

Lecture 16.3

Spark Lineage and Checkpoints

EN 600.320/420

Instructor: Randal Burns

9 April 2018



Department of Computer Science, *Johns Hopkins University*

Lineage and Recoverability

- Spark in the presence of failures:
 - Identify partitions of data (in an RDD) that have failed
 - Recompute the failed partitions using lineage
 - Parallelize recomputation (using Spark)
 - Easy because all RDDs are immutable
- Does not require *checkpoint/restart* or *rollback*
 - *Checkpoint* = save required memory (application state) to persistent storage sufficient to restart the computation
 - *Restart* = restart computation from a checkpoint
 - *Rollback* = Undo changes made to memory associated with computations that have failed or did not complete
 - All concepts related to managing a writeable memory related to computational progress in the presence of failures



Checkpointing versus Lineage

- It is desirable to checkpoint when:
 - Lineages become long (in dependencies)
 - Dependencies become wide
- Spark's default is to use the initial data load as the only checkpoint and restart from that checkpoint
 - But that depends on ability to localize recomputation to partitions
 - And on the length of the lineage



Example 2: PageRank

- Long lineage
 - Need to recompute entire on failure
- Checkpoint (save ranks)
 - Limit restart work

```
val links = spark.textFile(...).map(...).persist()
var ranks = // RDD of (URL, rank) pairs
for (i <- 1 to ITERATIONS) {
  // Build an RDD of (targetURL, float) pairs
  // with the contributions sent by each page
  val contribs = links.join(ranks).flatMap {
    (url, (links, rank)) =>
      links.map(dest => (dest, rank/links.size))
  }
  // Sum contributions by URL and get new ranks
  ranks = contribs.reduceByKey((x,y) => x+y)
    .mapValues(sum => a/N + (1-a)*sum)
}
```

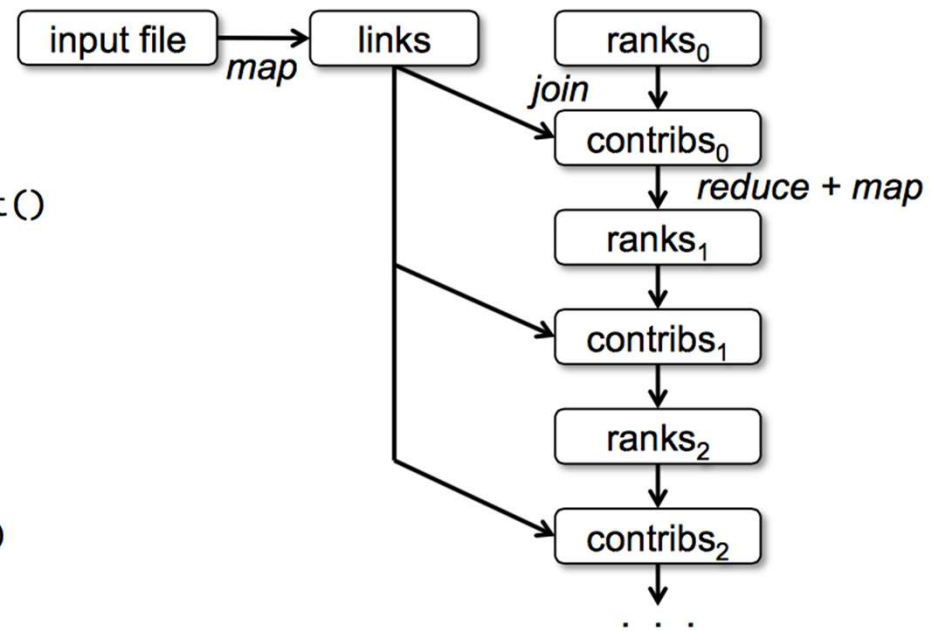


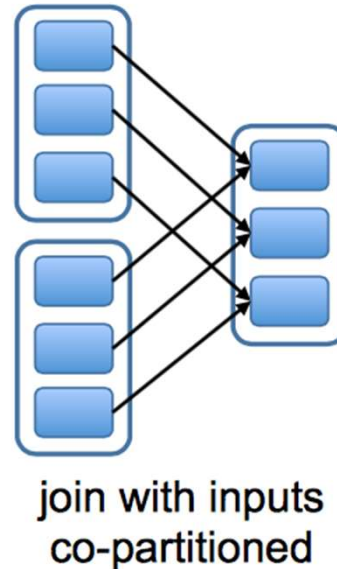
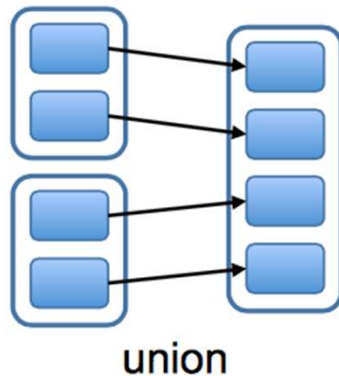
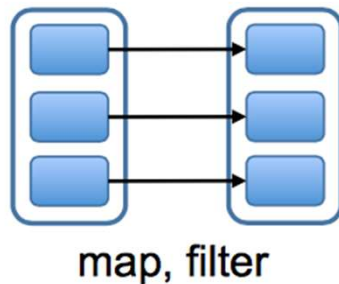
Figure 3: Lineage graph for datasets in PageRank.



Wide Dependencies

- Join, cross product, cogroup
 - Removes partition locality
- Checkpoint before or after wide dependency, because the entire RDD would need to be rebuilt

Narrow Dependencies:



Wide Dependencies:

