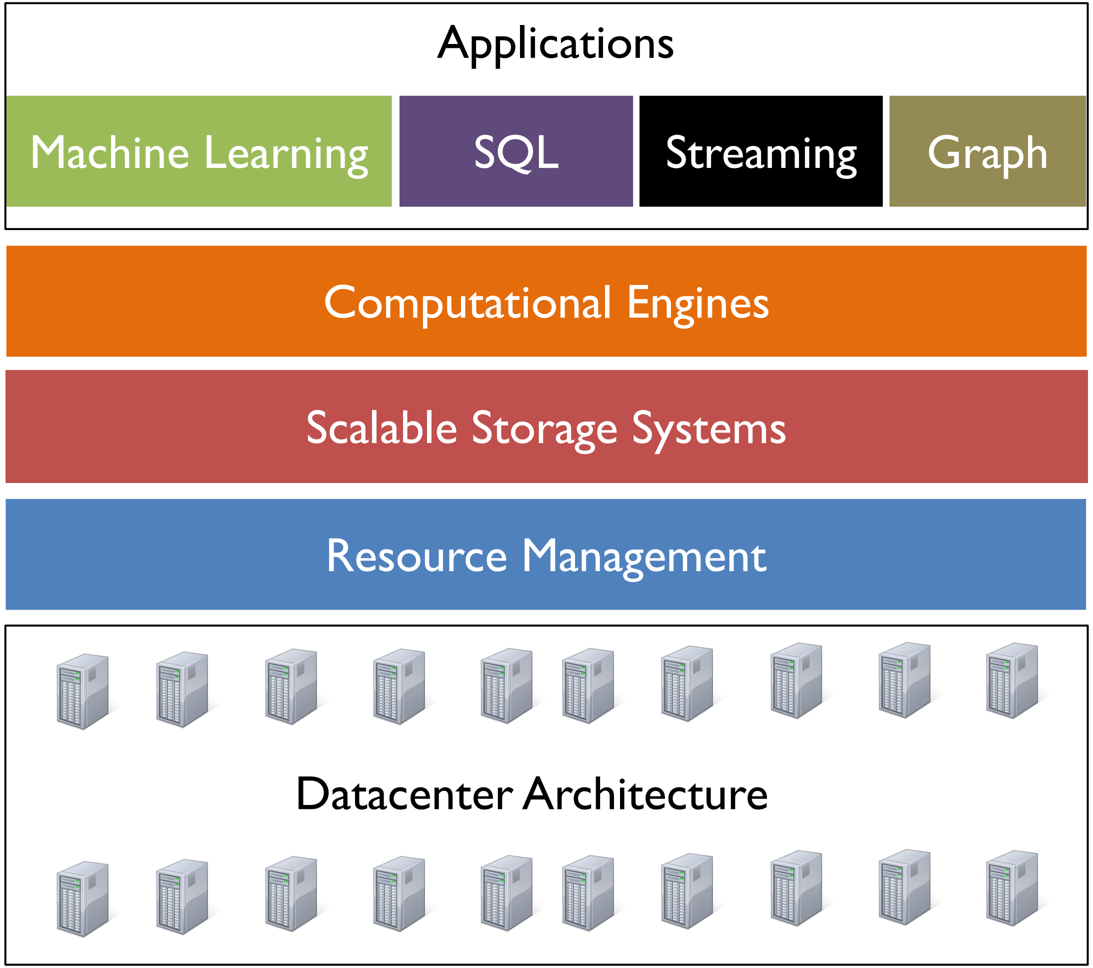
CS 744: PARAMETER SERVERS

Shivaram Venkataraman

Fall 2022

ADMINISTRIVIA

- Assignment 2 is due on Oct 12 (Wed) at 10am!
- Course project groups are also due same time?!


Applications
Machine Learning
SQL
Streaming
Graph
Computational Engines
Scalable Storage Systems
Resource Management
Datacenter Architecture

Machine Learning

PyTorch

Distributed DataParallel

Easy-to-use Interface

Model replicated on every worker

PipeDream

Model and pipeline-parallelism

Split model across workers

Commonalities?

PARAMETER SERVER: MOTIVATION

- Large training data 1TB to 1PB
- Models with 10^9 to 10^{12} parameters

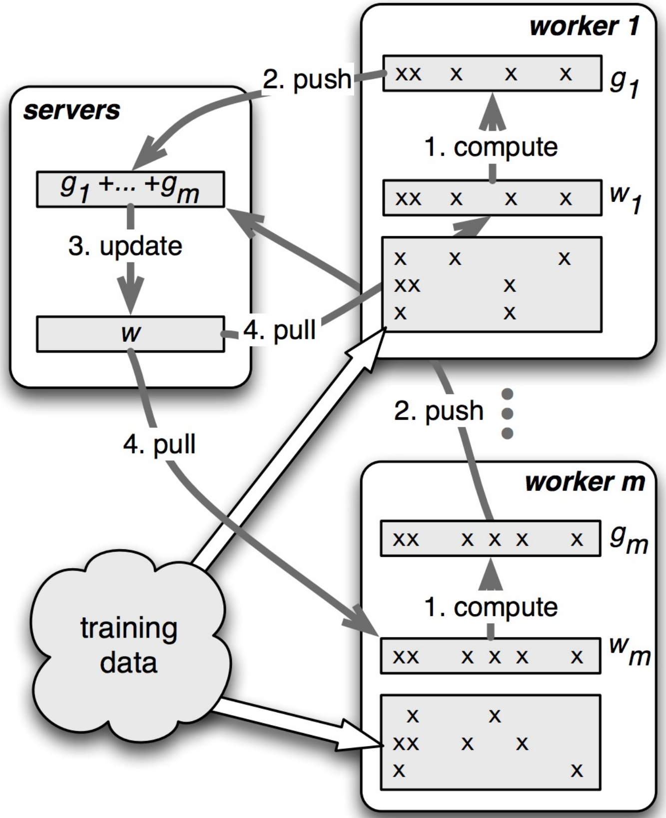
- Goals
  - Efficient communication
  - Flexible synchronization
  - Elastic Scalability
  - Fault Tolerance and Durability

EXAMPLE WORKLOAD

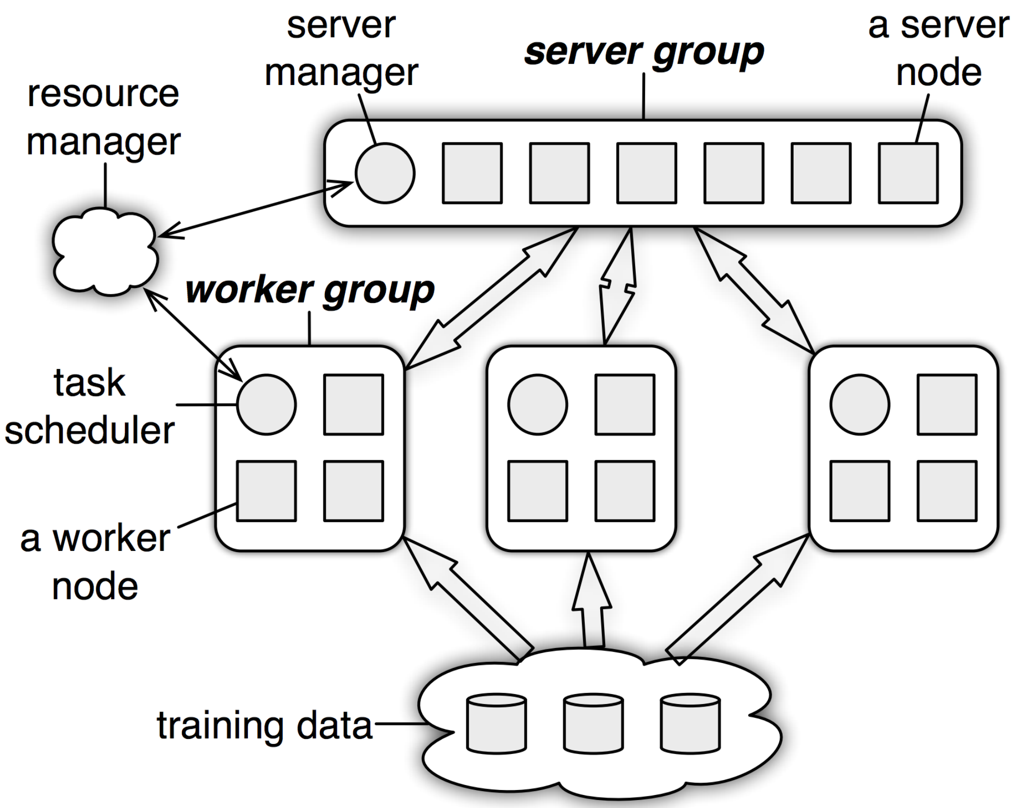
Ad Click Prediction

- Trillions of clicks per day
- Very sparse feature vectors

Computation flow



ARCHITECTURE



REPRESENTATION

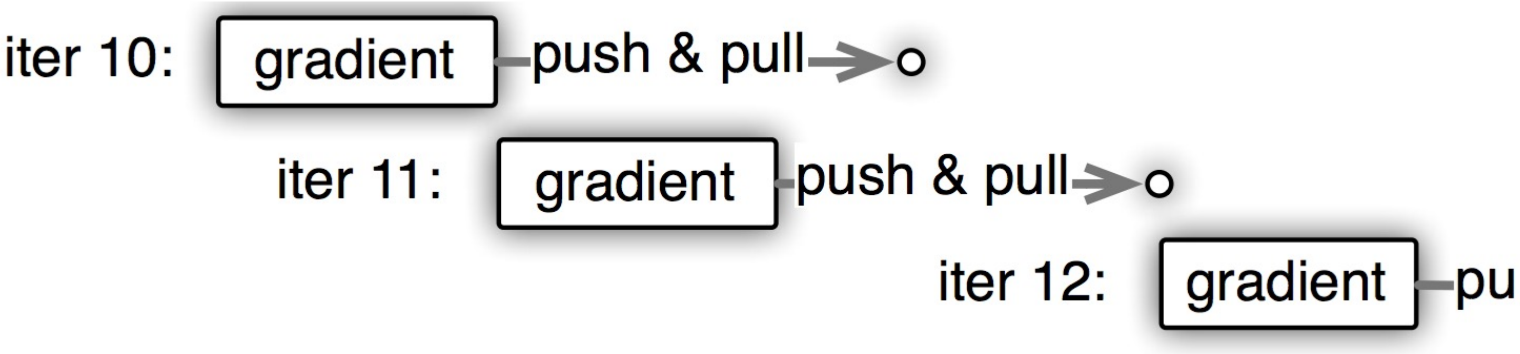
- Key value pairs e.g., (featureID, weight)

- Assume keys are ordered.
  Easier to apply linear algebra operations

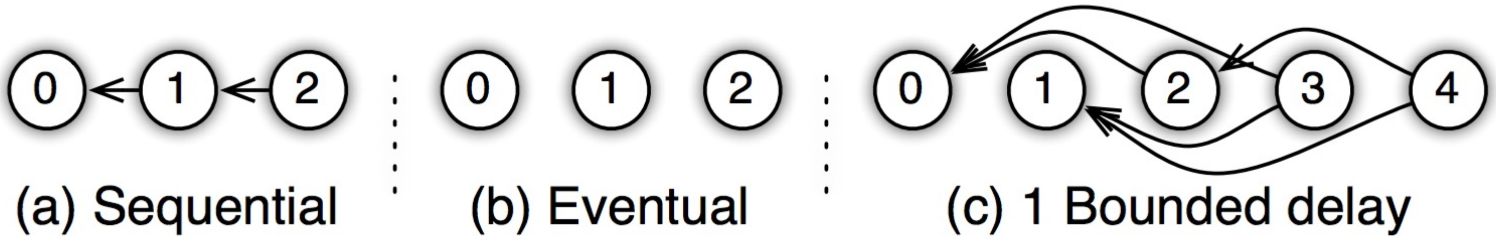
- Interface supports range push and pull
  w.push(R, dest)

