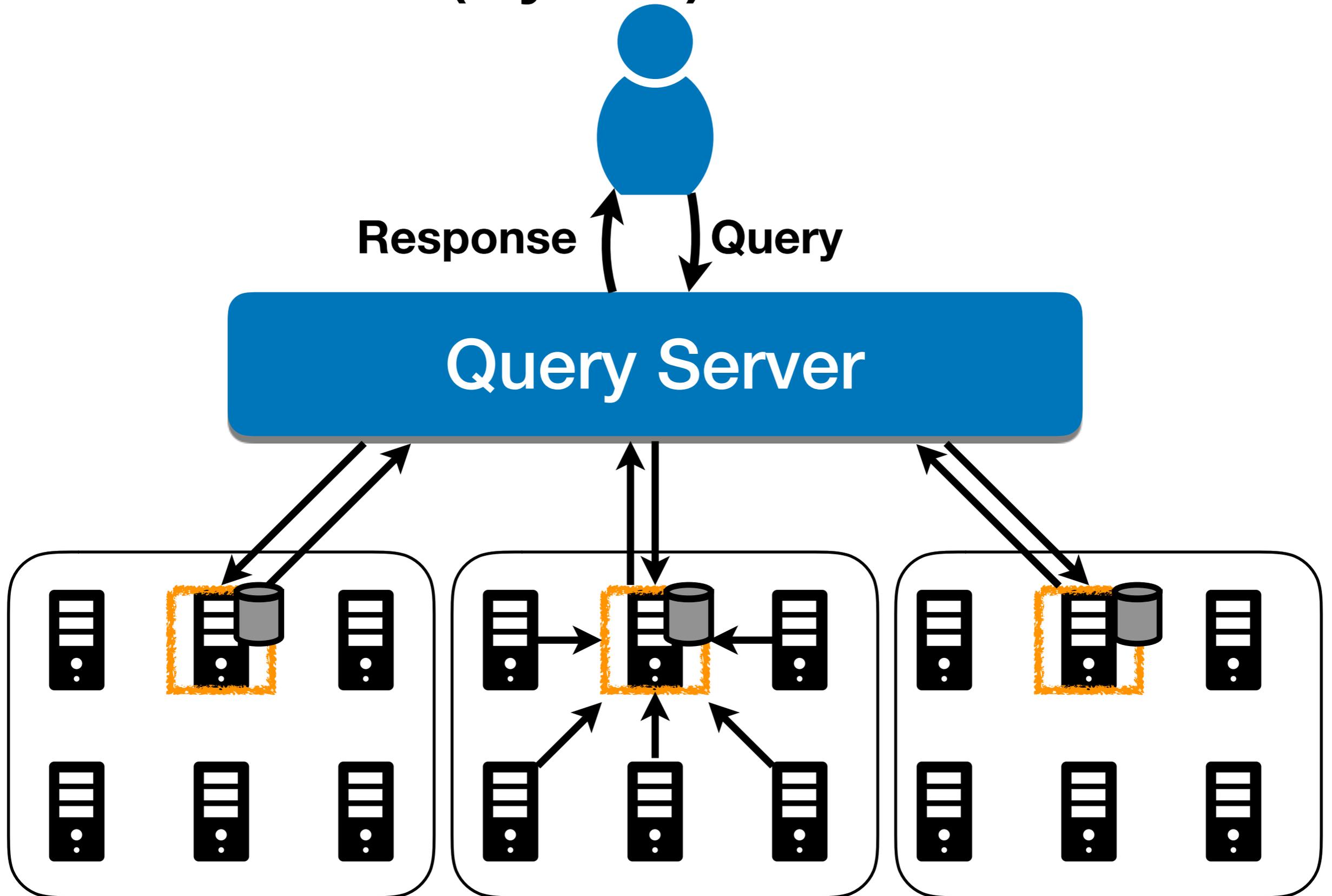
Hierarchical (Hybrid)



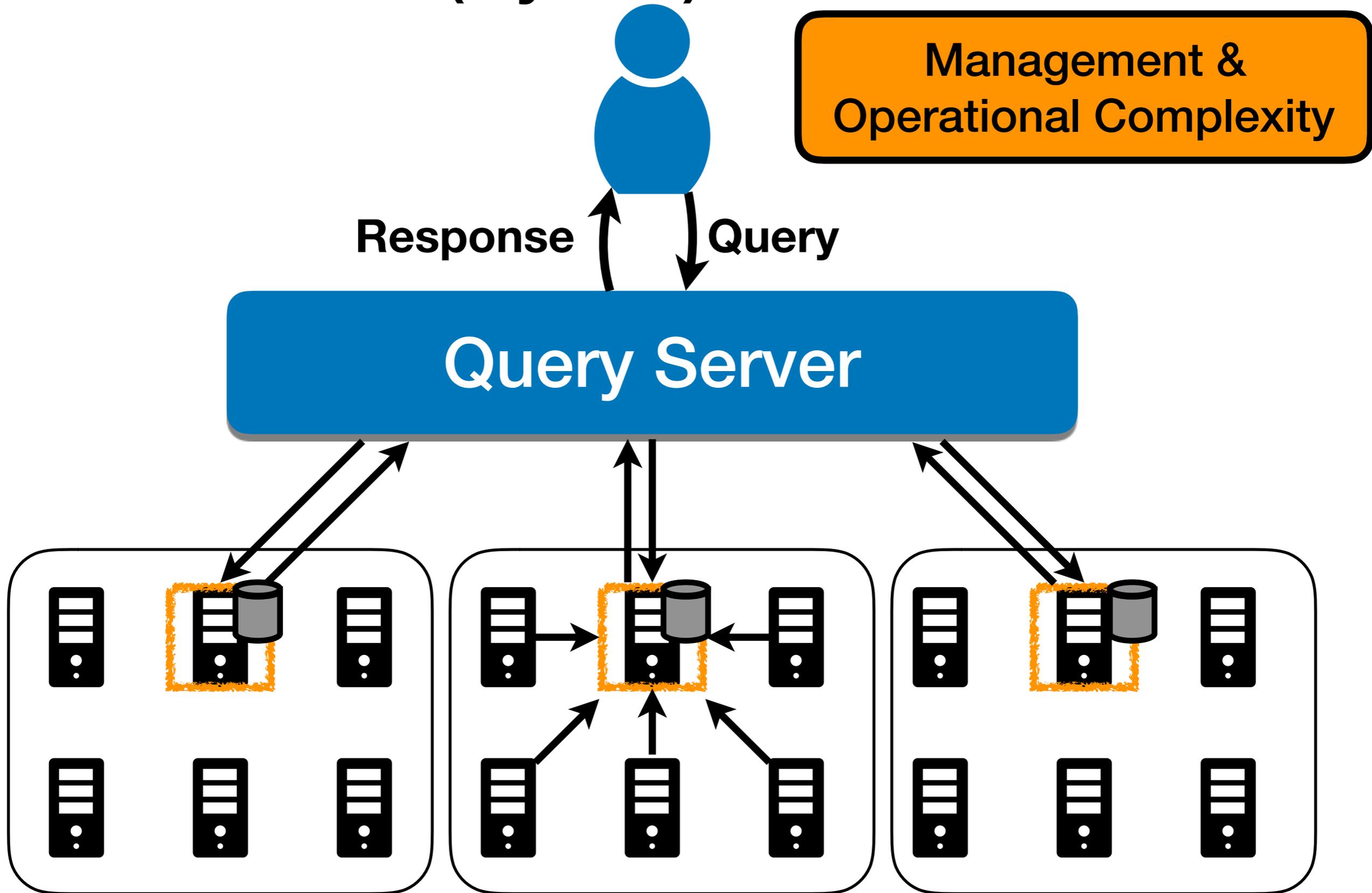
Query



Hierarchical (Hybrid)

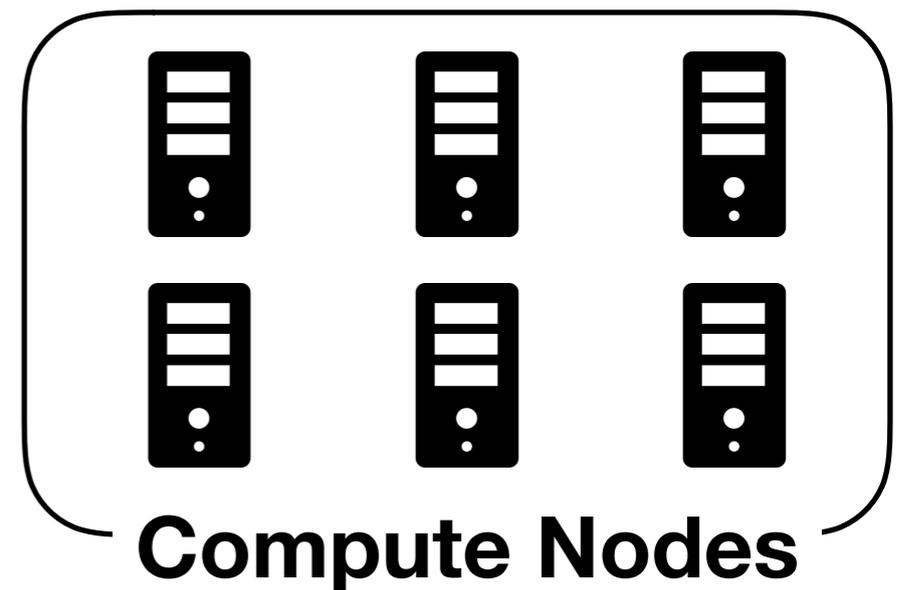


Hierarchical (Hybrid)



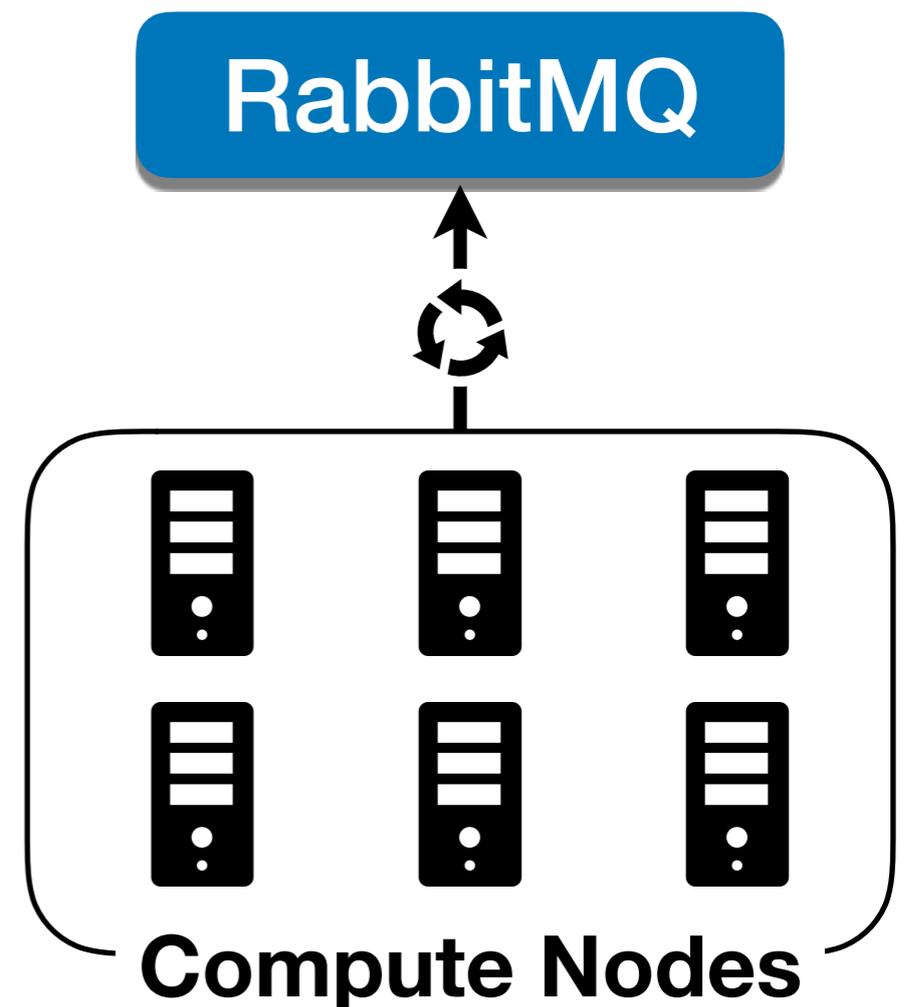
OpenStack Scalability Bottleneck

OpenStack Scalability Bottleneck



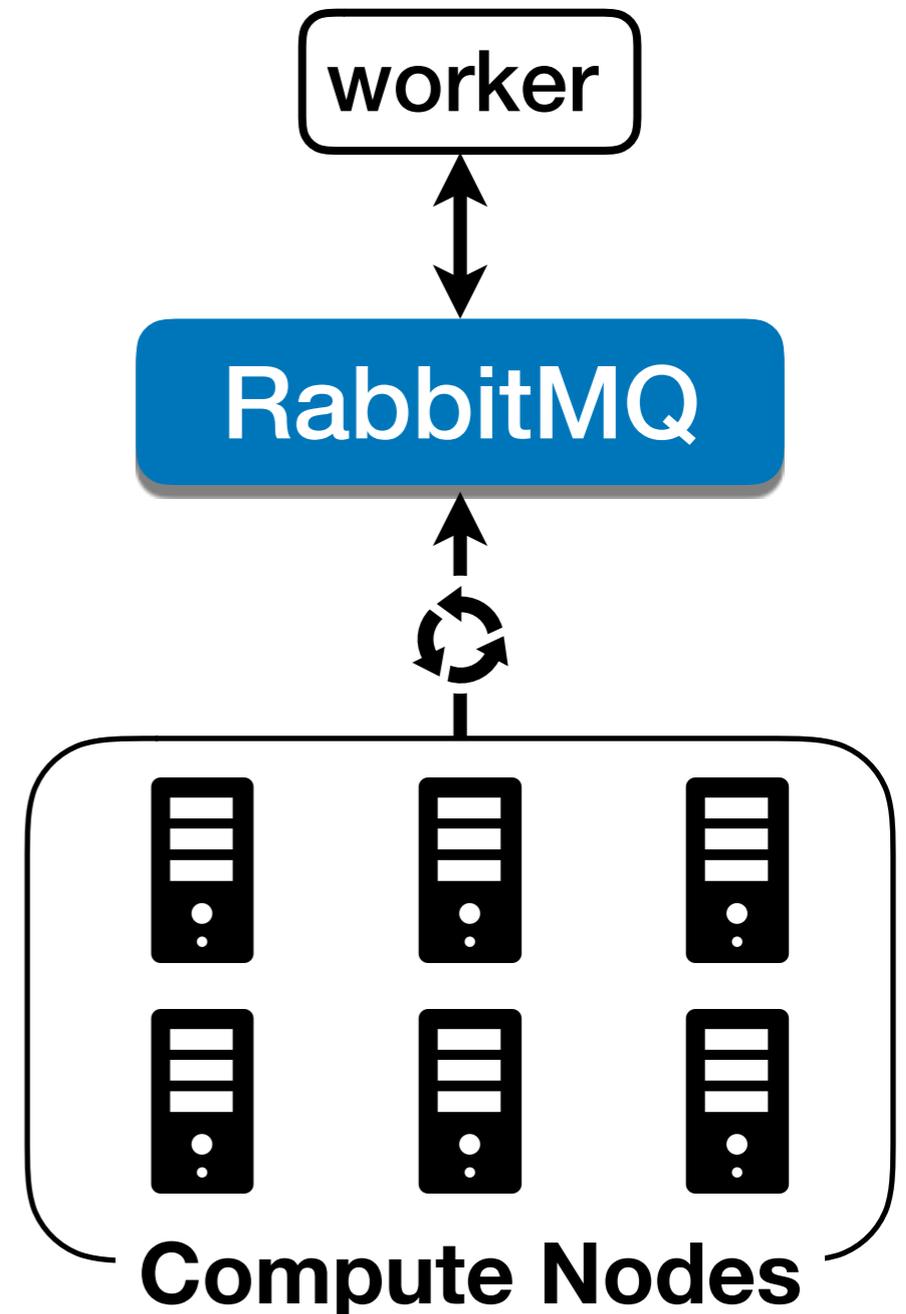
OpenStack Scalability Bottleneck

- Nova nodes push to RabbitMQ



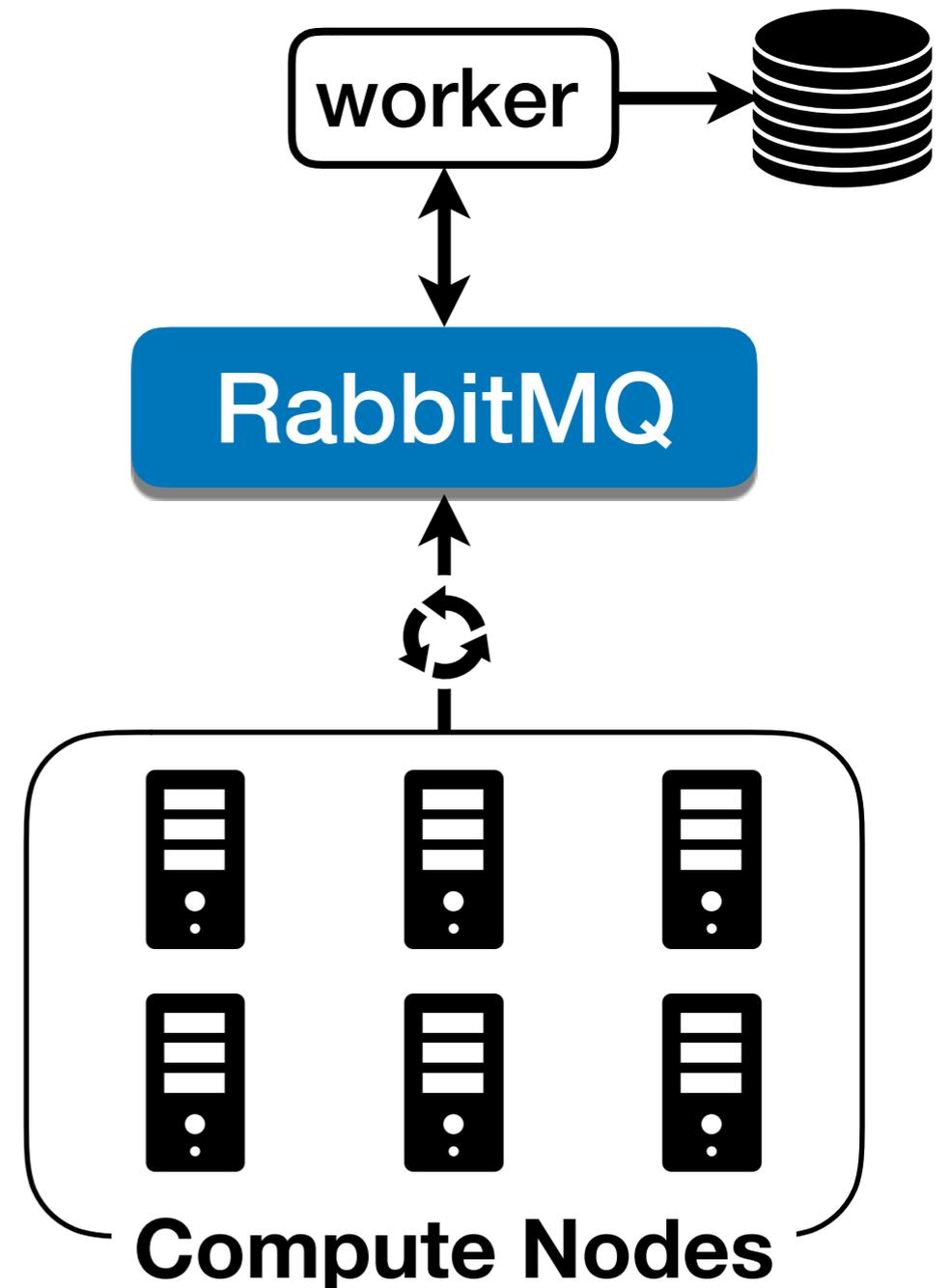
OpenStack Scalability Bottleneck

- Nova nodes push to RabbitMQ
- Worker (Conductor) dequeues



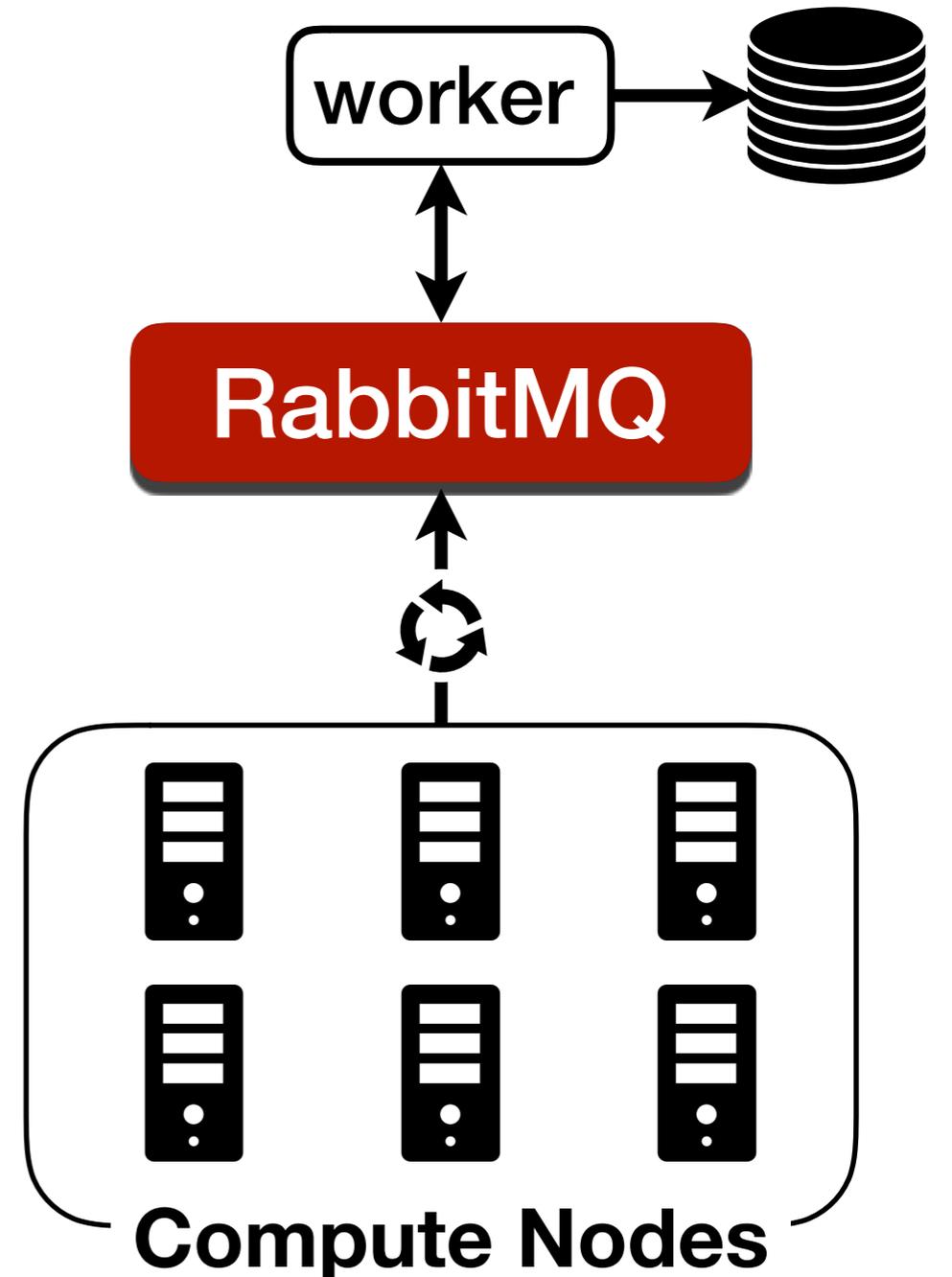
OpenStack Scalability Bottleneck

- Nova nodes push to RabbitMQ
- Worker (Conductor) dequeues
- Worker pushes to DB



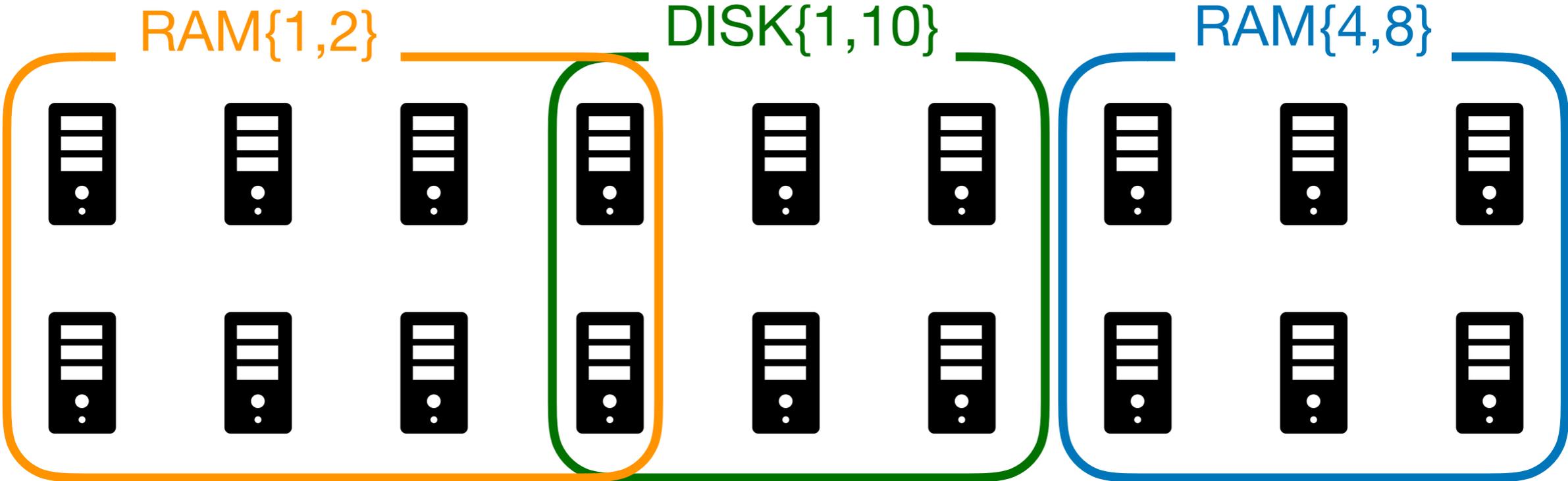
OpenStack Scalability Bottleneck

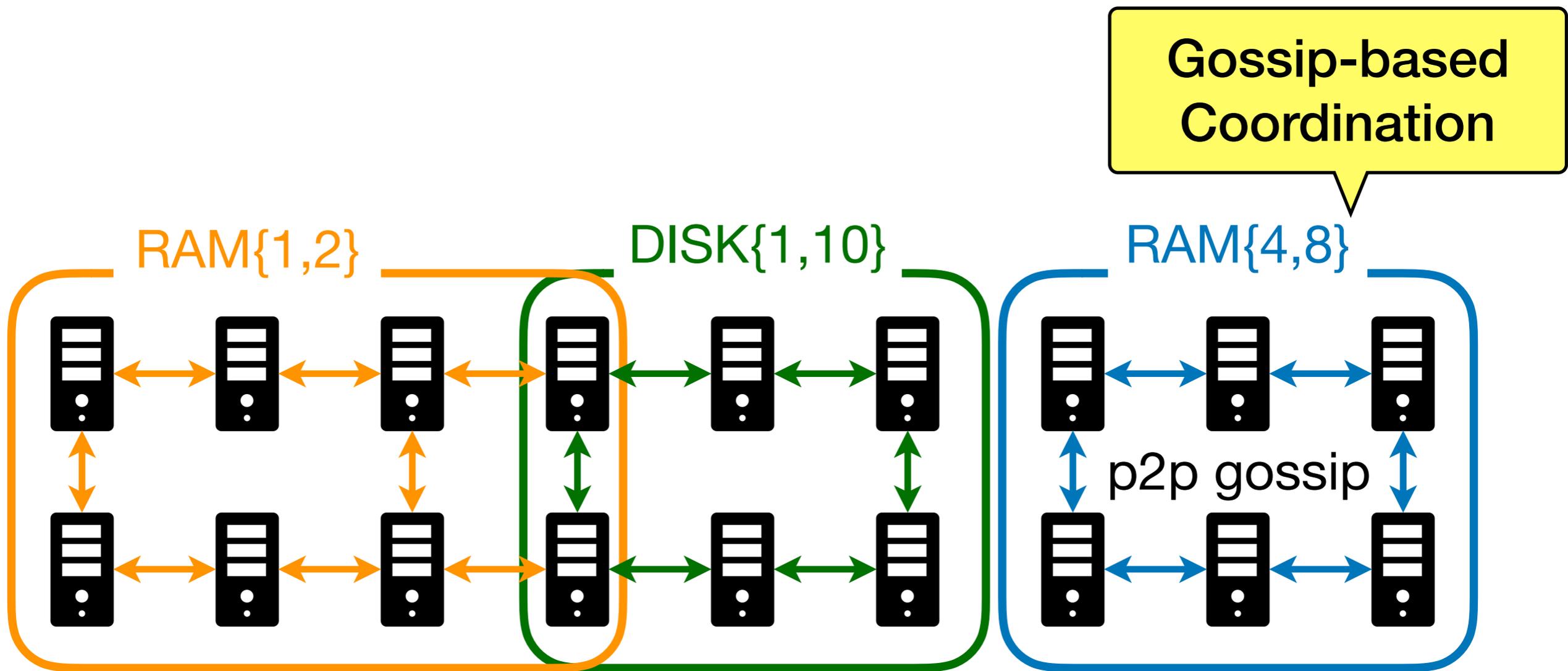
- Nova nodes push to RabbitMQ
- Worker (Conductor) dequeues
- Worker pushes to DB
- RabbitMQ node is a bottleneck



