CLOUD COMPUTING APPLICATIONS

Anchoring and Spout Replay

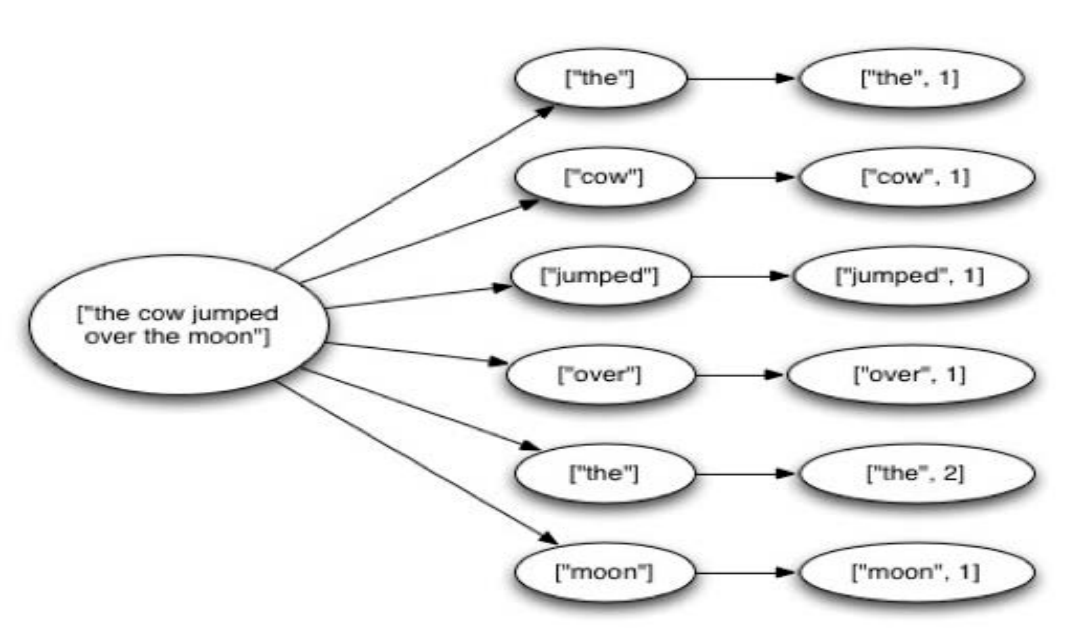
Roy Campbell & Reza Farivar

Guaranteeing Message Processing in Three Tasty Flavors

1. None (like the old S4)
2. *At Least Once: tuple trees, anchoring, and spout replay*
3. Exactly Once (like Hadoop or Puma)

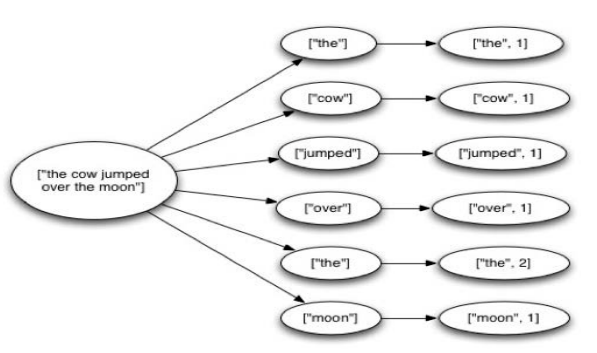
Tuple Tree

Tuple Tree



Tuple Tree

- A spout tuple is not fully processed until all tuples in the tree have been completed
- If the tuple tree is not completed within a specified timeout, the spout tuple is replayed
- Uses acker tasks to keep track of tuple progress



Anchoring

Reliability API for the user:

```
public void execute(Tuple tuple) {
    String sentence = tuple.getString(0);
    for(String word: sentence.split(" ")) {
        _collector.emit(tuple, new Values(word));
    }
    _collector.ack(tuple);
}
```

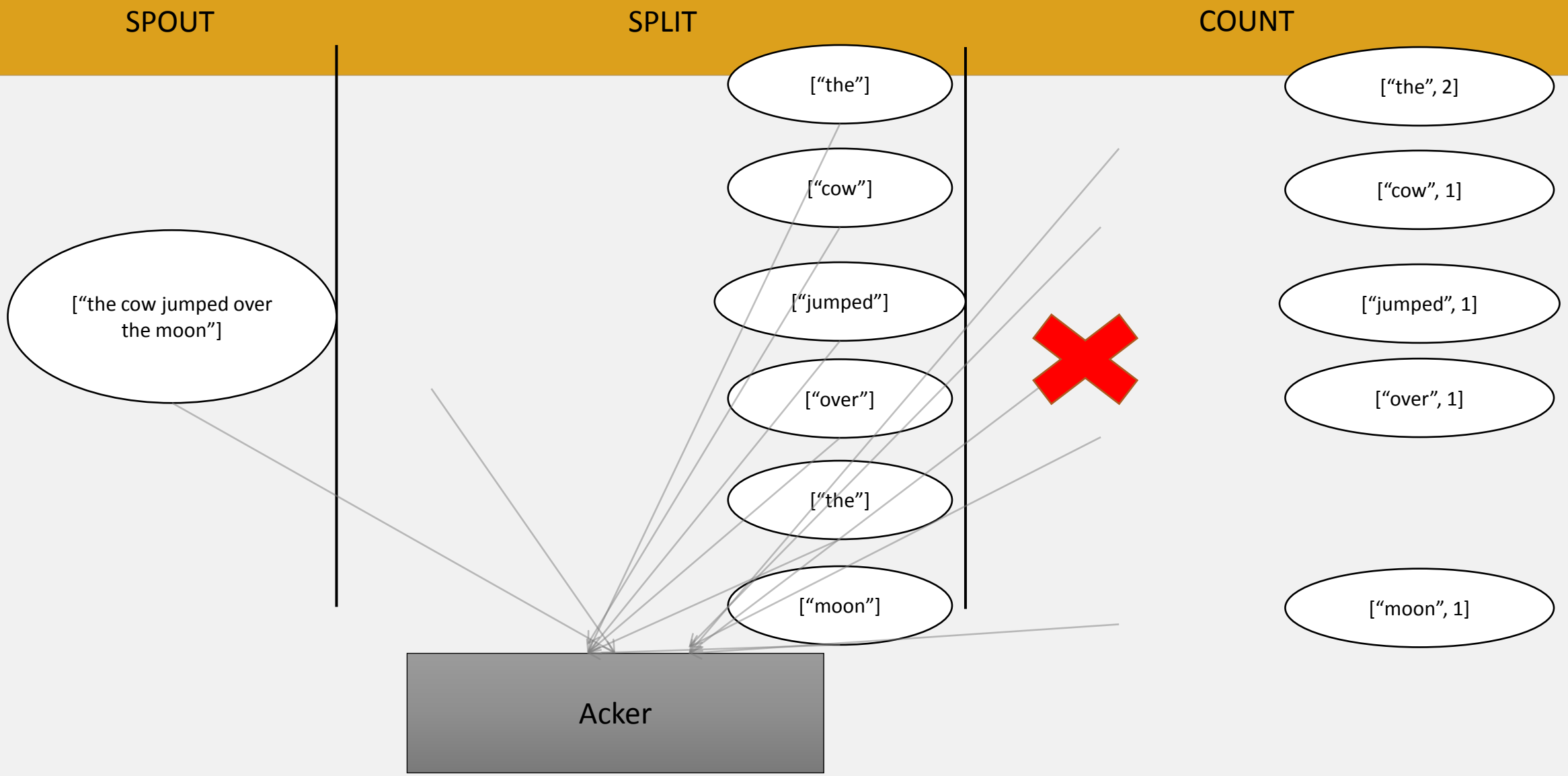
"Anchoring" creates a new edge in the tuple tree

Marks a single node in the tree as complete

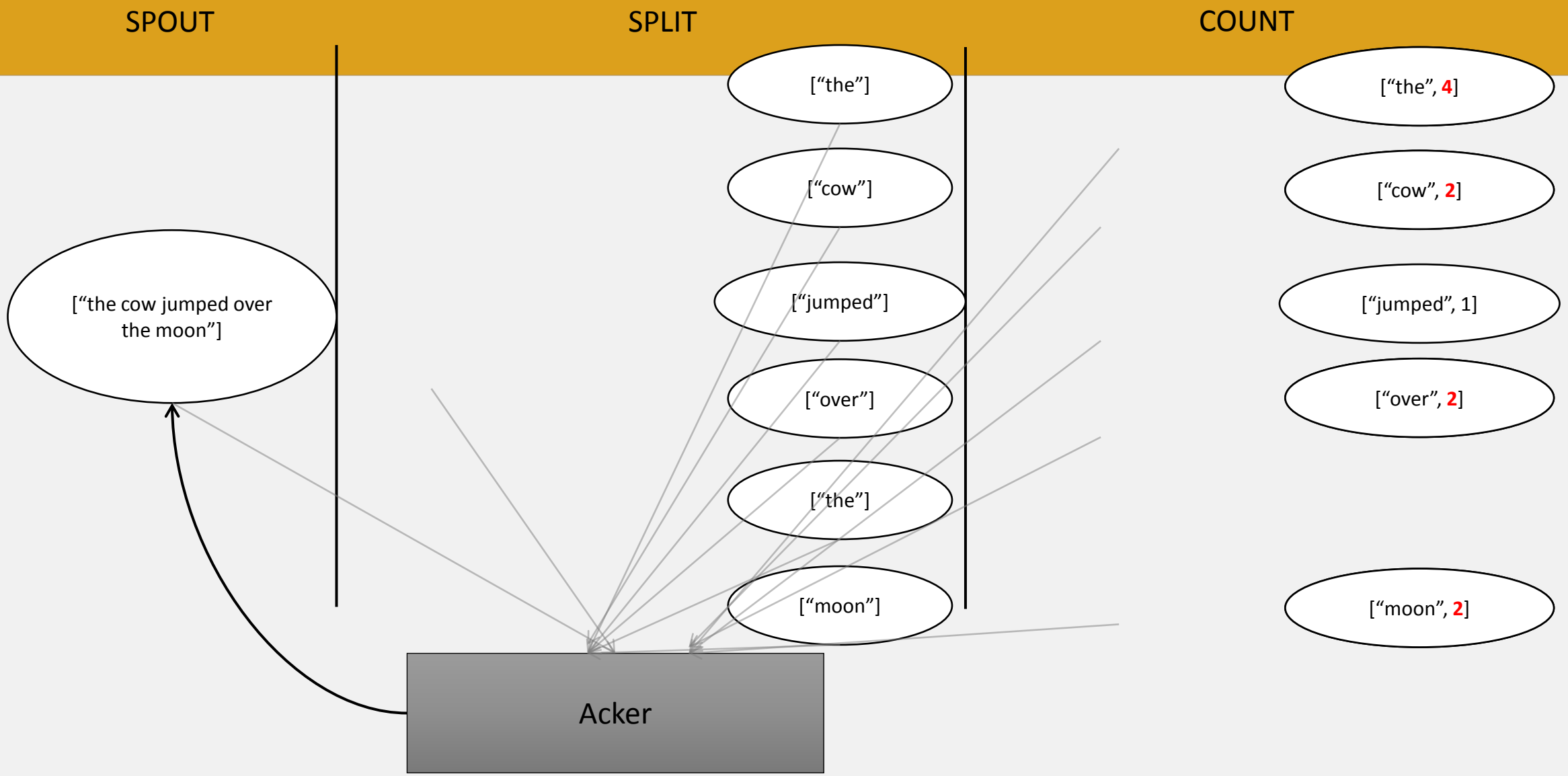
At Least Once

- What happens if there is a failure?
  - You can double process events
  - This is not so critical if you have something like Hadoop to back you up and correct the issue later
  - Or if you are looking at statistical trends and replay does not happen that often
- This requires you to have a spout that supports replay. Not all messaging infrastructure does

Example


SPOUT
SPLIT
COUNT
["the cow jumped over the moon"]
["the"]
["cow"]
["jumped"]
["over"]
["the"]
["moon"]
["the", 2]
["cow", 1]
["jumped", 1]
["over", 1]
["moon", 1]
Acker

Example


SPOUT
SPLIT
COUNT
["the cow jumped over the moon"]
["the"]
["cow"]
["jumped"]
["over"]
["the"]
["moon"]
["the", 4]
["cow", 2]
["jumped", 1]
["over", 2]
["moon", 2]
Acker