- Support for user-defined functions on server-side

TASK DEPENDENCY



CONSISTENCY MODELS

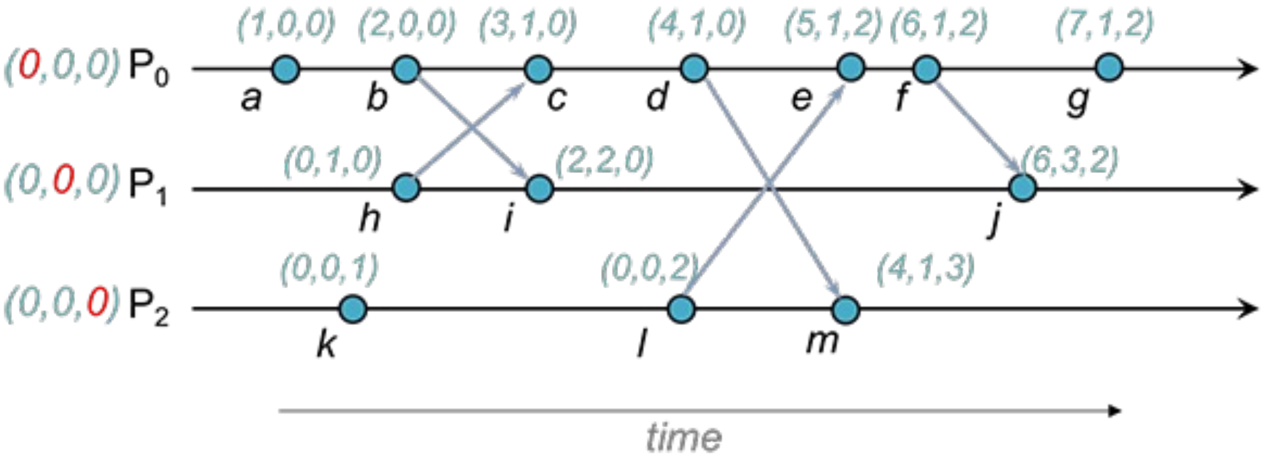


User defined filters

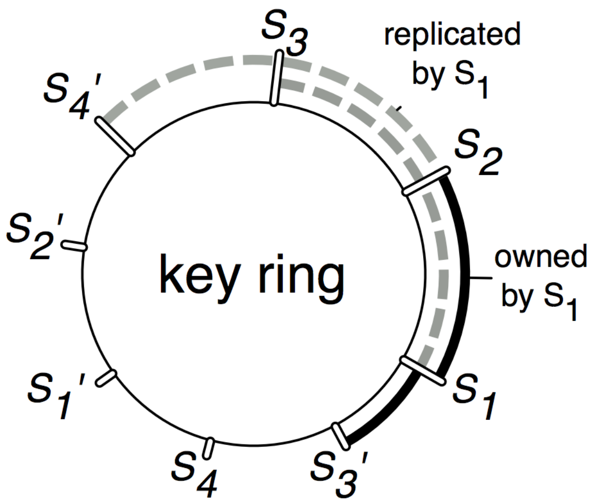
Significantly modified filter

KKT filter

IMPLEMENTATION: VECTOR CLOCKS



IMPLEMENTATION: REPLICATION



Replication after aggregation

FAULT TOLERANCE

1. Server manager assigns the new node a key range to serve as master.

2. The node fetches the range of data to maintains as master and k additional ranges to keep as slave.

3. The server manager broadcasts the node changes. The recipients of the message may shrink their own data

SPARSE LR

Algorithm 3 Delayed Block Proximal Gradient [31]

Scheduler:

Partition features into b ranges $\mathcal{R}_1, \ldots, \mathcal{R}_b$
for $t = 0$ **to** T **do**
    Pick random range $\mathcal{R}_{i_t}$ and issue task to workers
end for

Worker r at iteration t

Wait until all iterations before $t - \tau$ are finished
Compute first-order gradient $g_r^{(t)}$ and diagonal second-order gradient $u_r^{(t)}$ on range $\mathcal{R}_{i_t}$
Push $g_r^{(t)}$ and $u_r^{(t)}$ to servers with the KKT filter
Pull $w_r^{(t+1)}$ from servers

Servers at iteration t

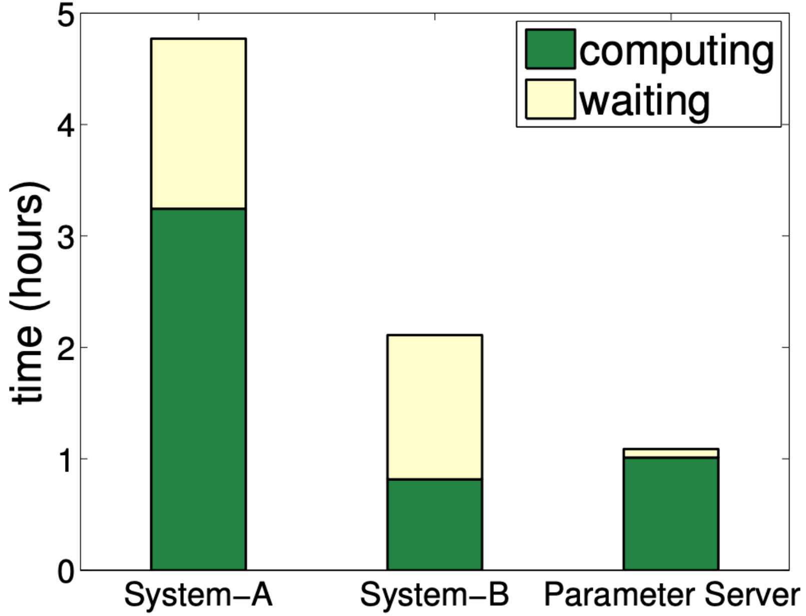
Aggregate gradients to obtain $g^{(t)}$ and $u^{(t)}$
Solve the proximal operator

$$w^{(t+1)} \leftarrow \underset{u}{\operatorname{argmin}}\, \Omega(u) + \frac{1}{2\eta}\|w^{(t)} - \eta g^{(t)} + u\|_H^2,$$

where $H = \operatorname{diag}(h^{(t)})$ and $\|x\|_H^2 = x^T H x$

DISCUSSION

https://forms.gle/qPX1bBCAsd2fhL2i6



5
4
3
2
1
0
time (hours)
computing
waiting
System−A
System−B
Parameter Server

What are some of the downsides of using PS compared to implementing Gradient Descent in PyTorch / Spark?

How would you integrate PS with a resource manager like Mesos? What would be some of the challenges?

NEXT STEPS

Next class: Gavel

Assignment 2 is due soon!