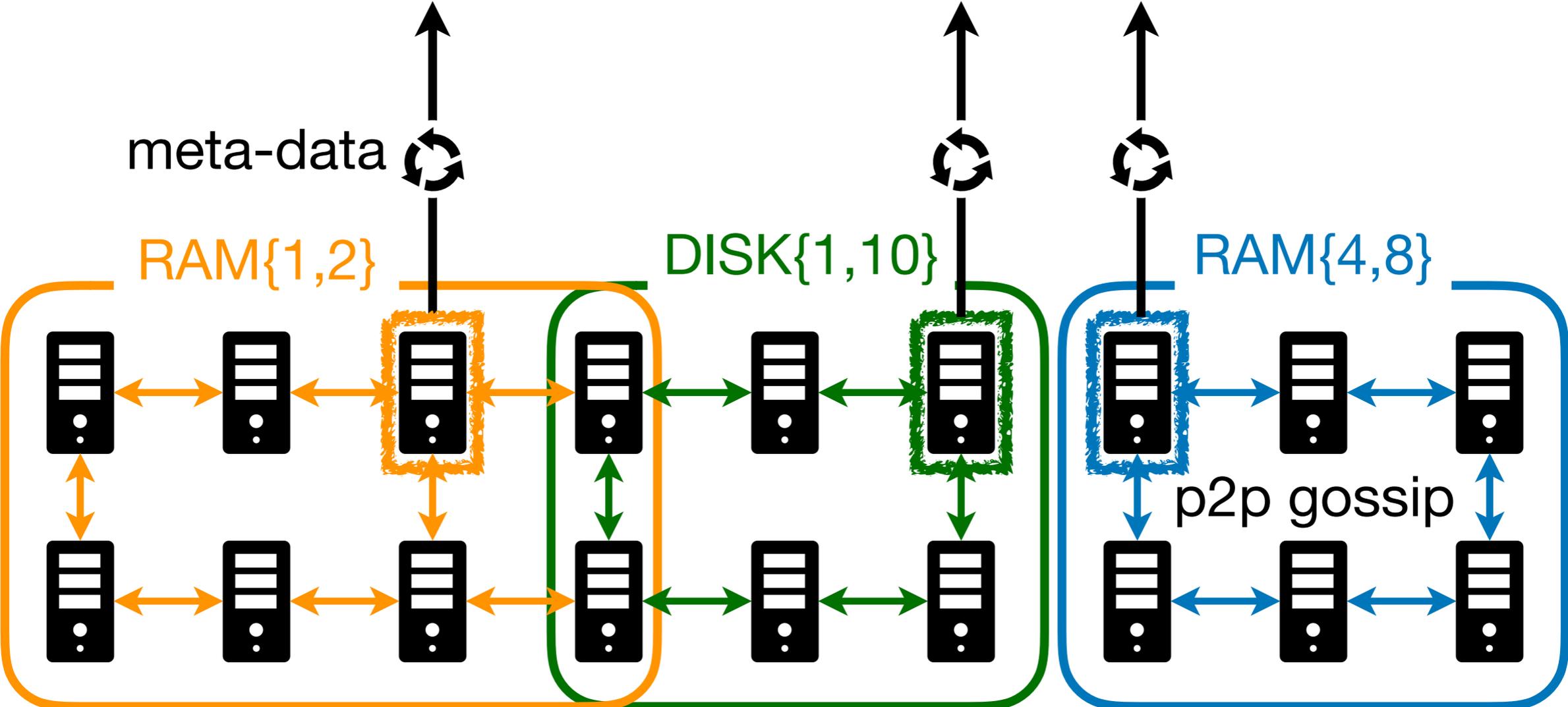
NodeFinder

Attribute-based Grouping



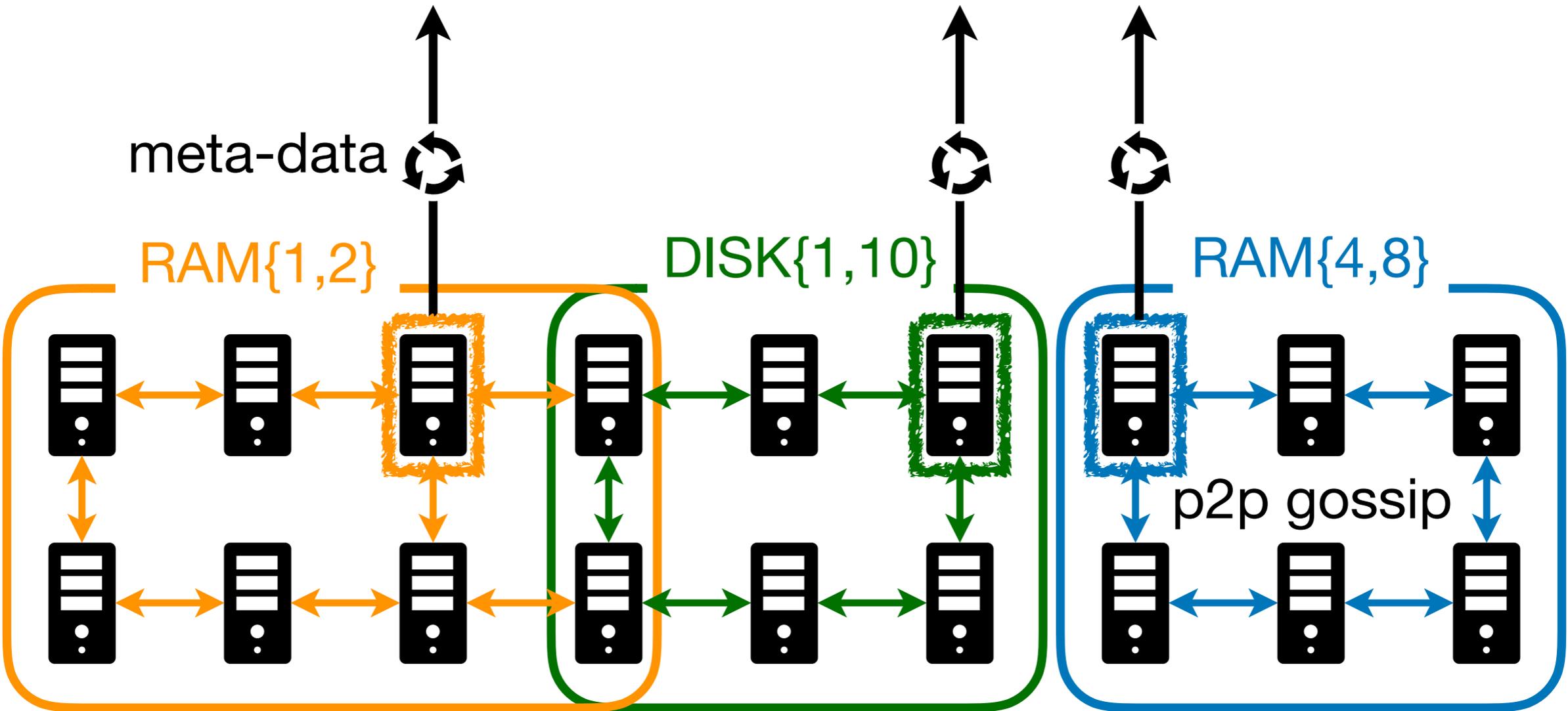


NodeFinder



Find nodes with 1GB of free RAM & 5GB free DISK

REST API
NodeFinder



Find nodes with 1GB of free RAM & 5GB free DISK

REST API
NodeFinder

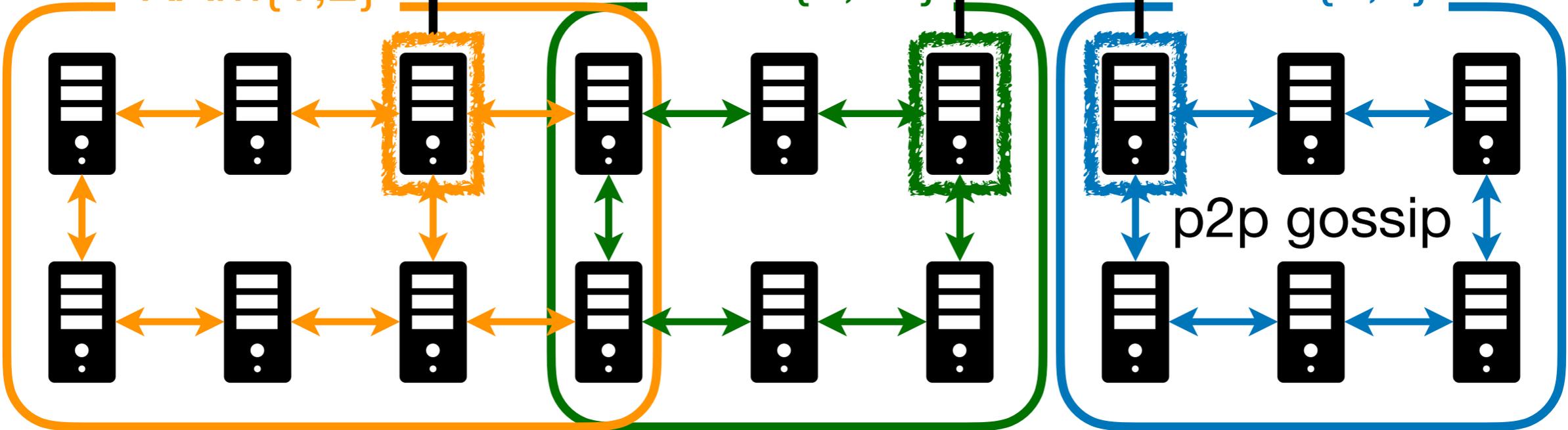
Directed Pulling

meta-data

RAM{1,2}

DISK{1,10}

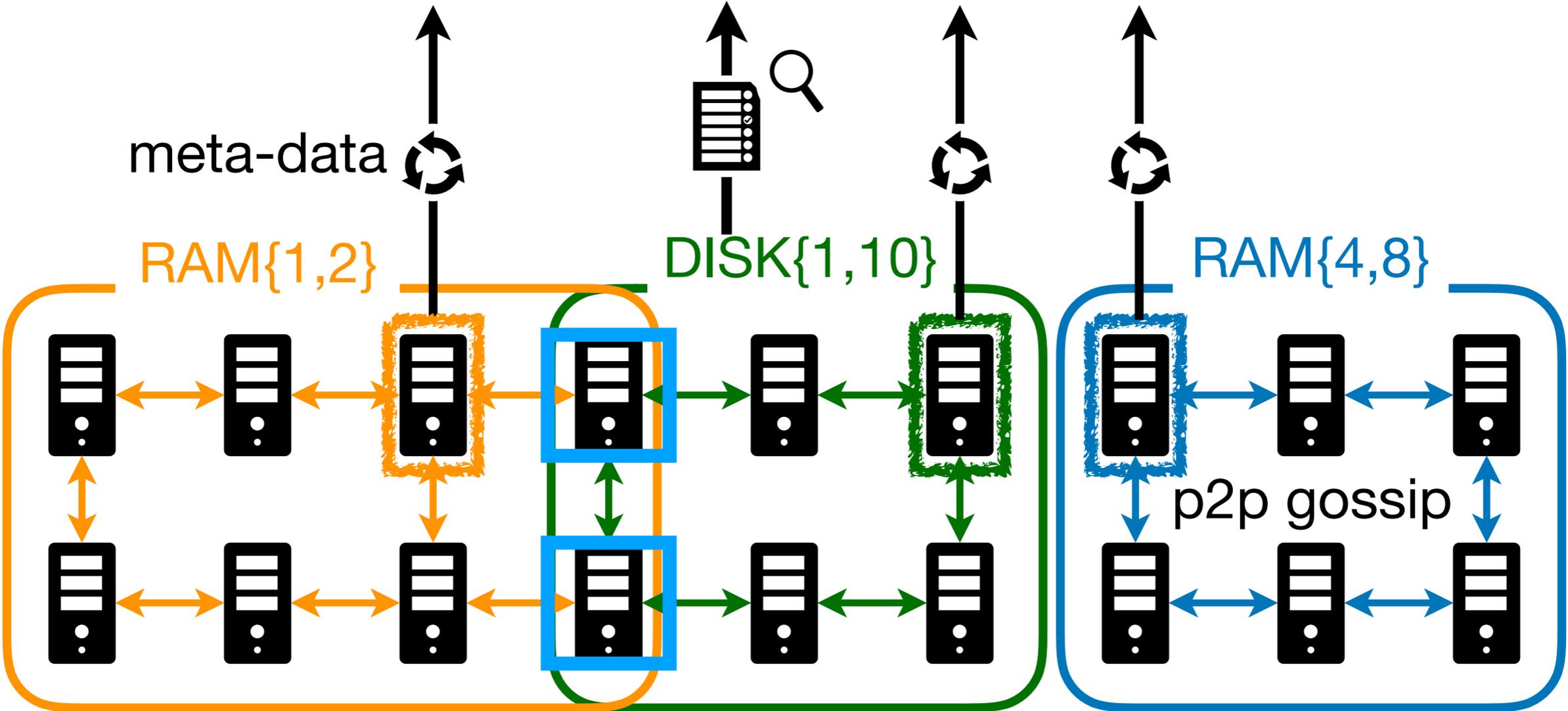
RAM{4,8}



Find nodes with 1GB of free RAM & 5GB free DISK

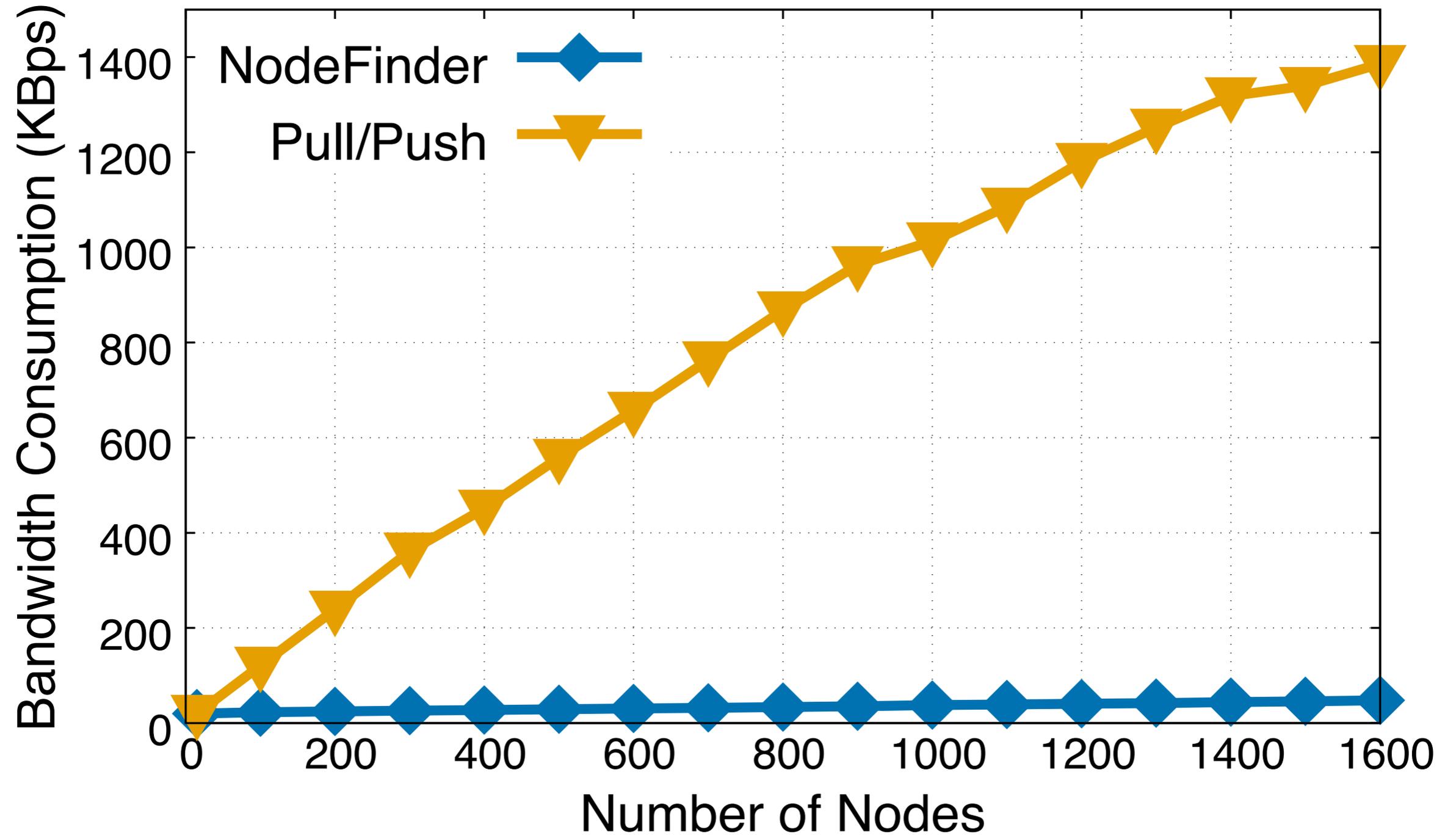
Response

REST API
NodeFinder

