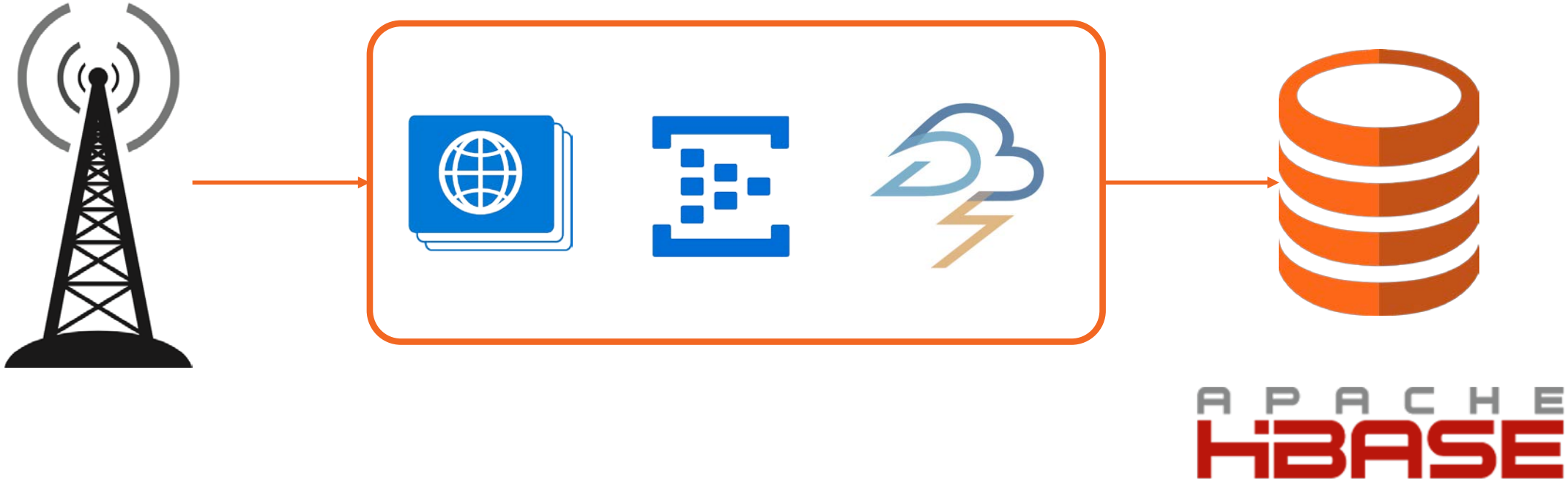
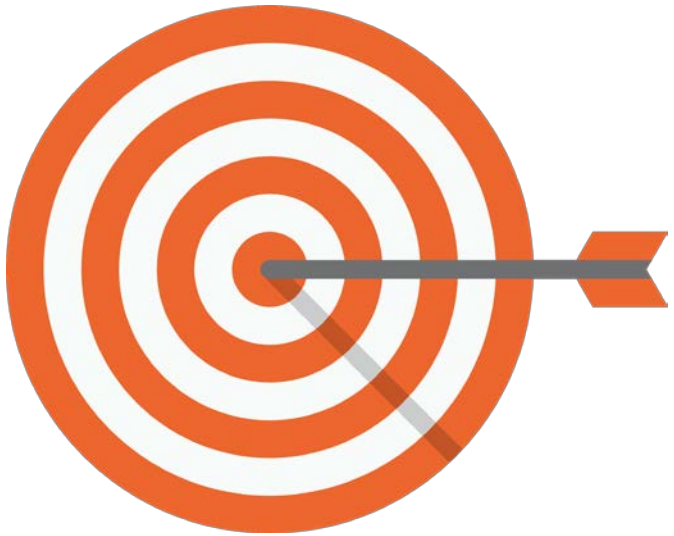
Processing Timing Events with Storm

Elton Stoneman

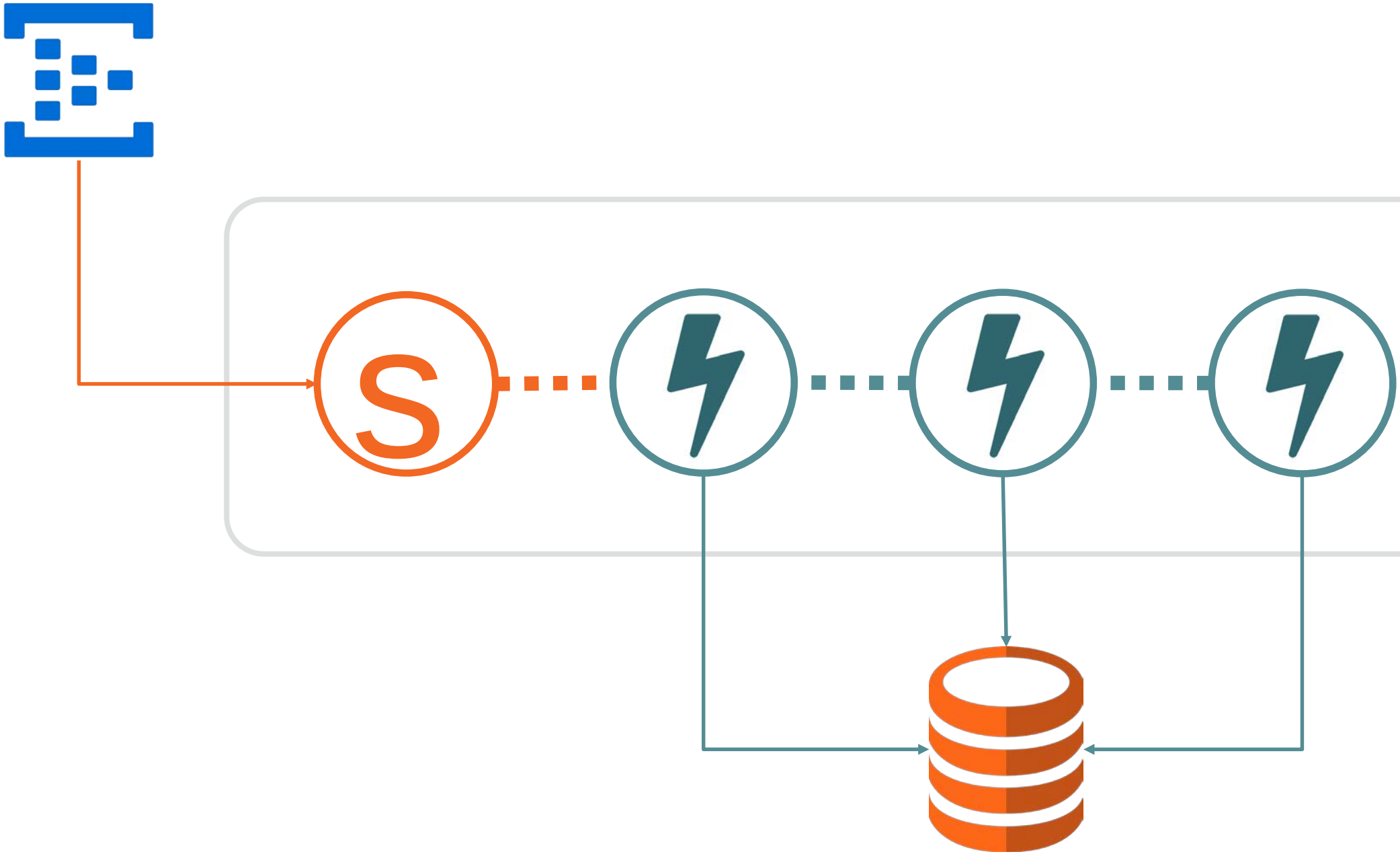
@EltonStoneman | blog.sixeyed.com

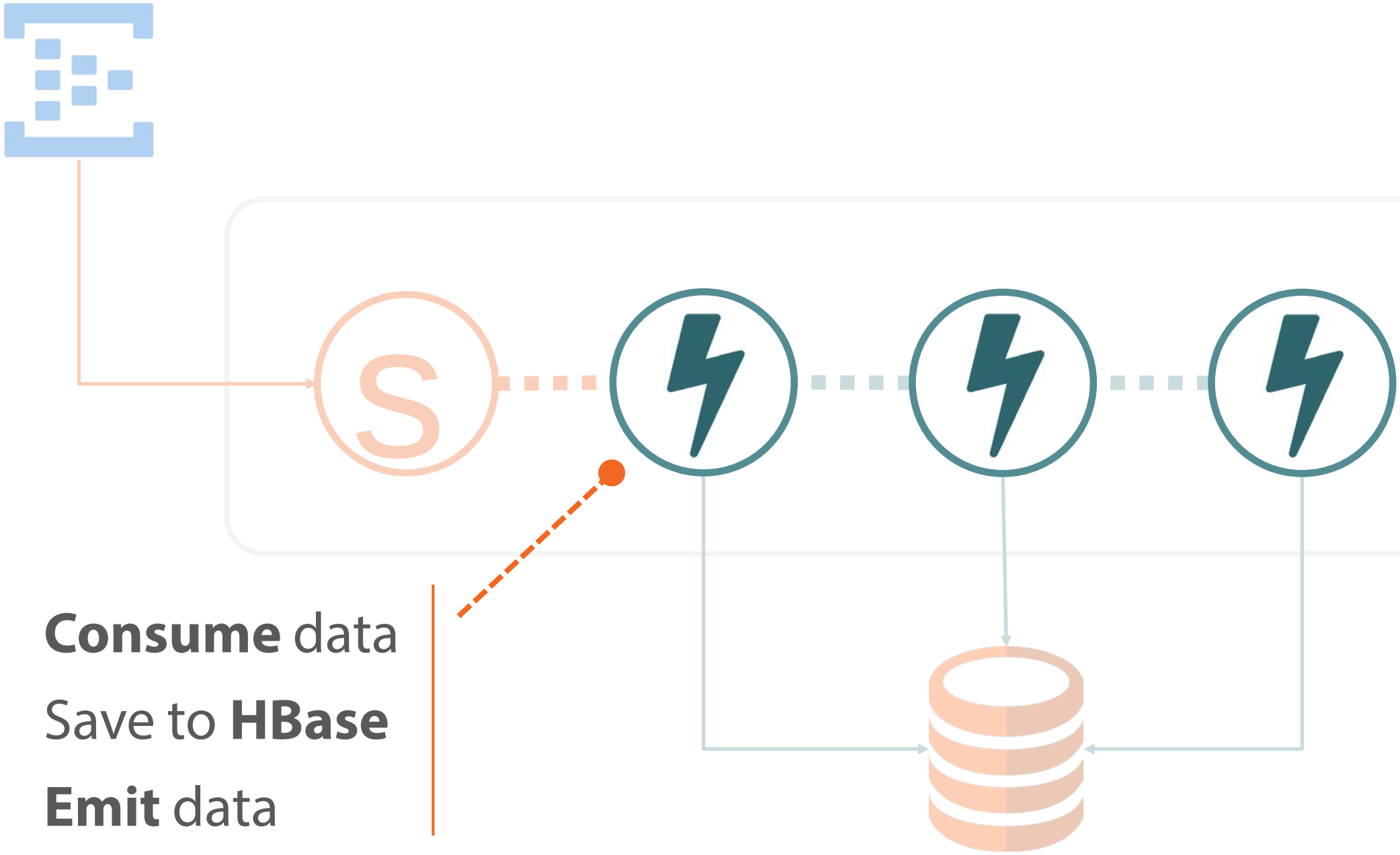


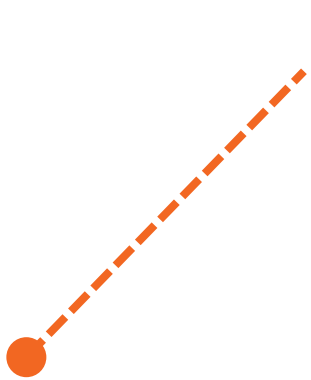
Module Goals



- Build Race Timing Components
- Configure Topology
- Deploy to HDInsight
- Verify End-to-End




S
Consume data
Save to **HBase**
Emit data



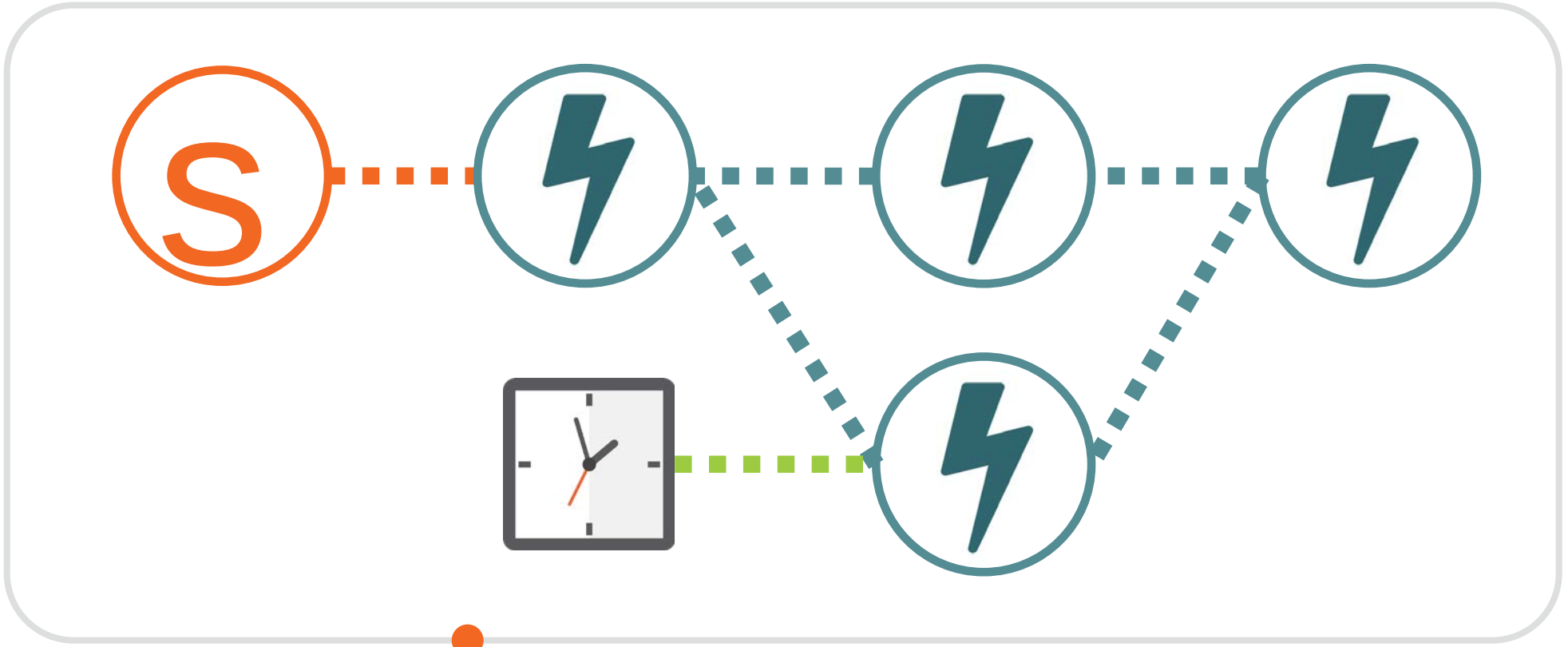
Parallel

Low-latency

Stream processing

APACHE
STORM™

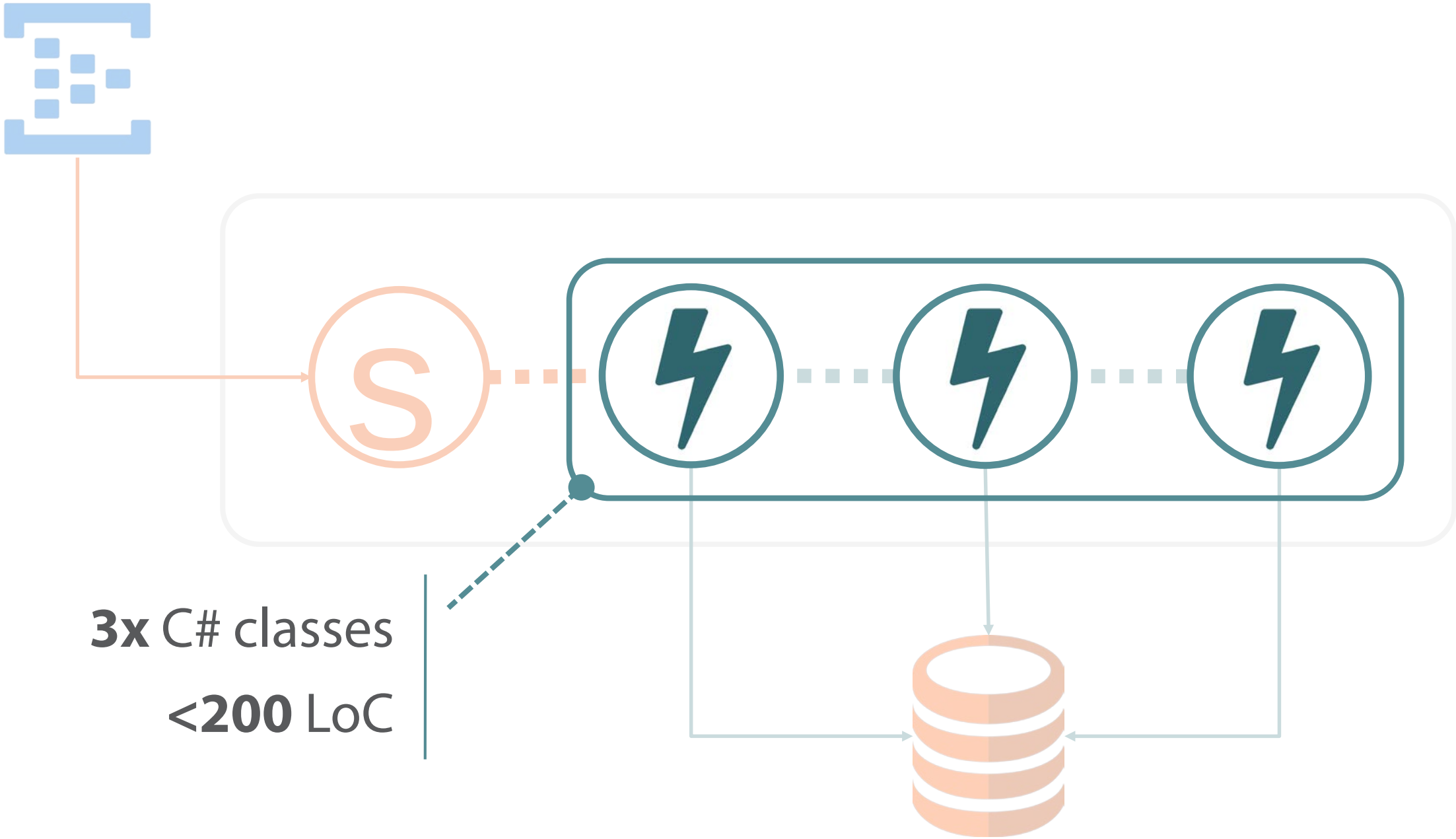
Distributed • Resilient • Real-time

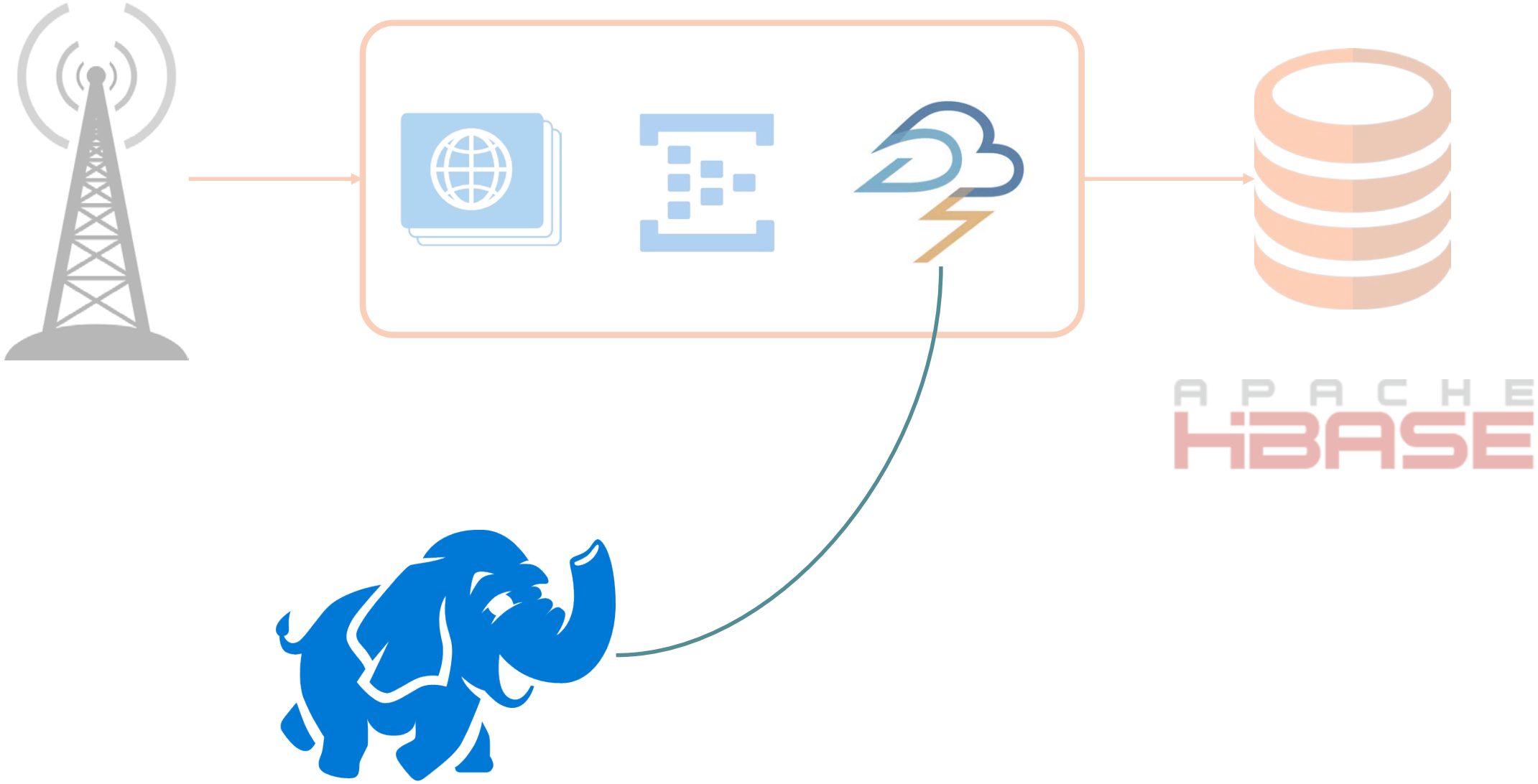


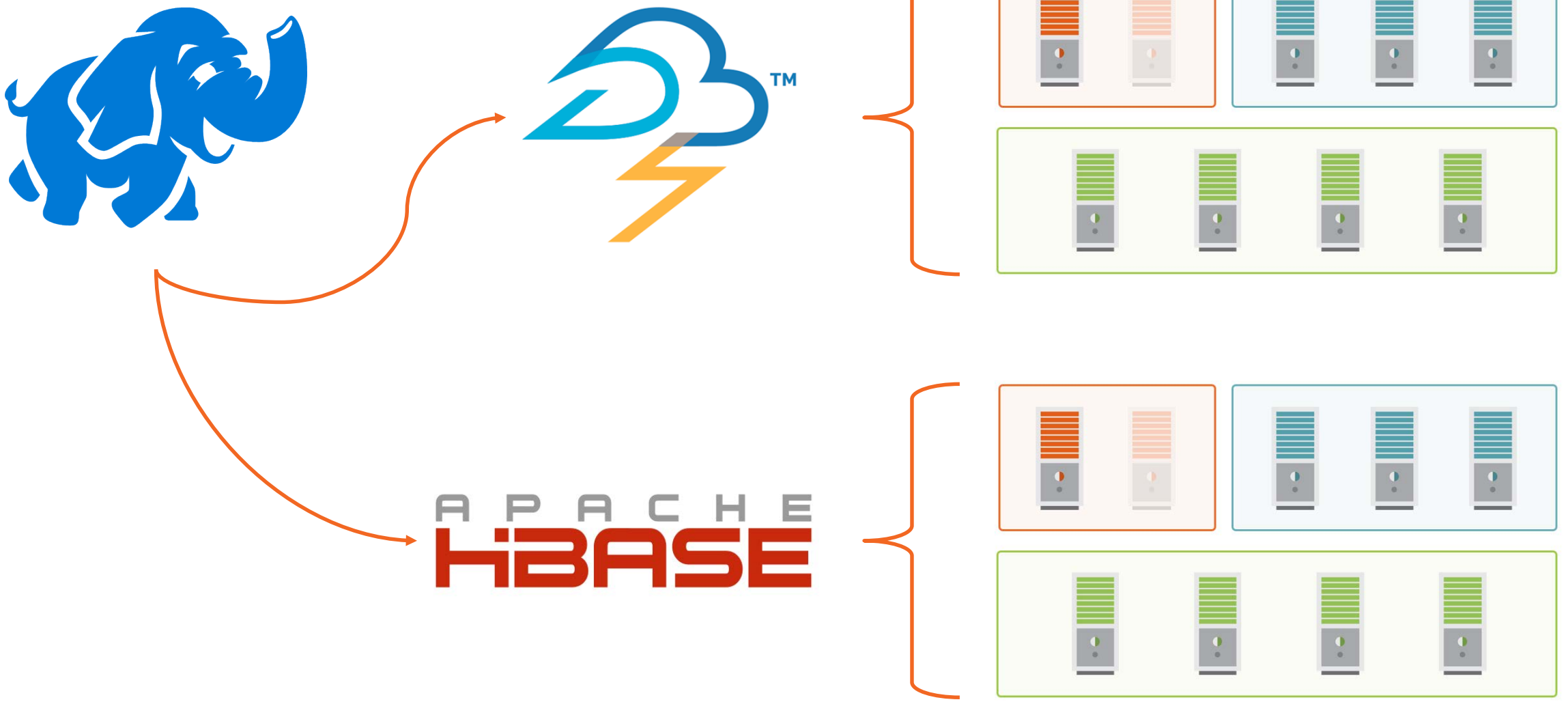
Custom components

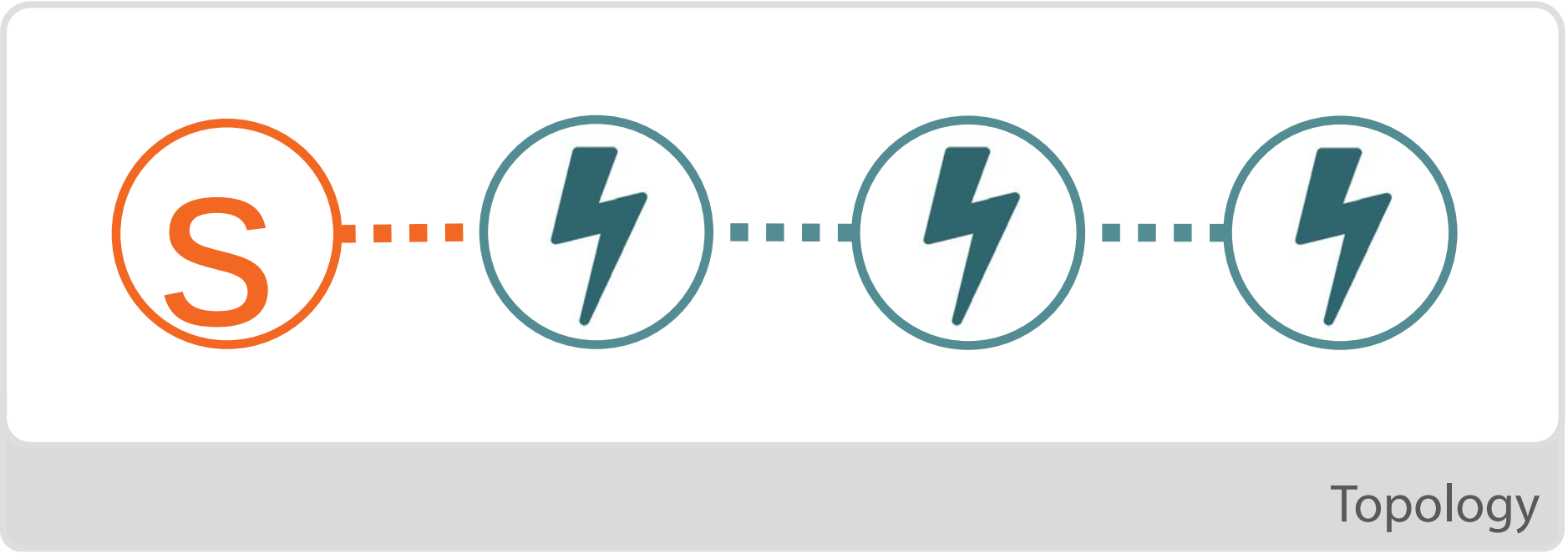
Packaged components

System components

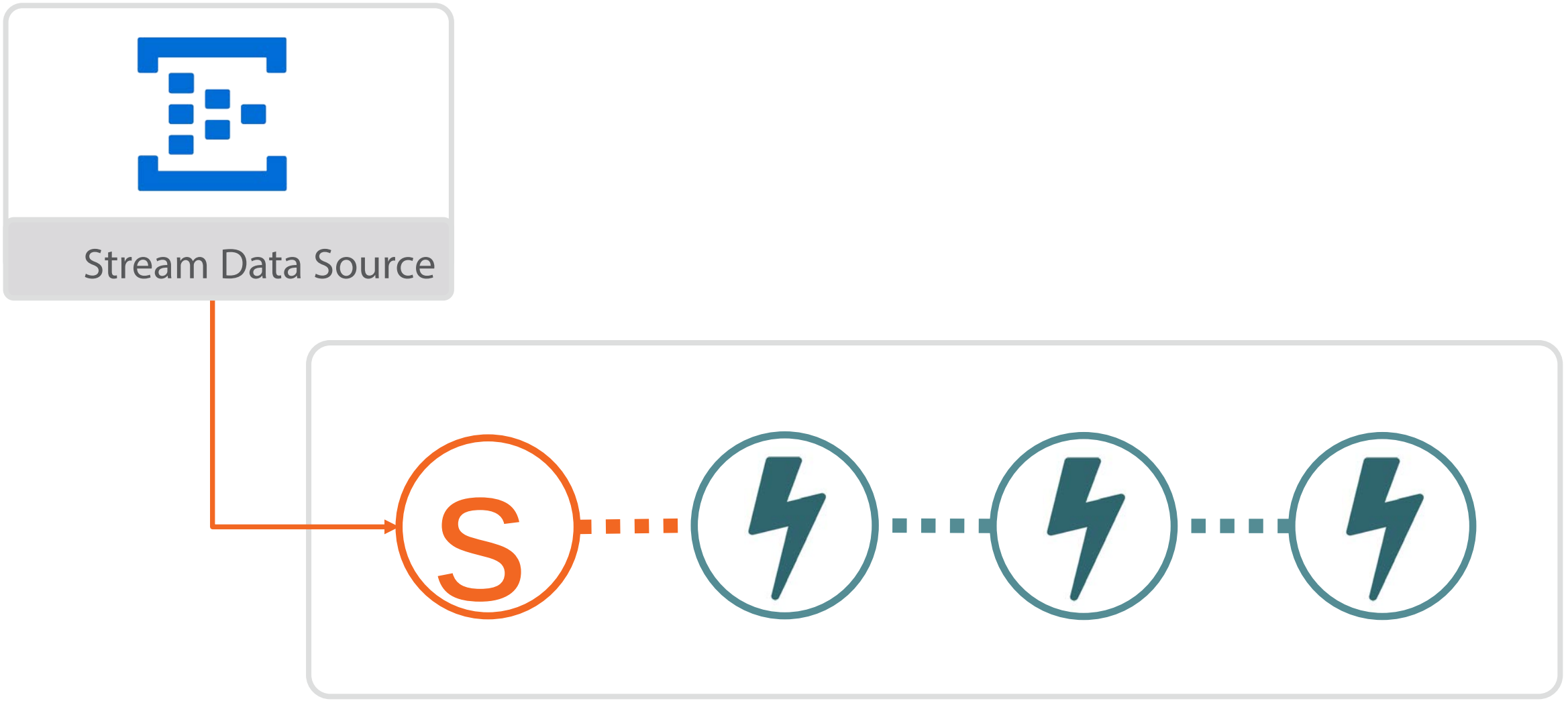


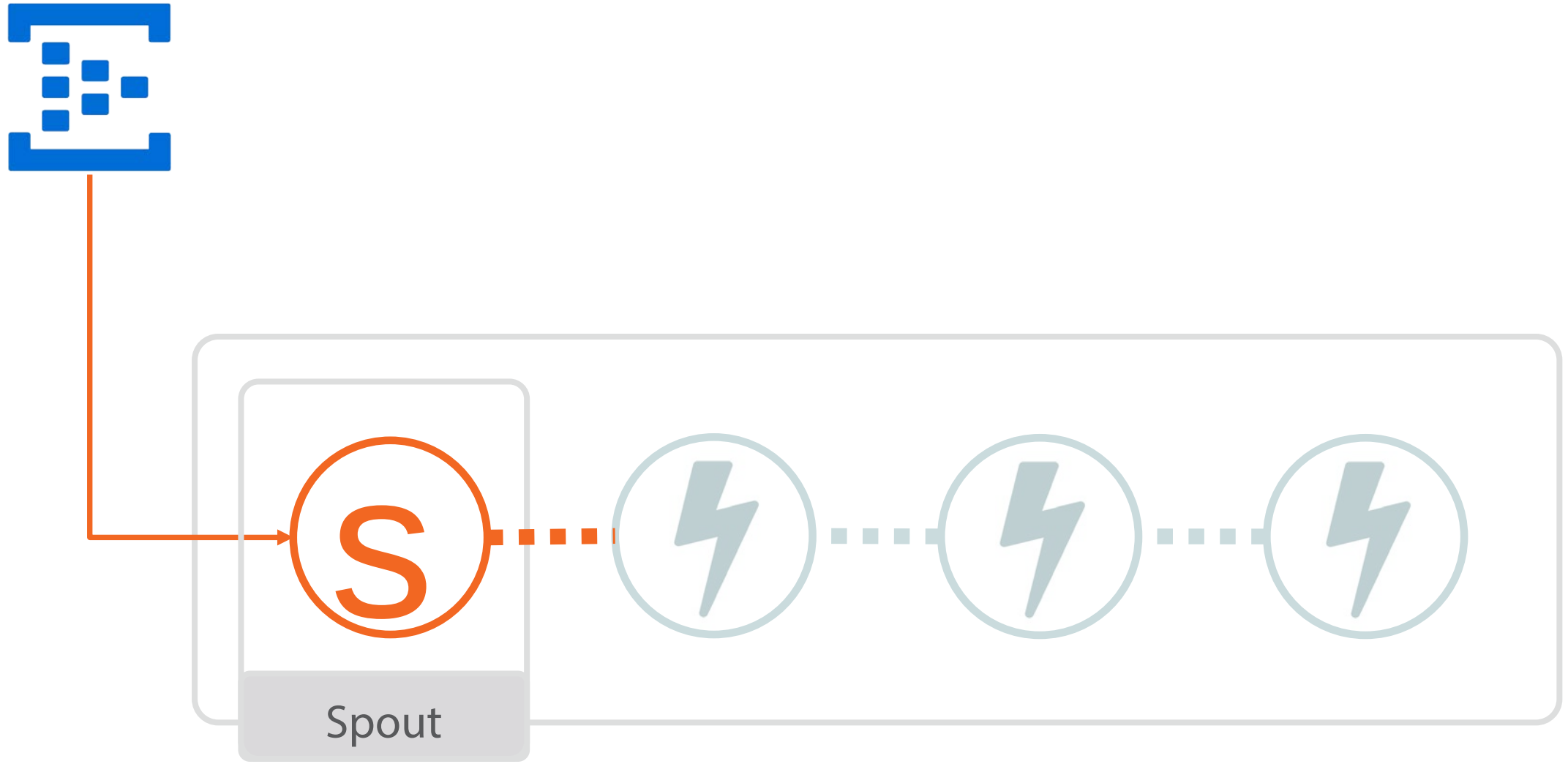






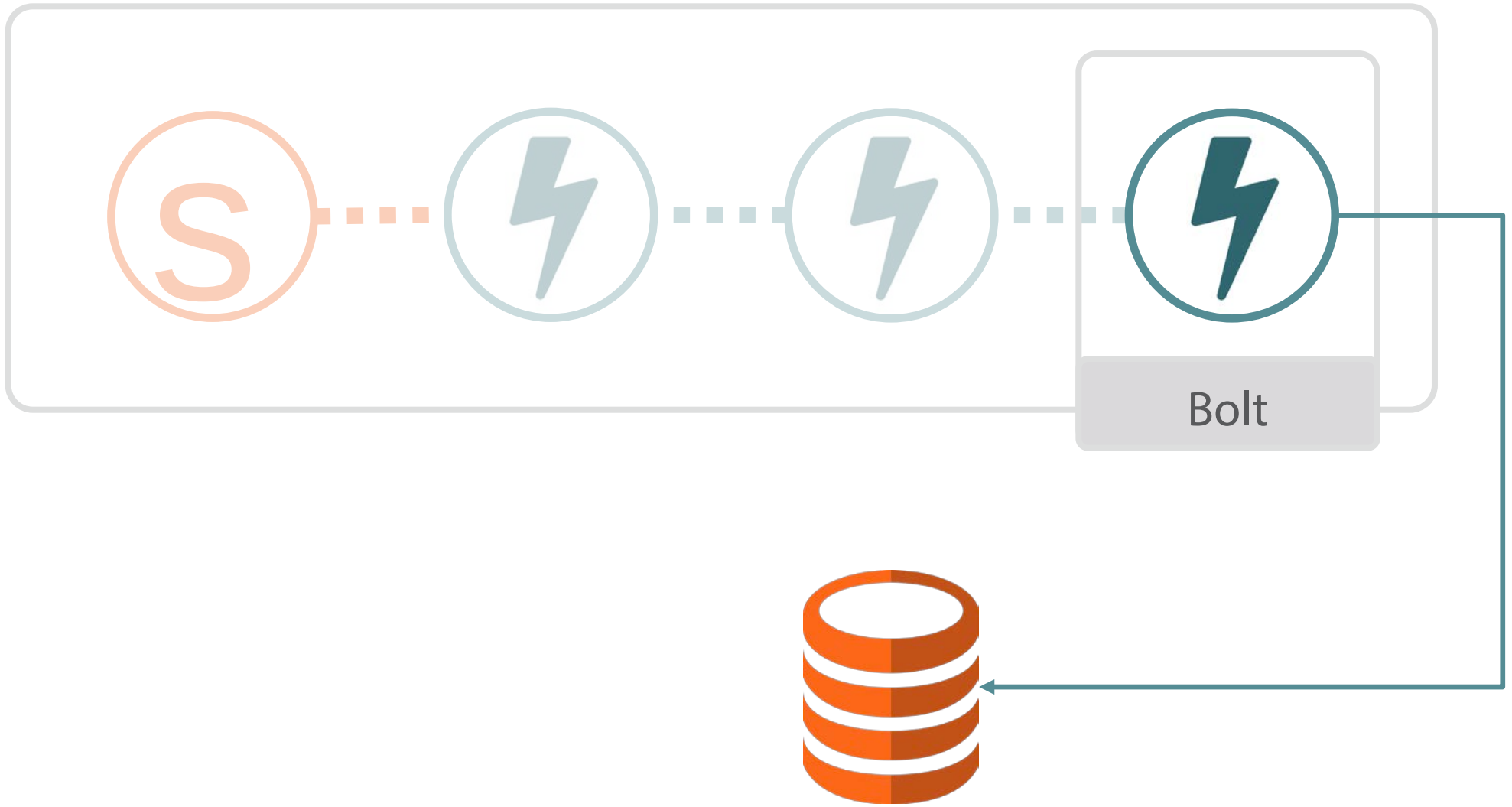
Topology

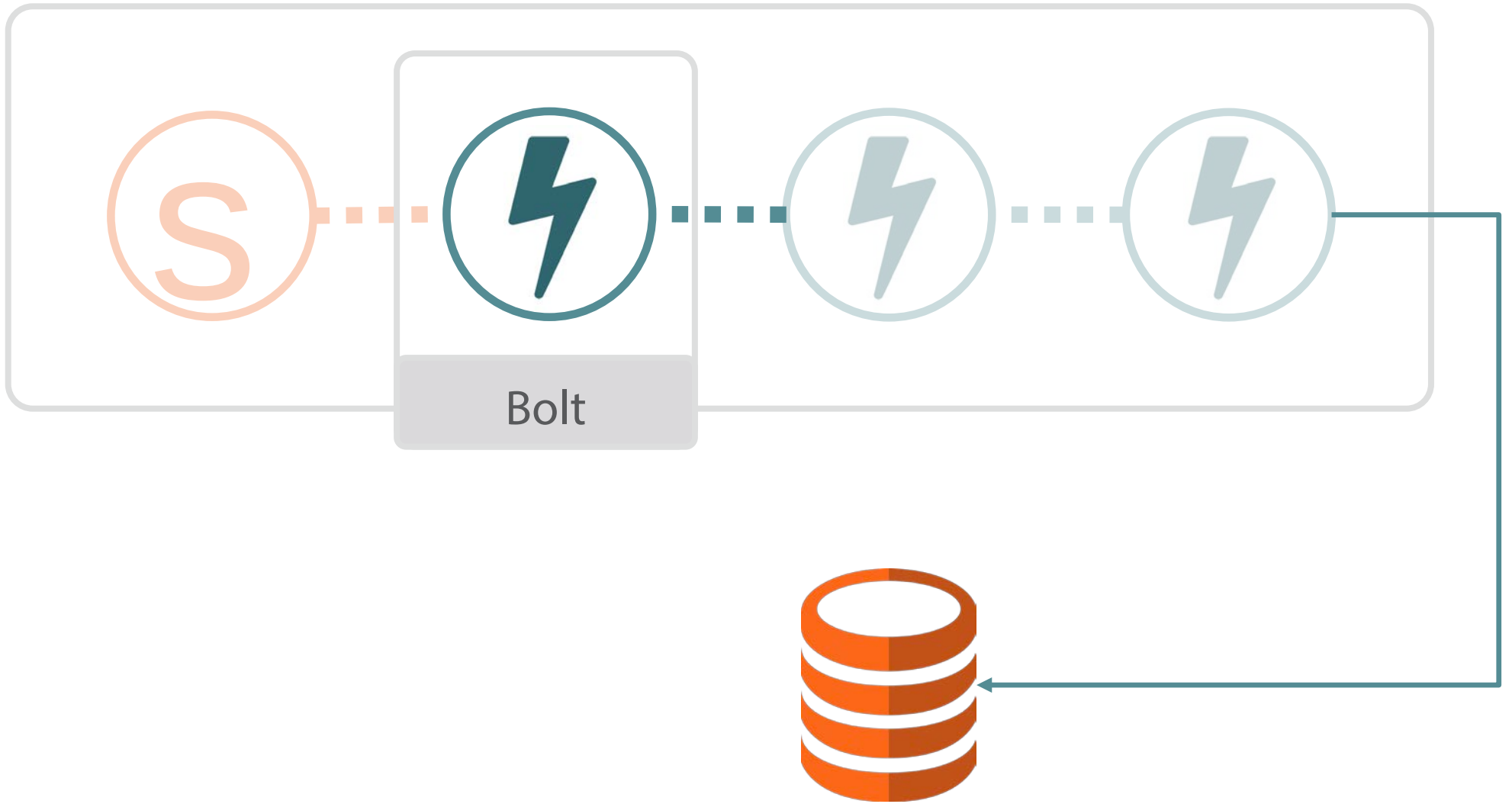


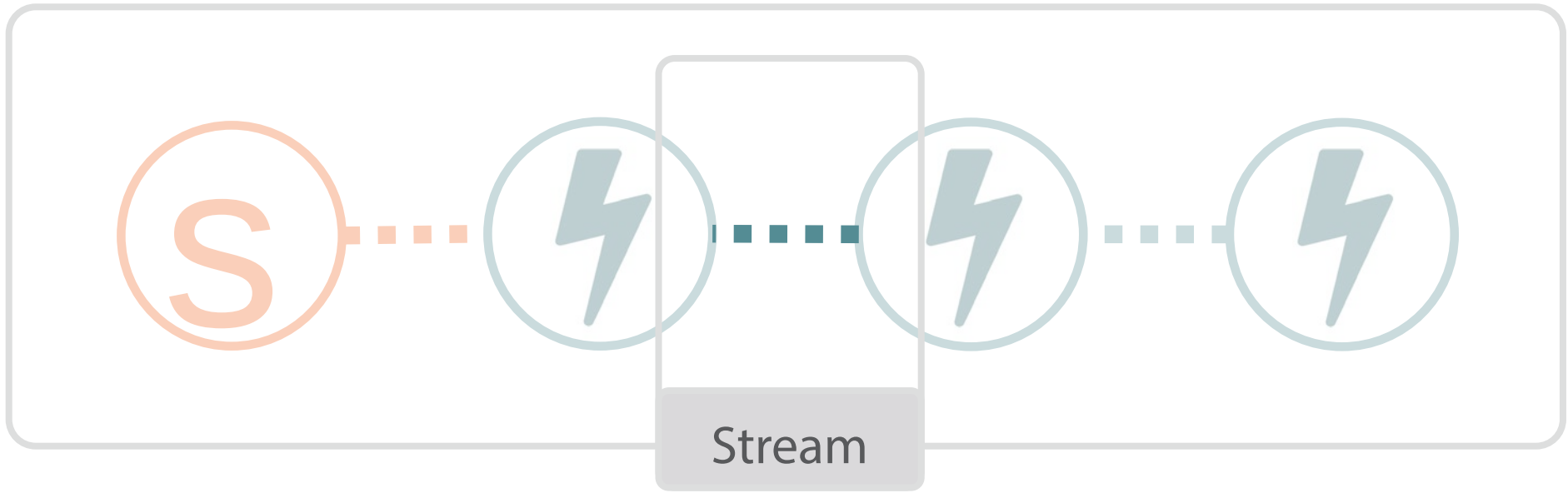


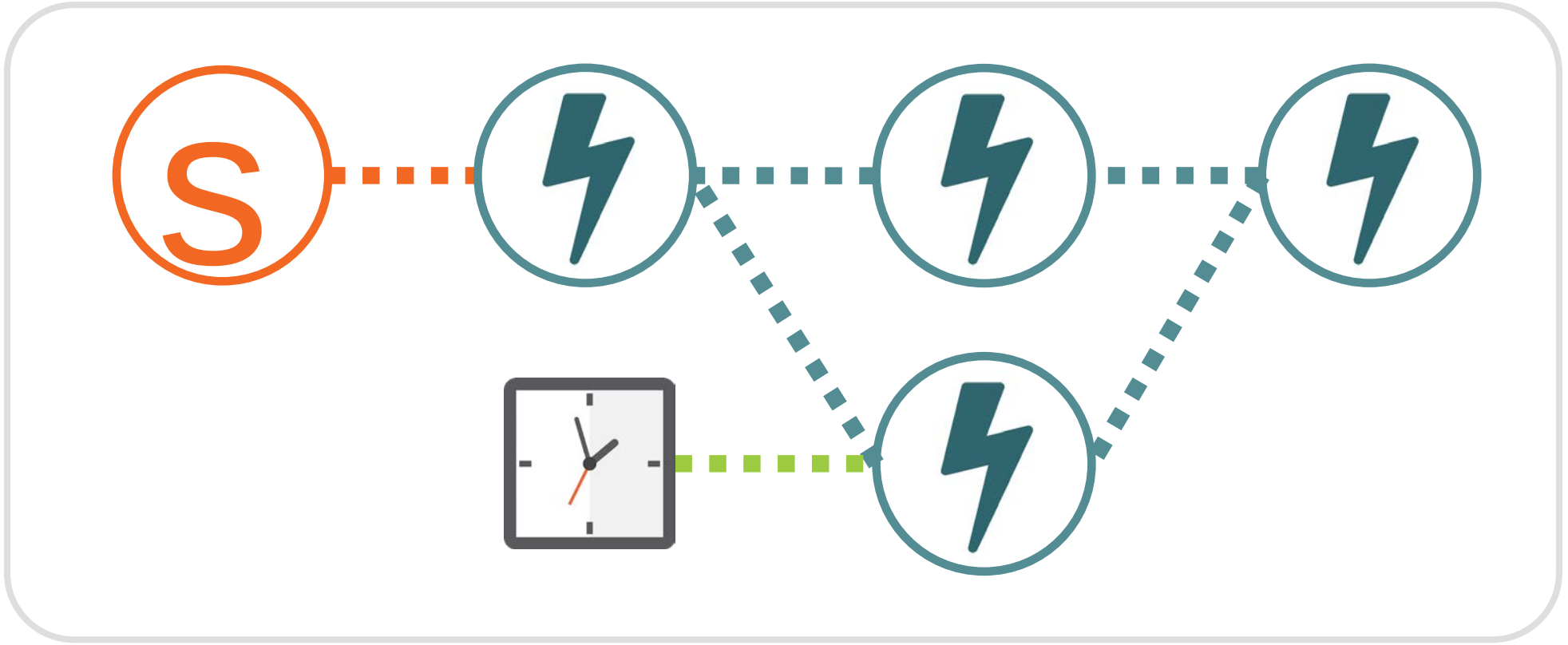
Tuple

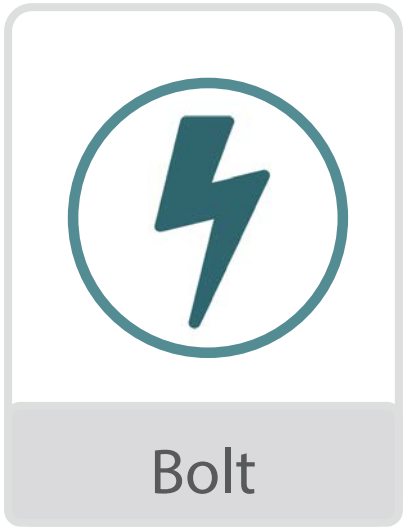
eventPayload	{ "timerId" : "eb9a5", "racerId" : "2134e13", "timestamp" : 1444052588000 }







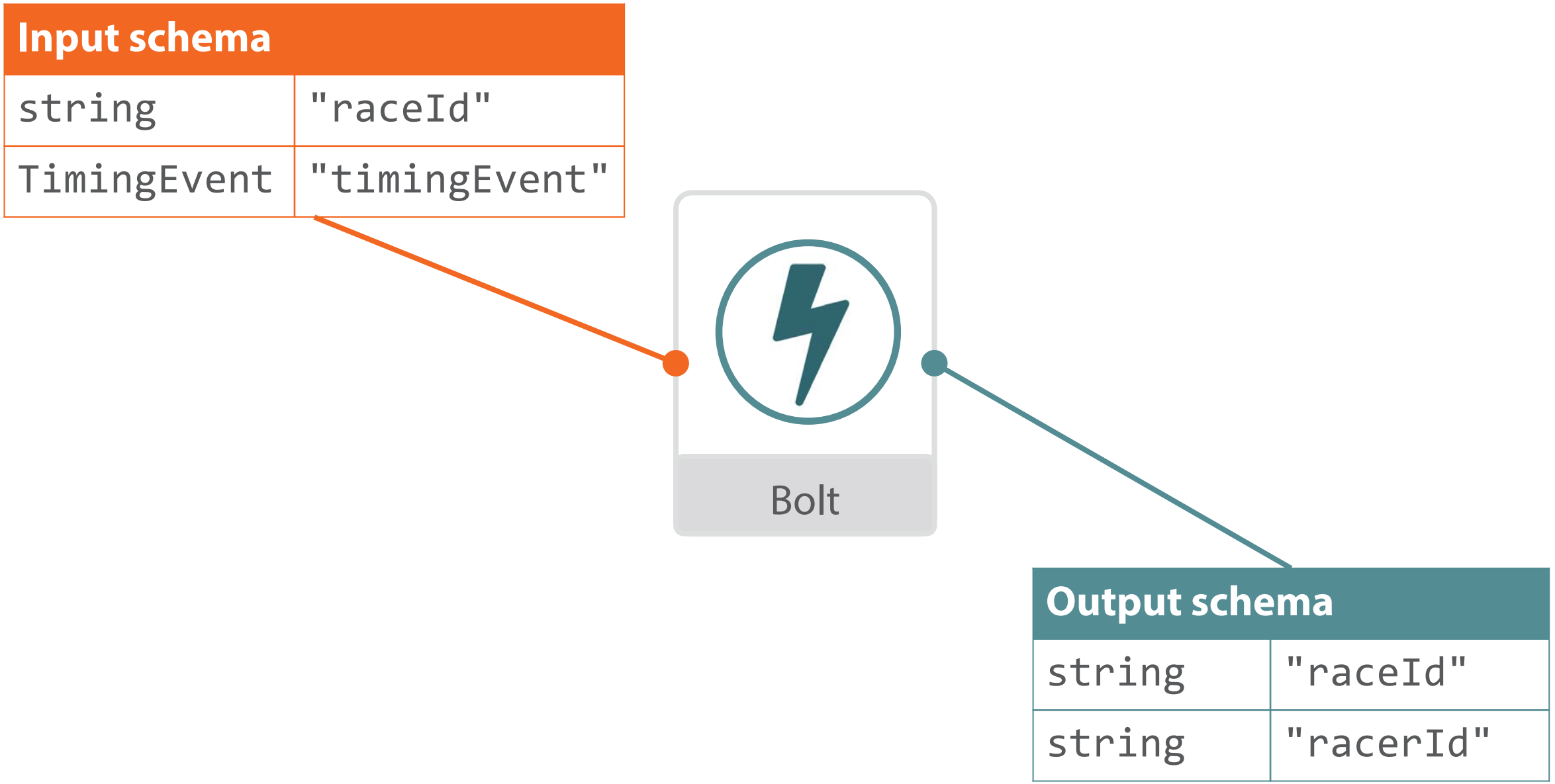


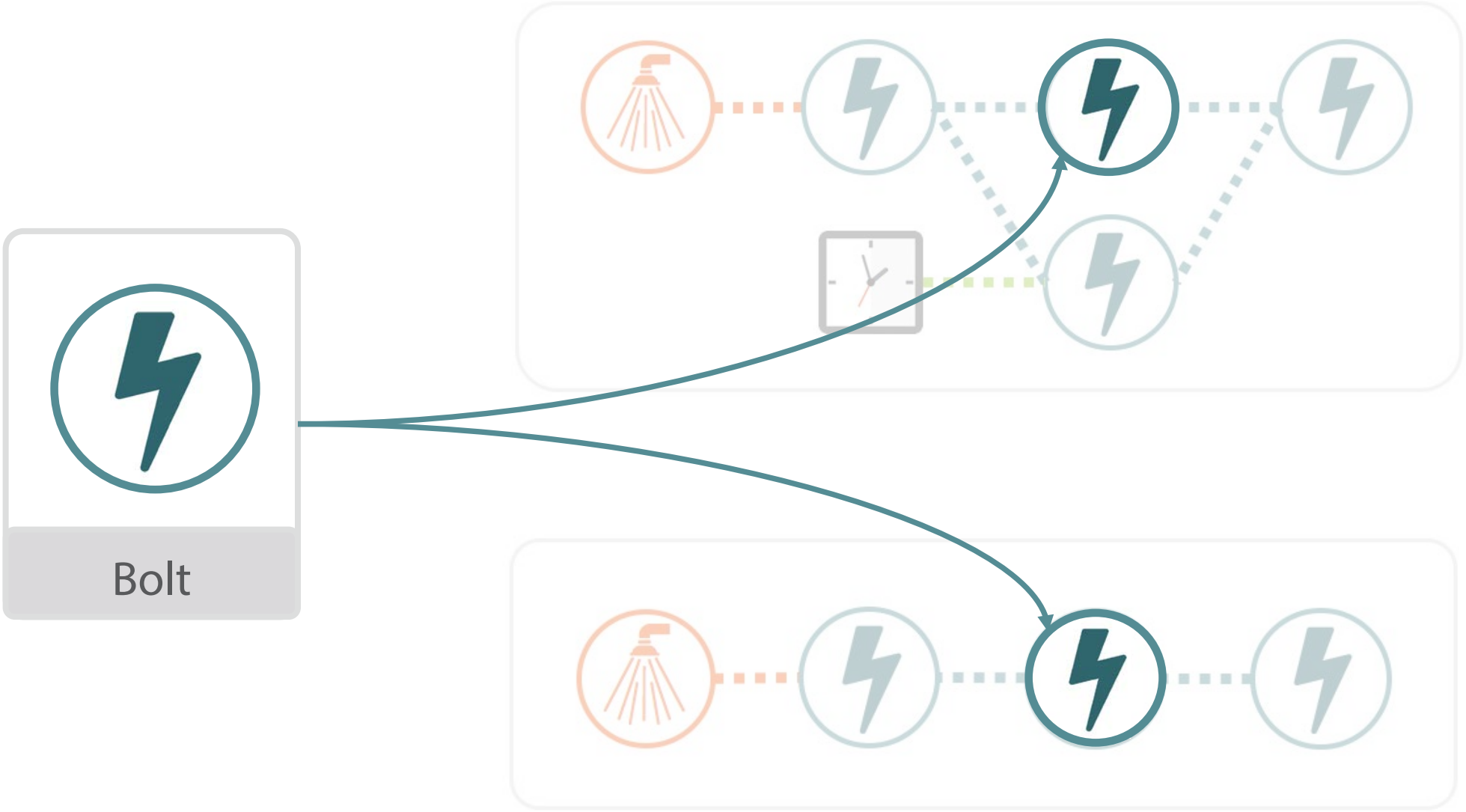


Small, simple classes

Easily testable

SOLID

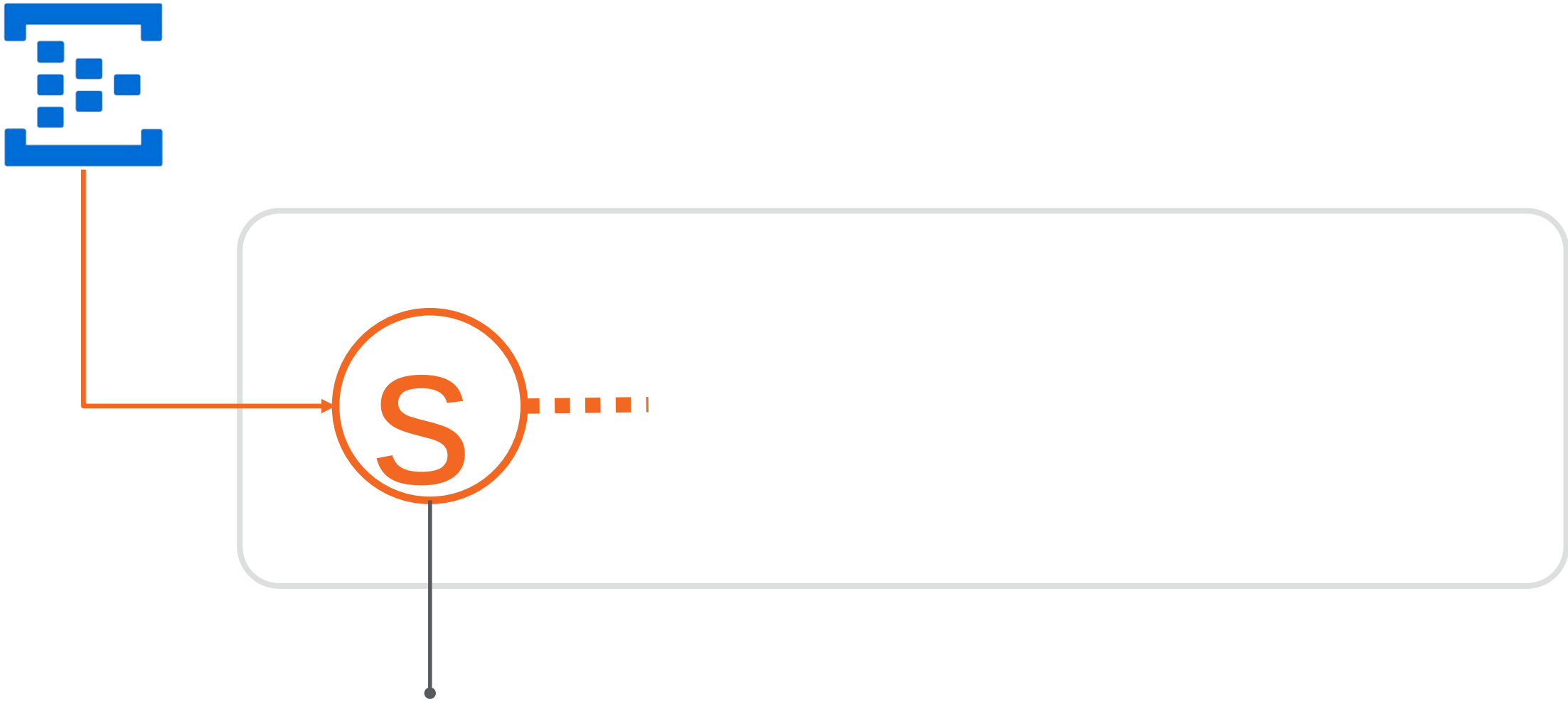

Input schema
string
"raceId"
TimingEvent
"timingEvent"
Bolt
Output schema
string
"raceId"
string
"racerId"



Bolt & spout instances **x80**

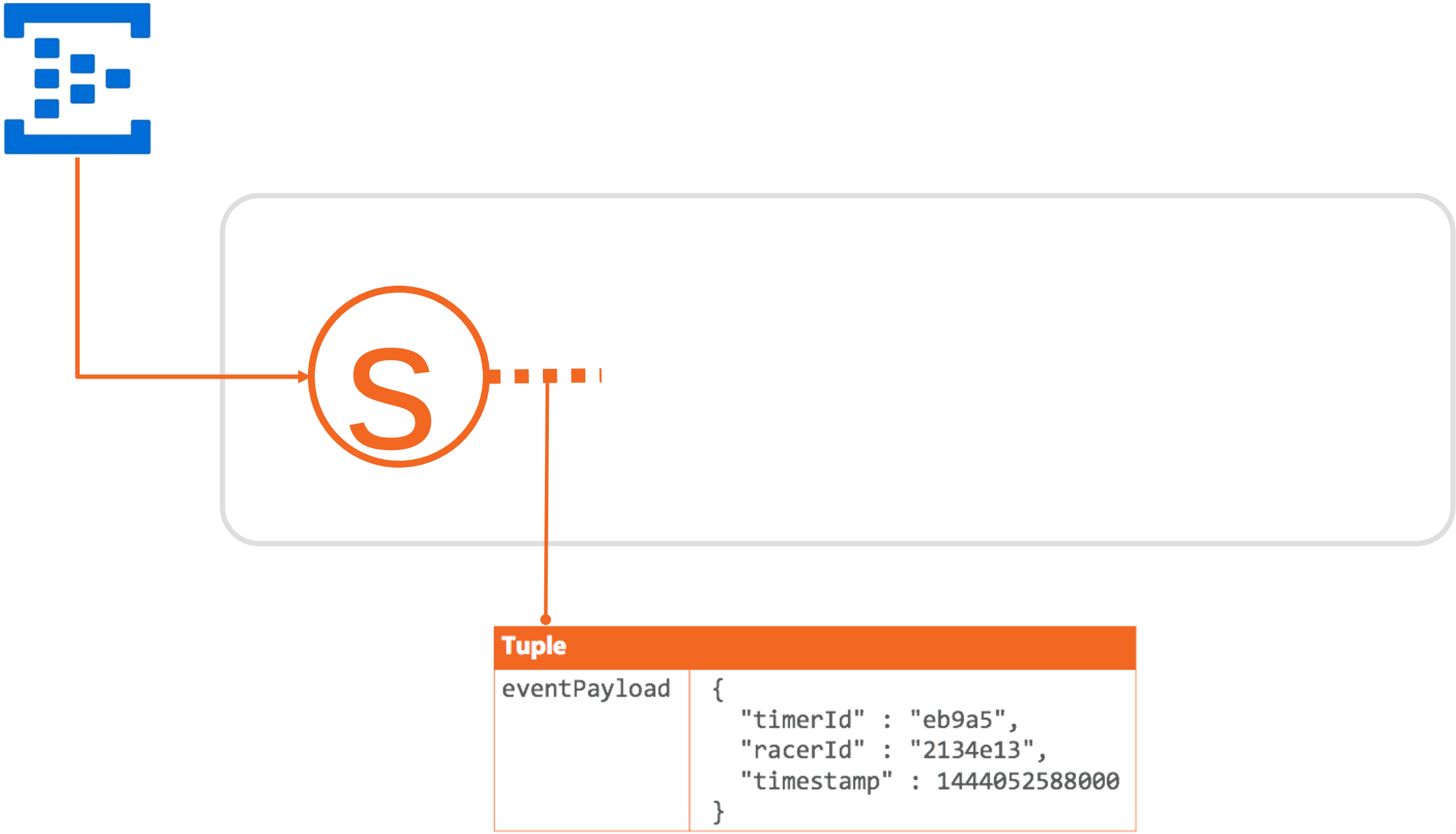
Bolt & spout instances **x144**

Timer ID	Racer ID	Timestamp
eb9a5	2134e13	1444052588000
eb9a5	a545abf	1444052591740



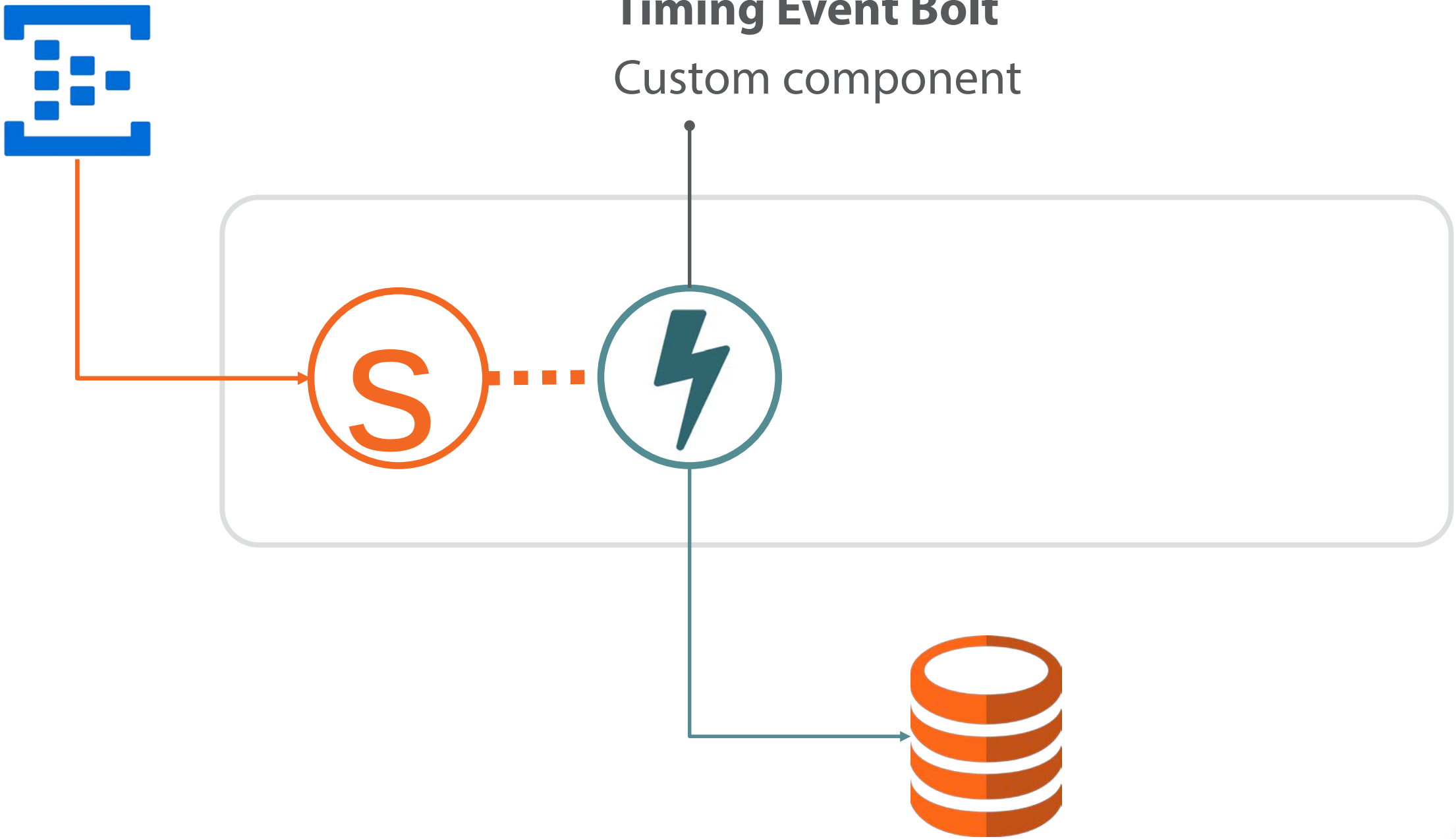
Java Event Hub Spout

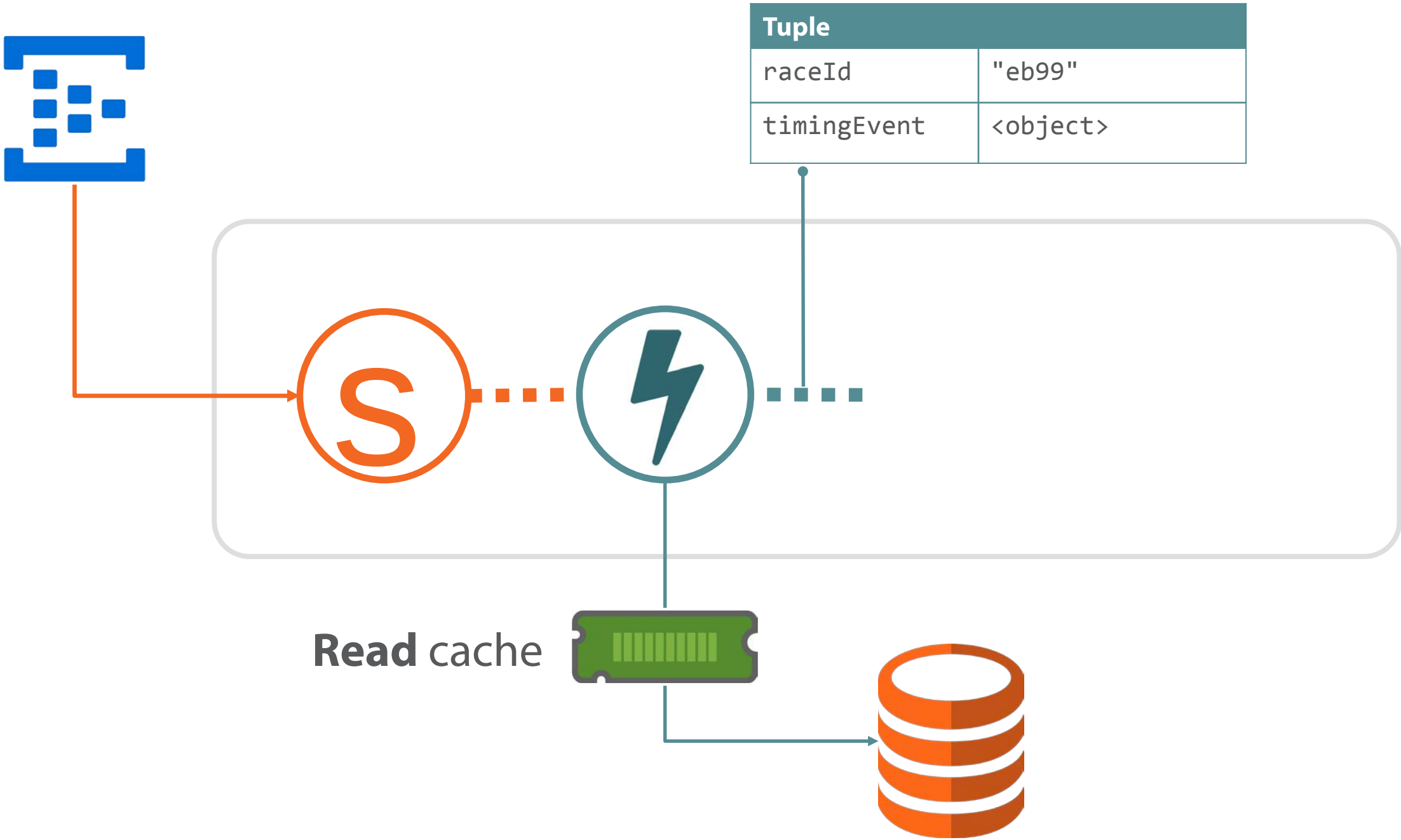
Microsoft package

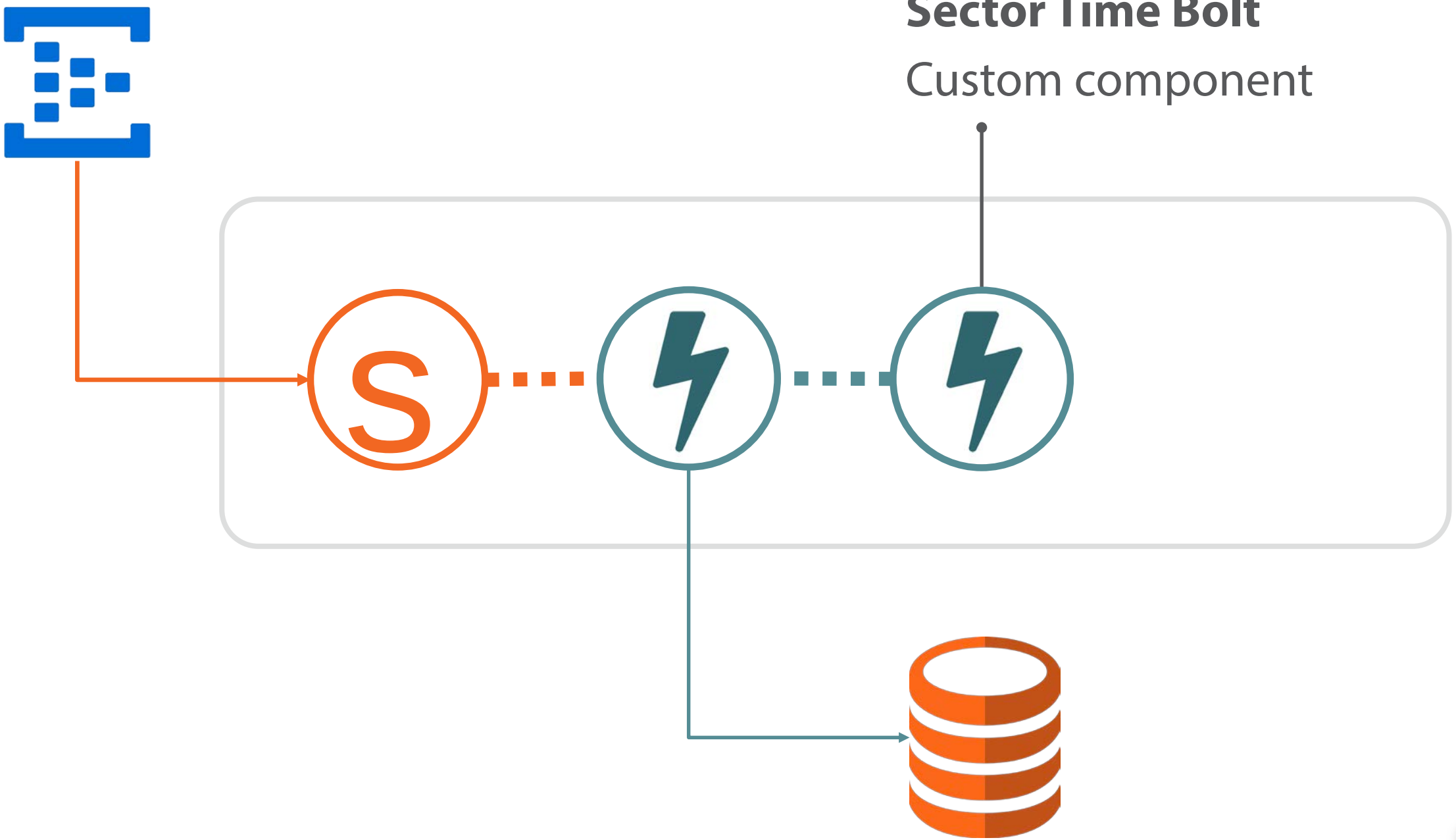


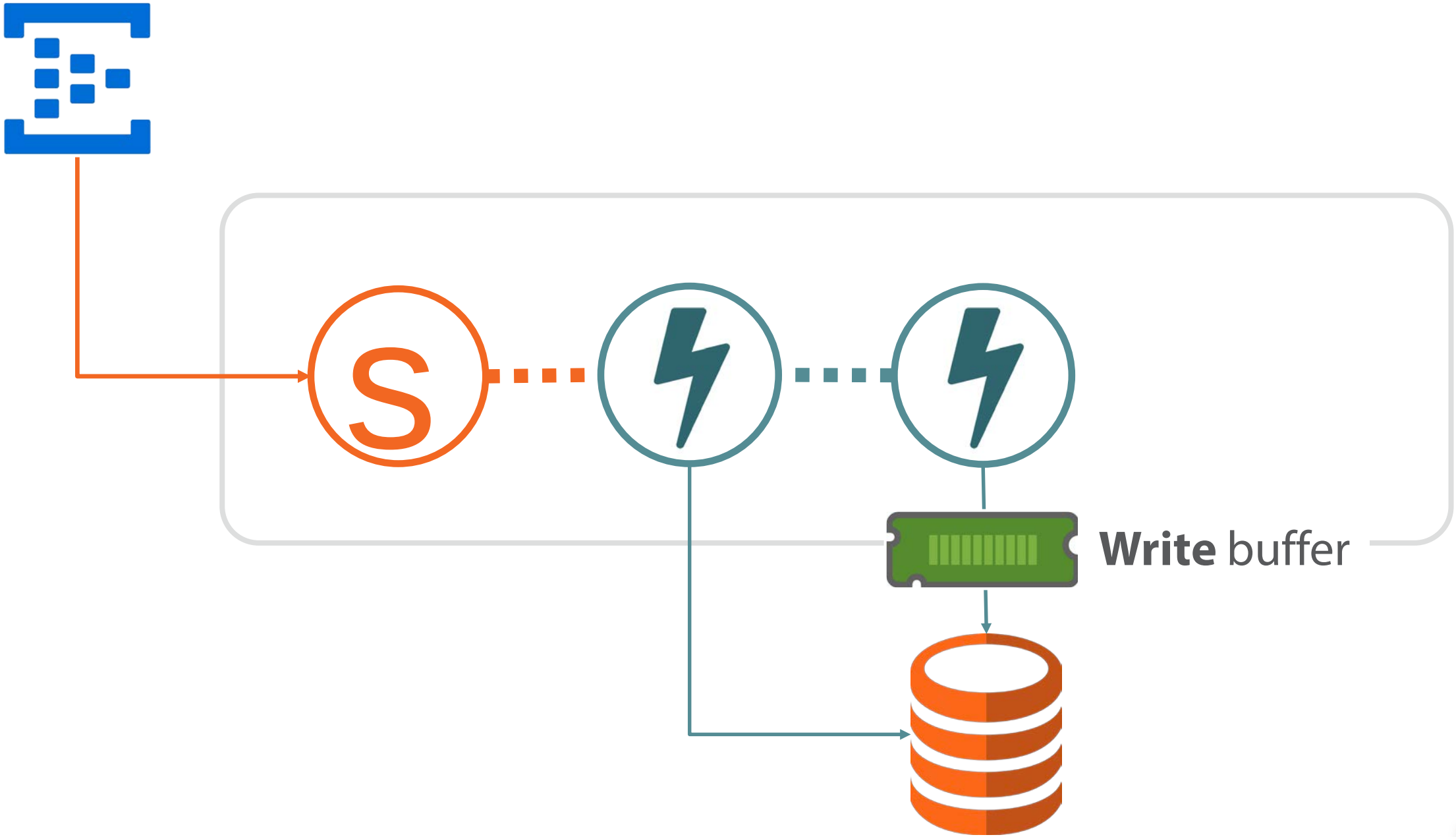
Timing Event Bolt

Custom component

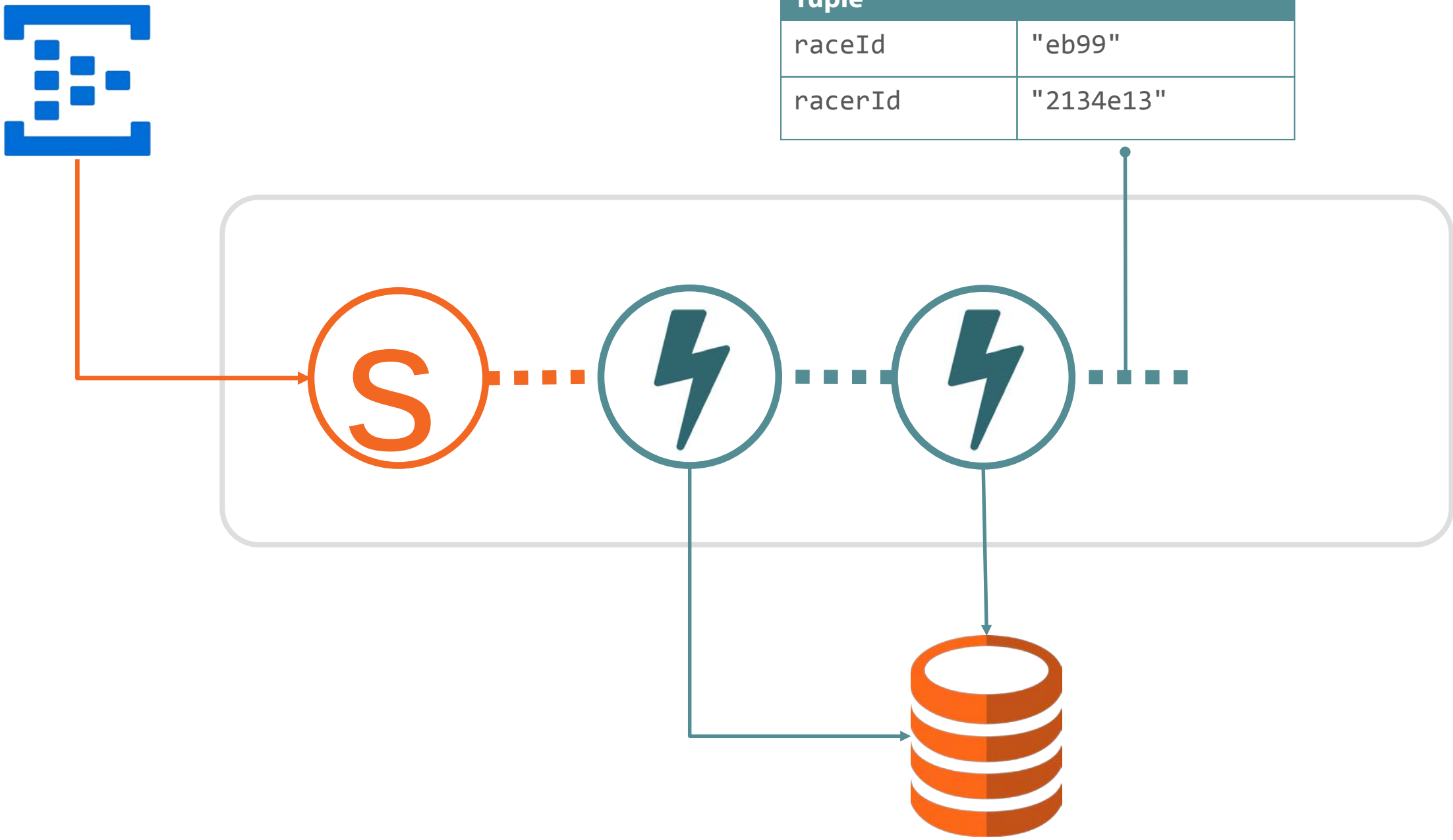


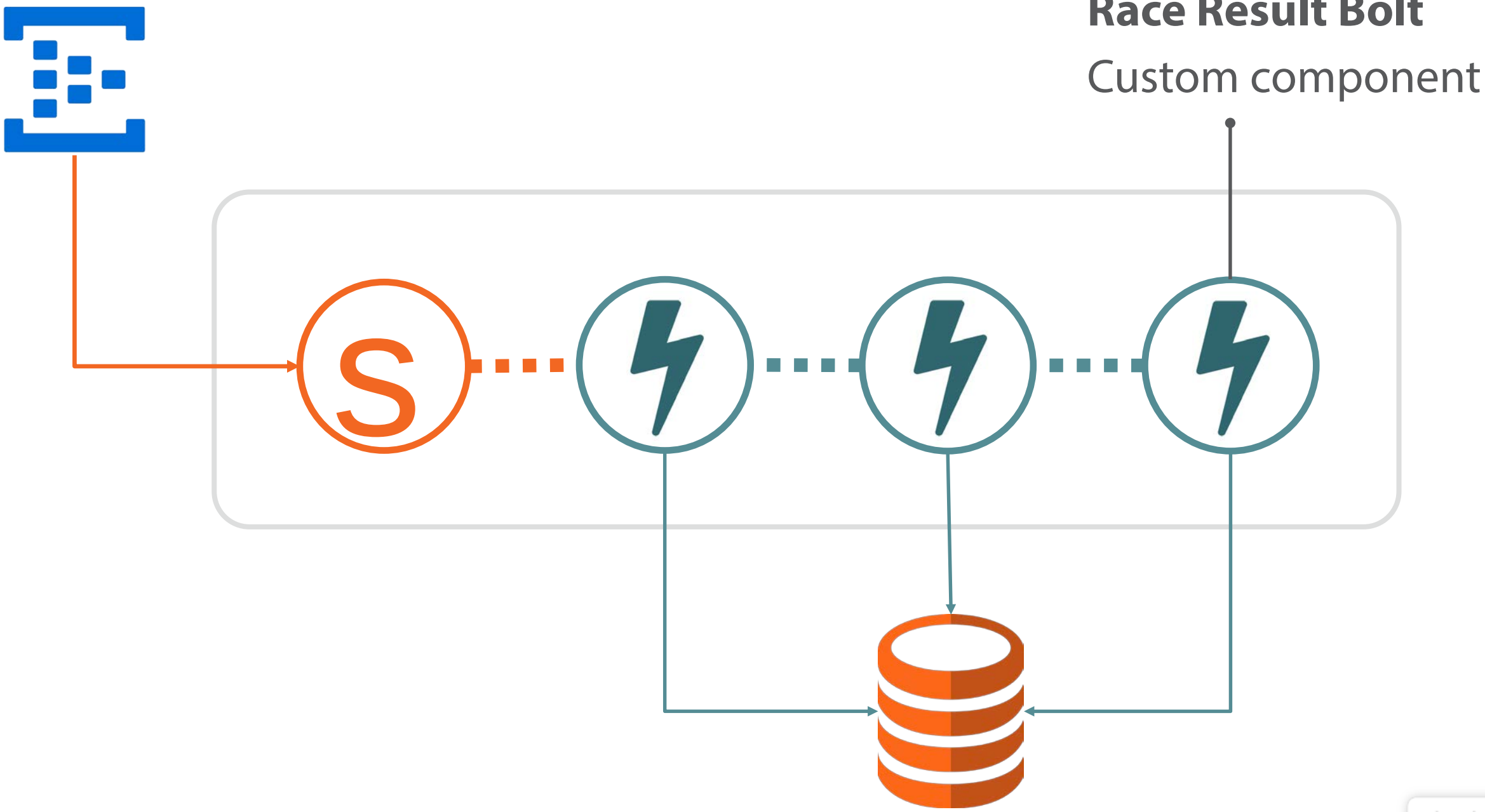



Sector Time Bolt
Custom component
S


S
Write buffer

Tuple	
raceId	"eb99"
racerId	"2134e13"




Race Result Bolt
Custom component
S

Demo: Timing Event Bolt

ISCPBolt Implementation

DAL Persist

Emit Tuple

Demo: Timing Event Bolt

In-Memory Cache

DAL Lookup

Input & Output Schemas

Demo: Sector Time Bolt

Input & Output Schemas

In-Memory Buffer

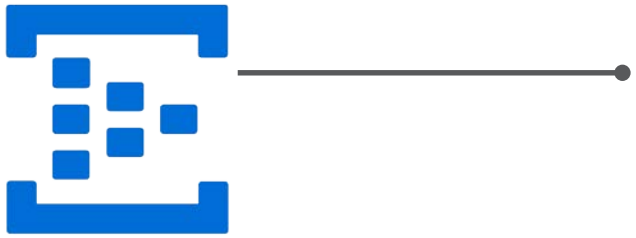
Tick Stream

Demo: Sector Time Bolt

DAL Buffer Flush

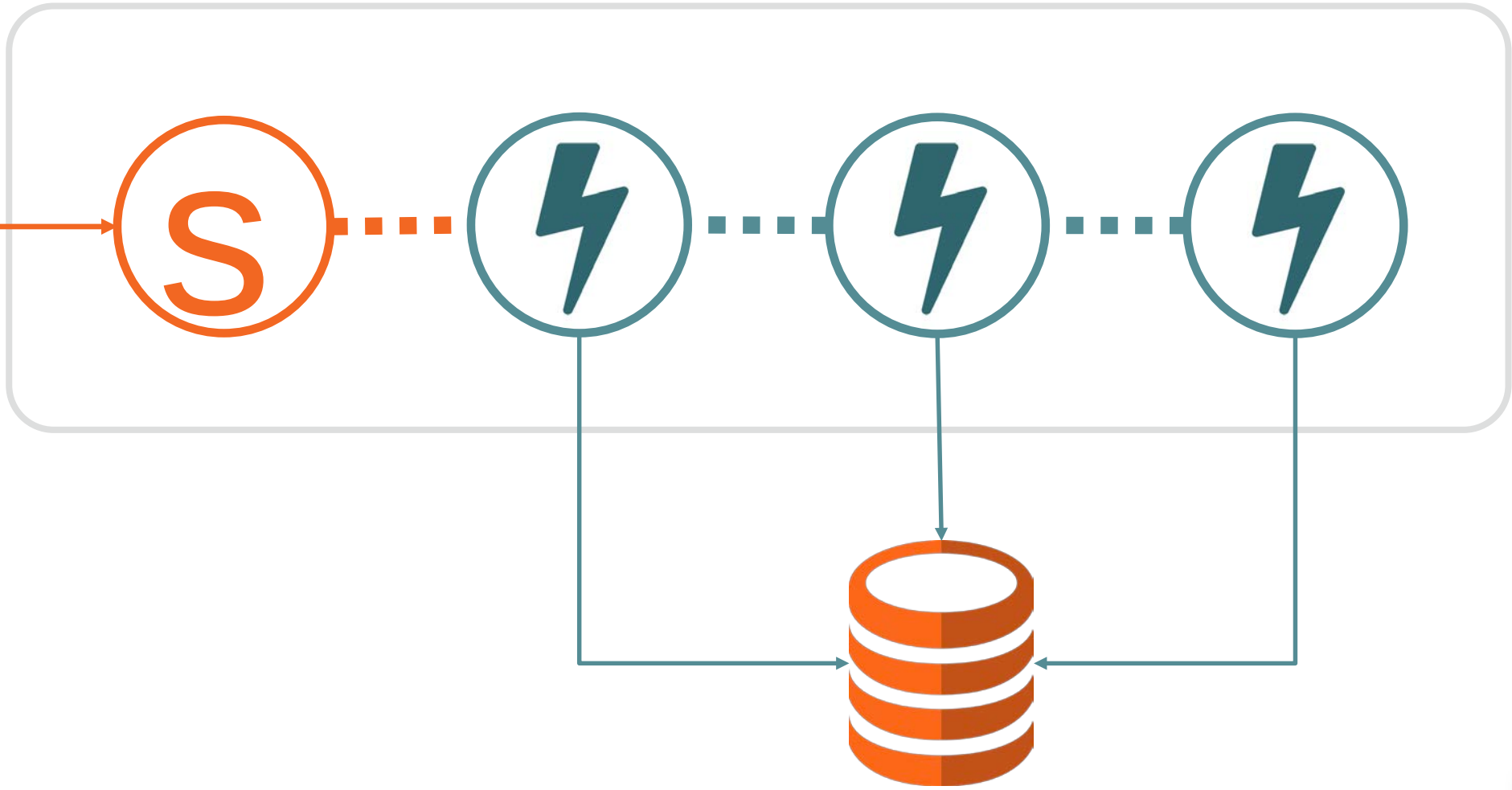
Emit Tuple

Race Result Bolt