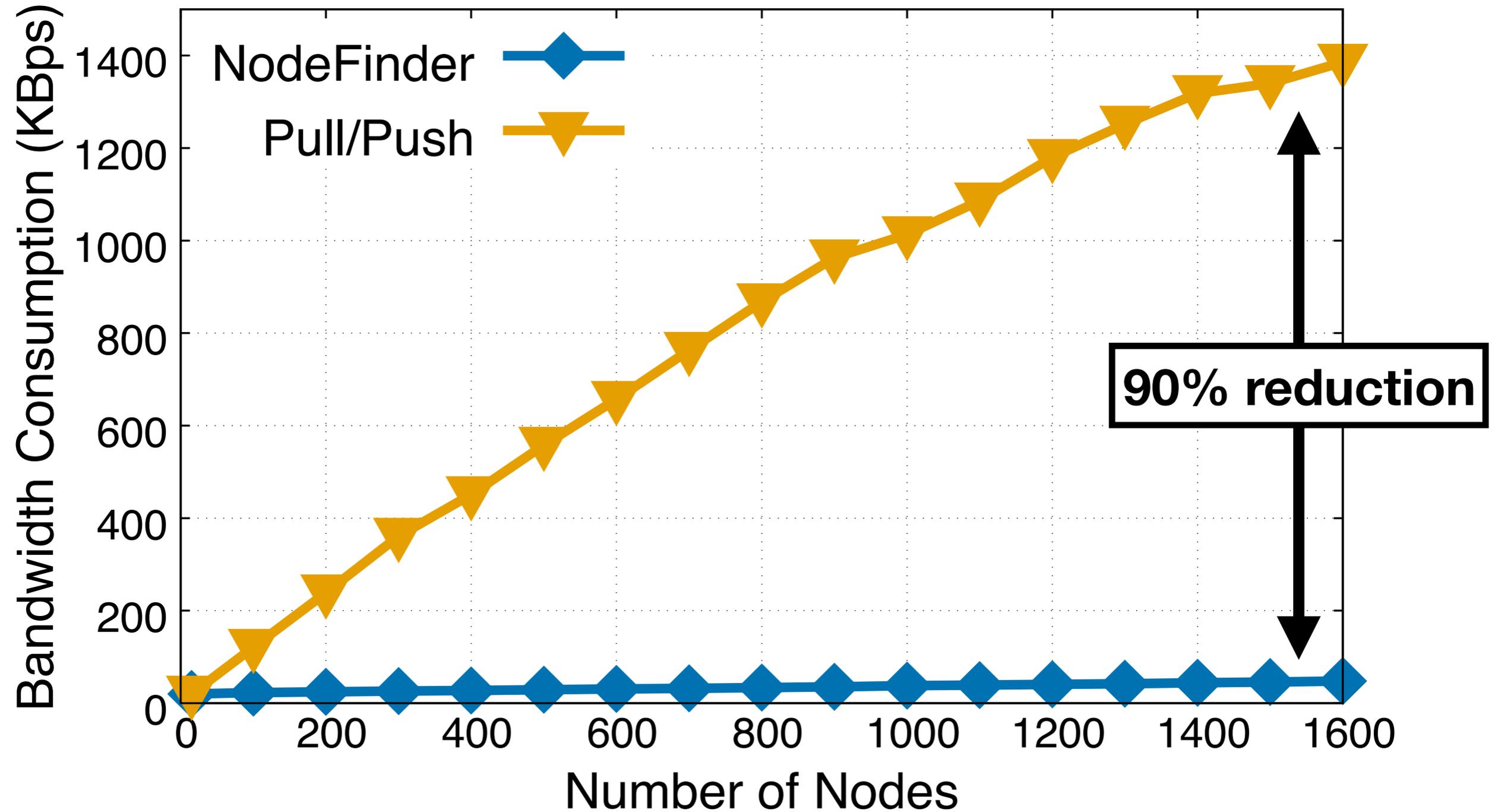


Evaluation

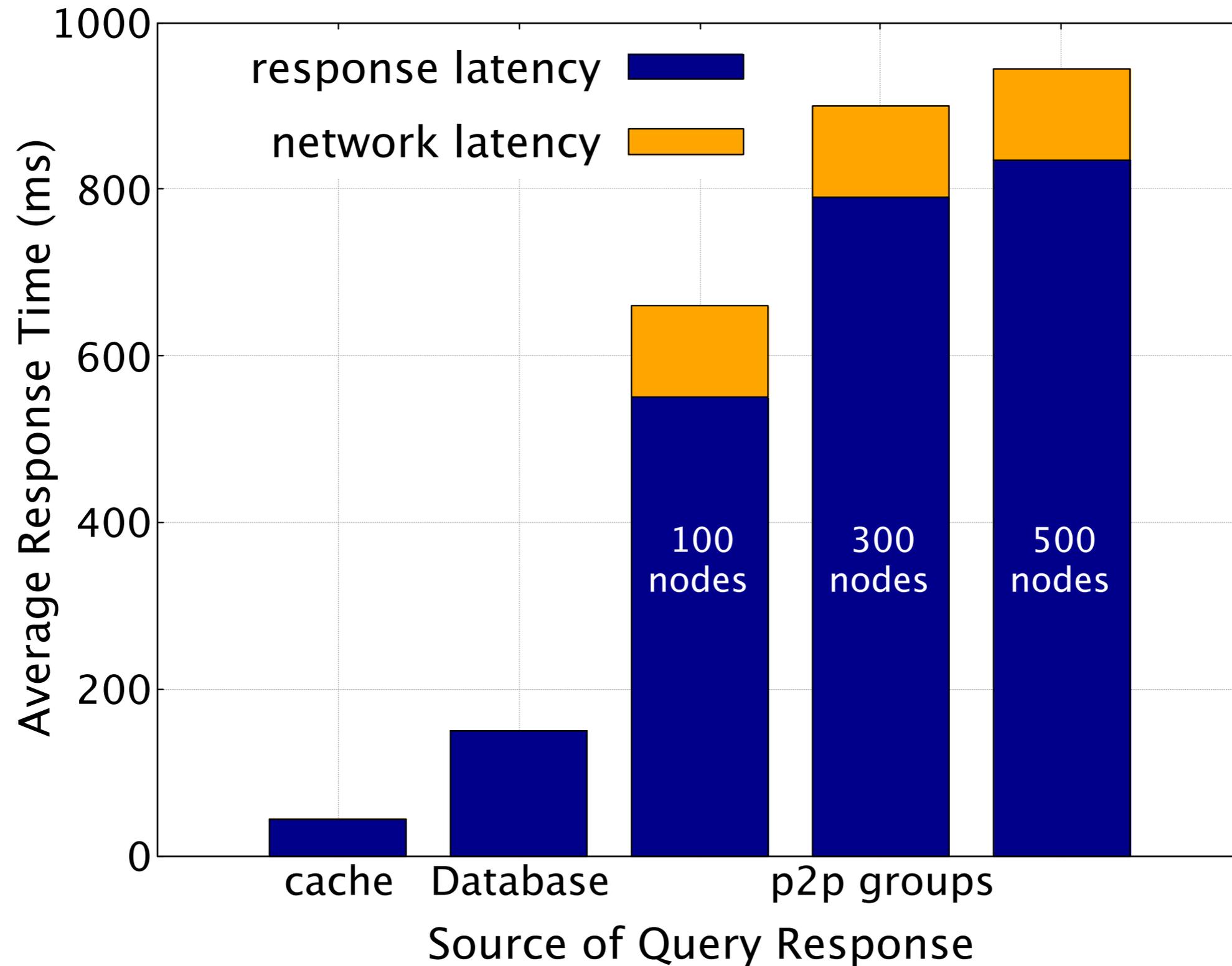
NodeFinder's Scalability (BW)



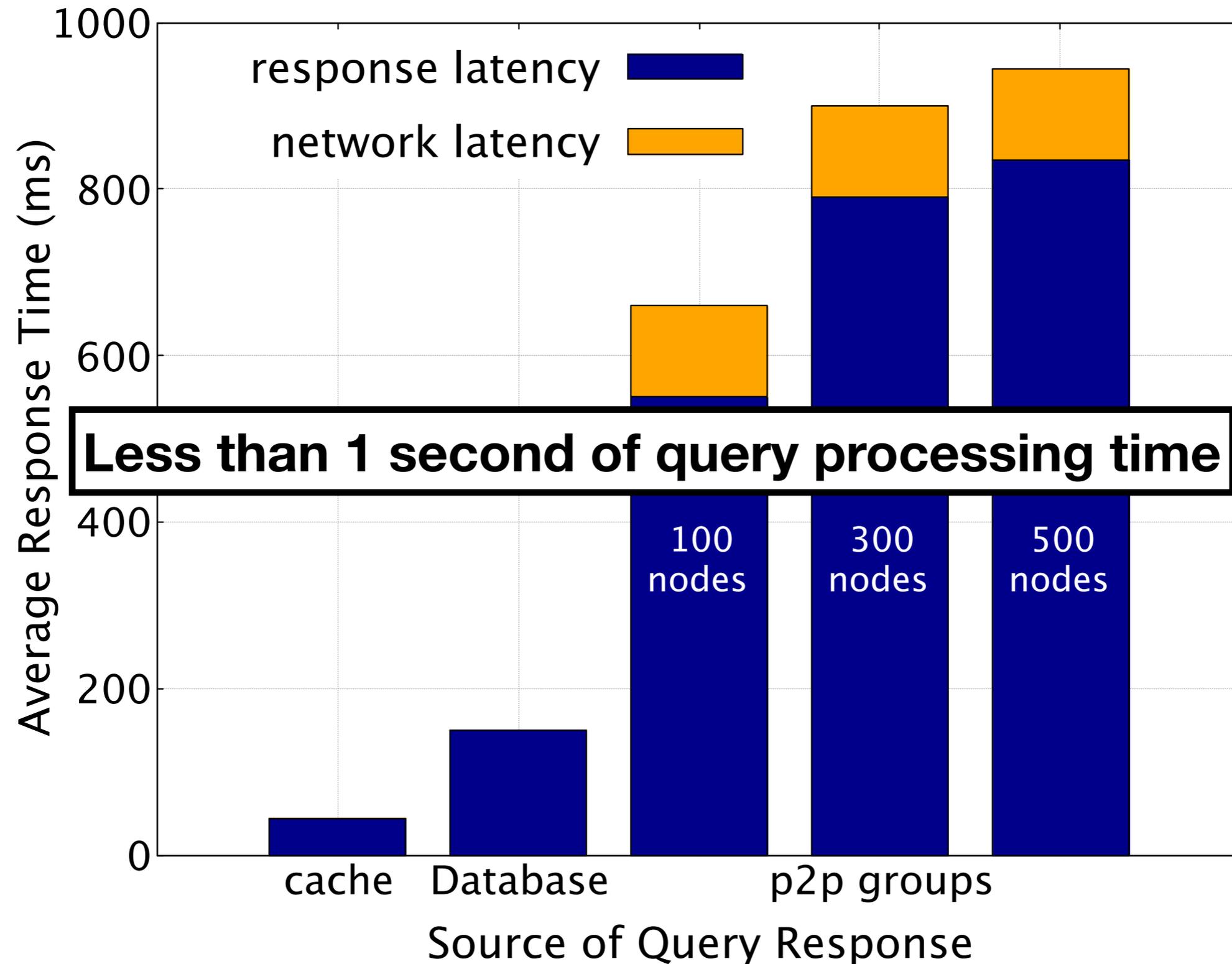
NodeFinder's Scalability (BW)



NodeFinder's Scalability (Query Time)



NodeFinder's Scalability (Query Time)



Future Work

Future Work

- Integration into existing cloud systems (e.g, OpenStack)

Future Work

- Integration into existing cloud systems (e.g, OpenStack)
- Perform thorough evaluation

Future Work

- Integration into existing cloud systems (e.g, OpenStack)
- Perform thorough evaluation
- Extensions: continuous monitoring

Conclusion

Conclusion

NodeFinder

General scalable search service through **attribute-based grouping, gossip-based coordination, and directed pulling.**

Conclusion

NodeFinder

General scalable search service through **attribute-based grouping, gossip-based coordination, and directed pulling.**

Open Questions

- How to implement NodeFinder as a general service (“plug-n-run”)?

Conclusion

NodeFinder

General scalable search service through **attribute-based grouping, gossip-based coordination, and directed pulling.**

Open Questions

- How to implement NodeFinder as a general service (“plug-n-run”)?
- How to decide the right group ranges (i.e., attribute cutoffs)?

Conclusion

NodeFinder

General scalable search service through **attribute-based grouping, gossip-based coordination, and directed pulling.**

Open Questions

- How to implement NodeFinder as a general service (“plug-n-run”)?
- How to decide the right group ranges (i.e., attribute cutoffs)?
- What level of dynamics can NodeFinder support? IoT?