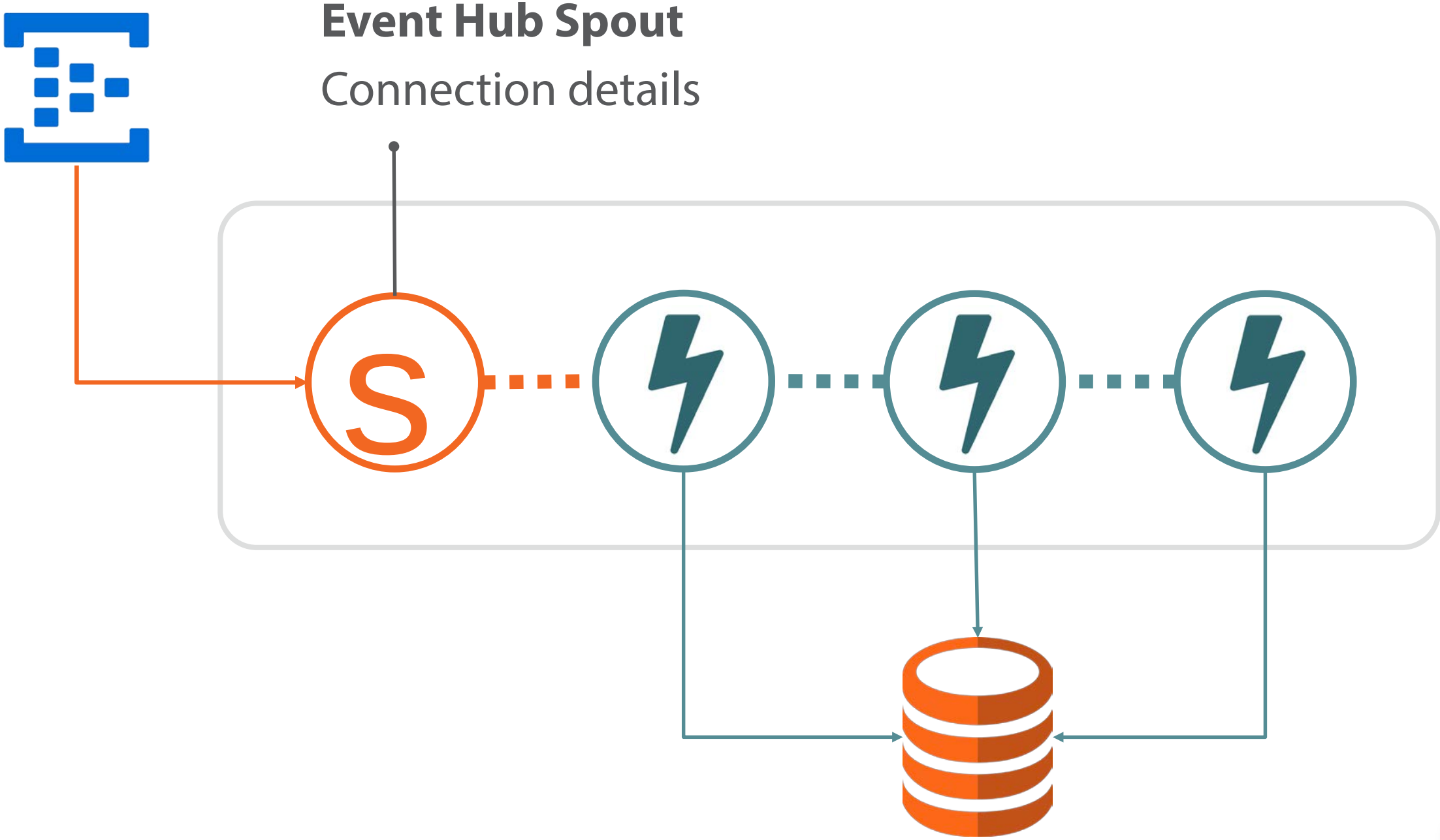
Azure Event Hub

Event data stream



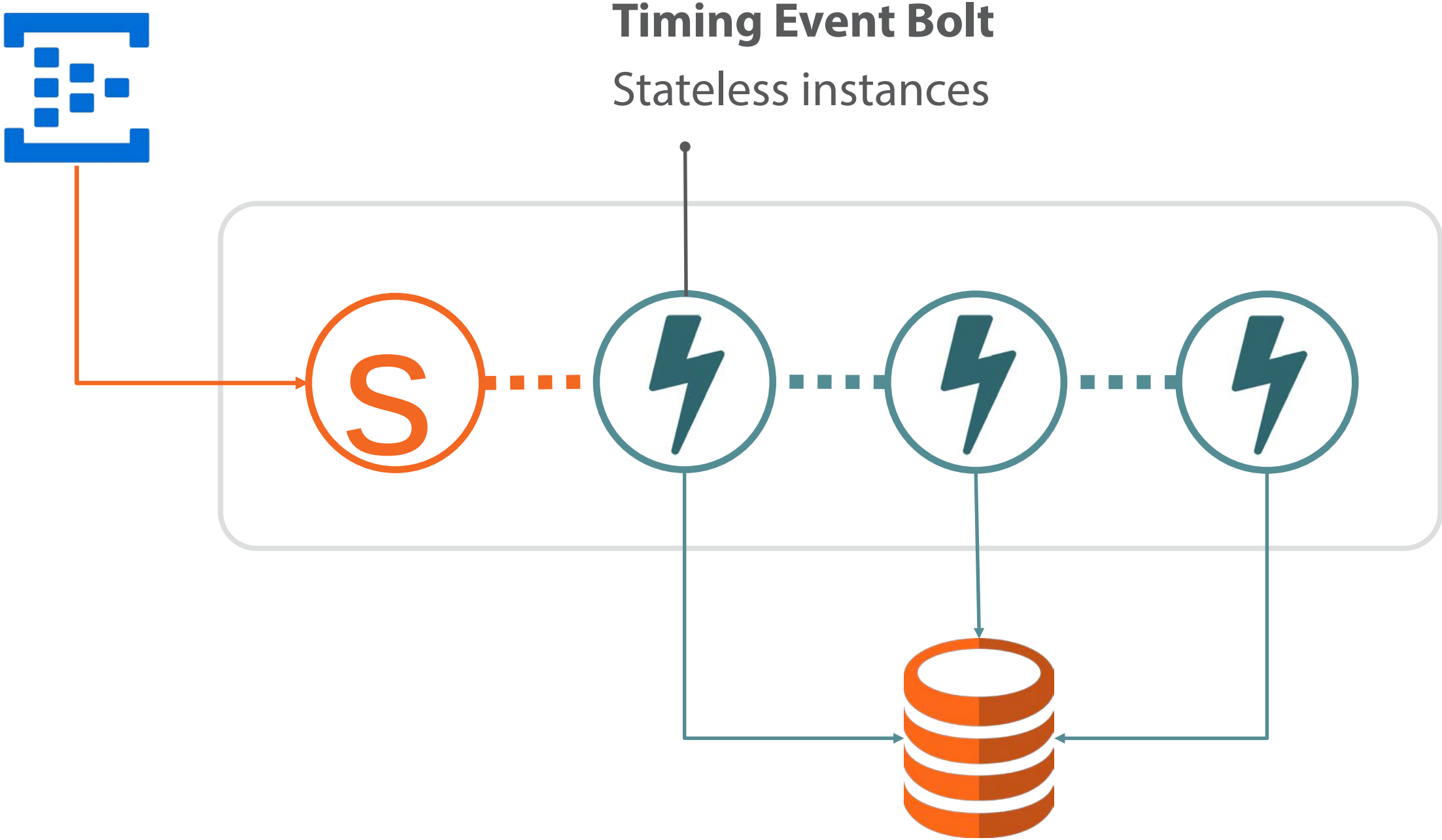
Event Hub Spout

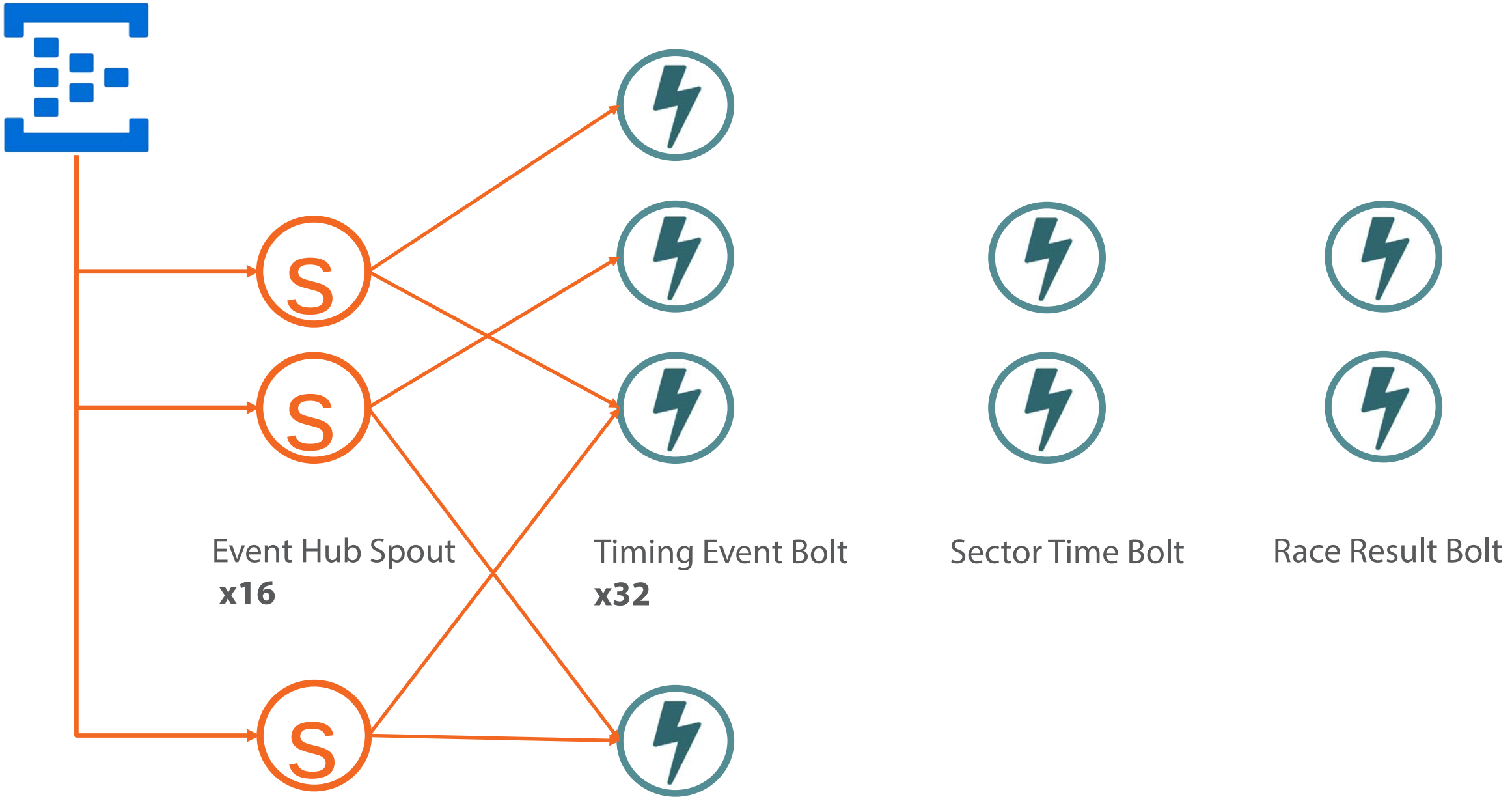
Connection details



Timing Event Bolt

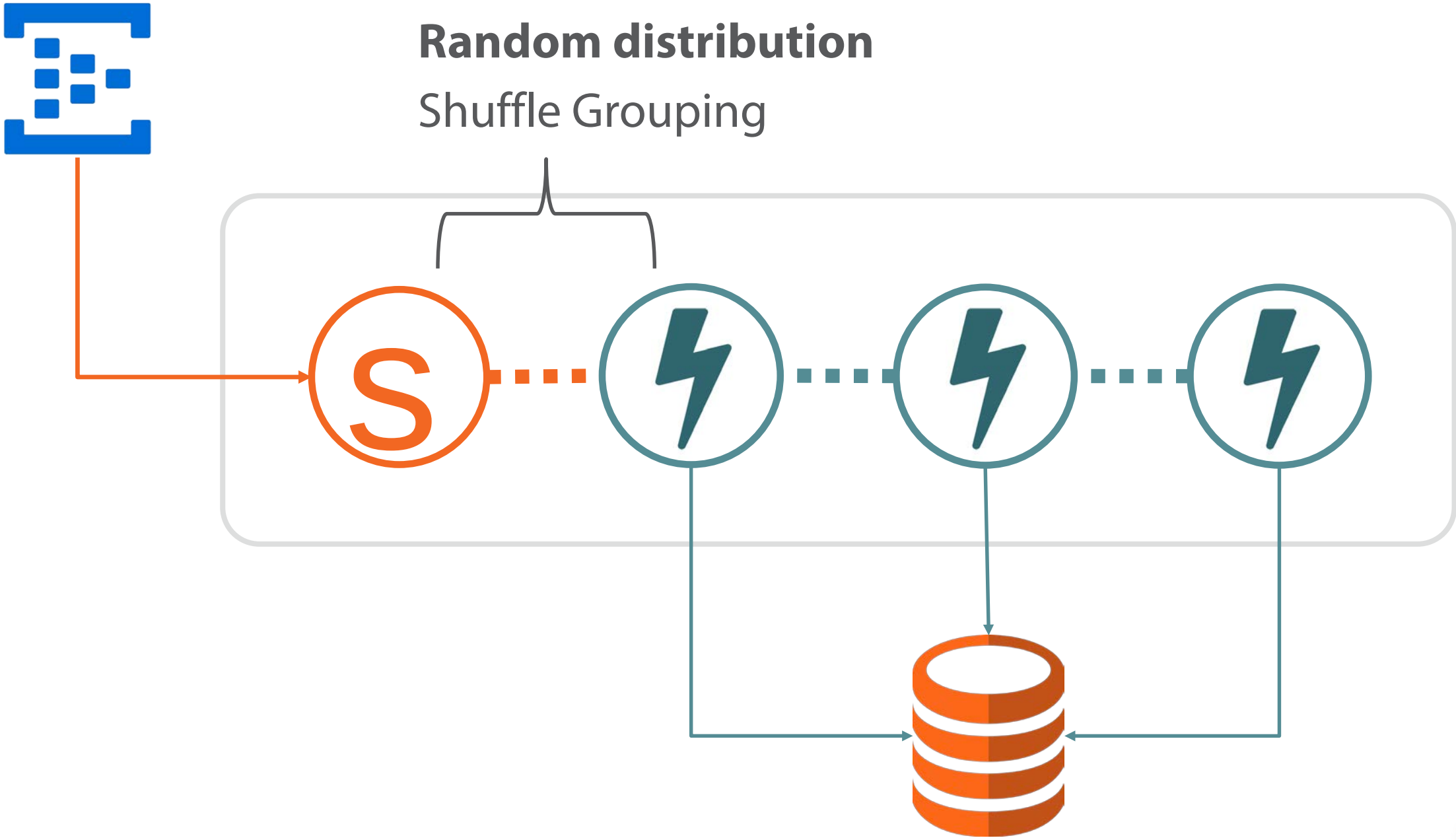
Stateless instances





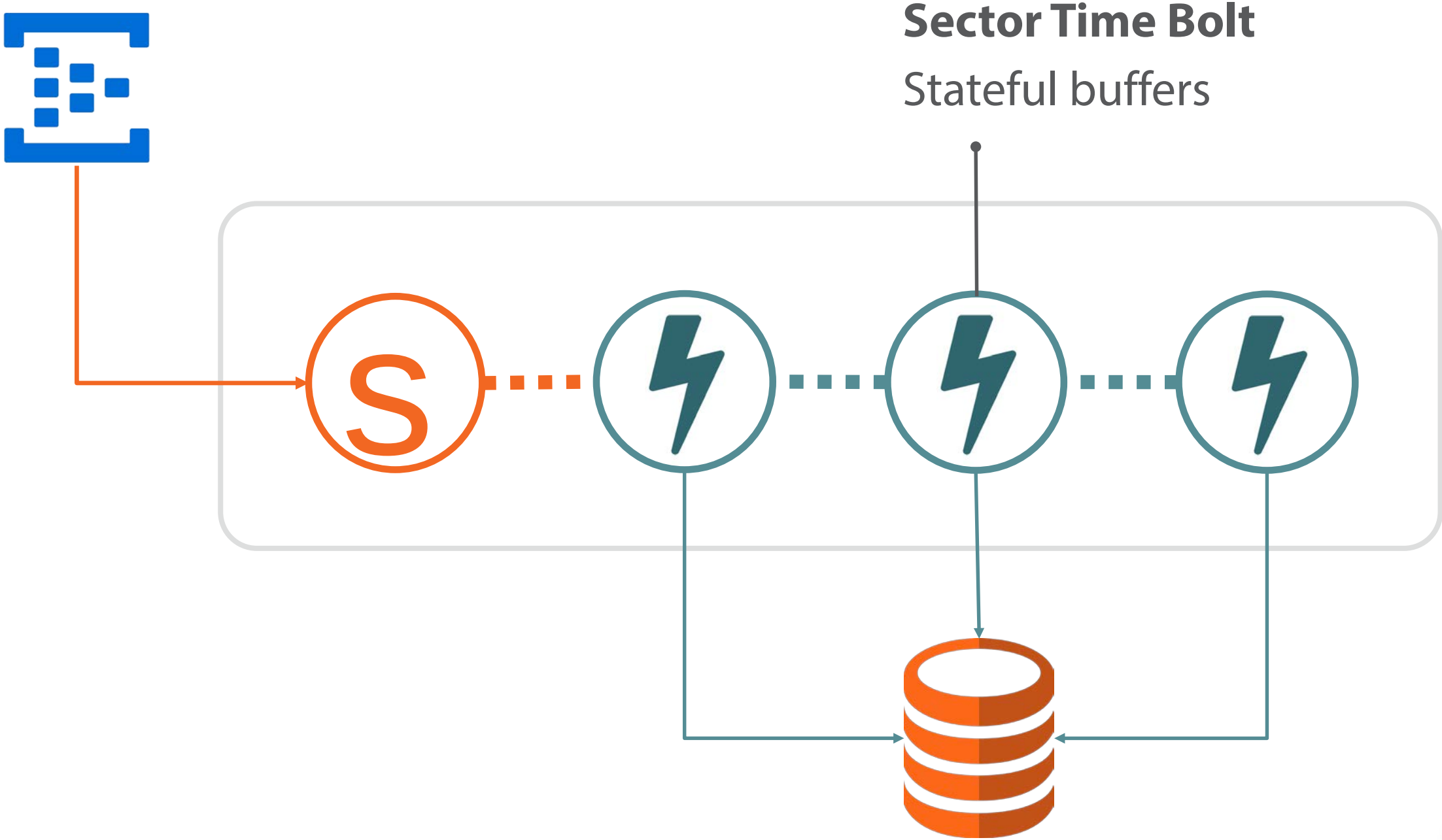
Random distribution

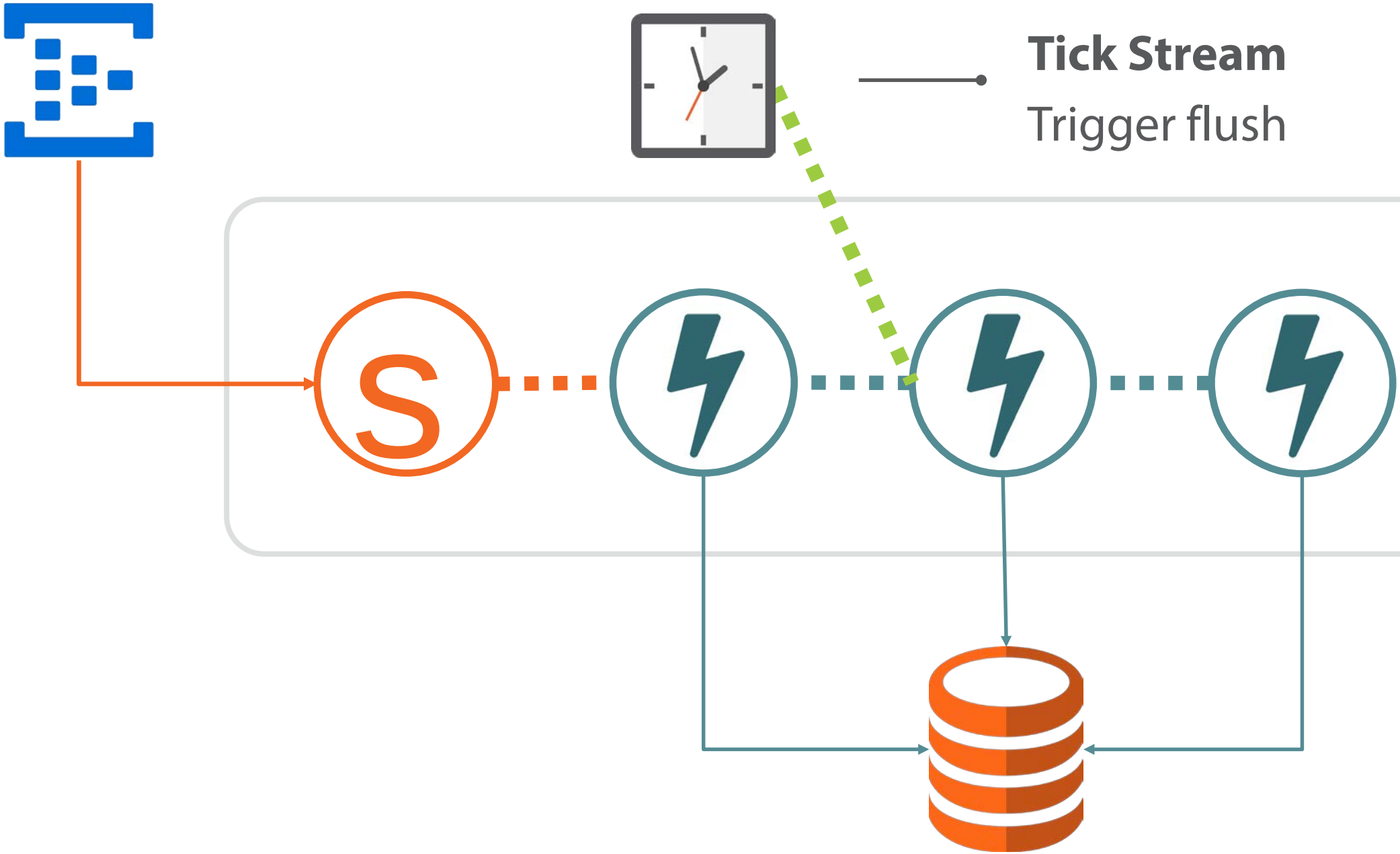
Shuffle Grouping

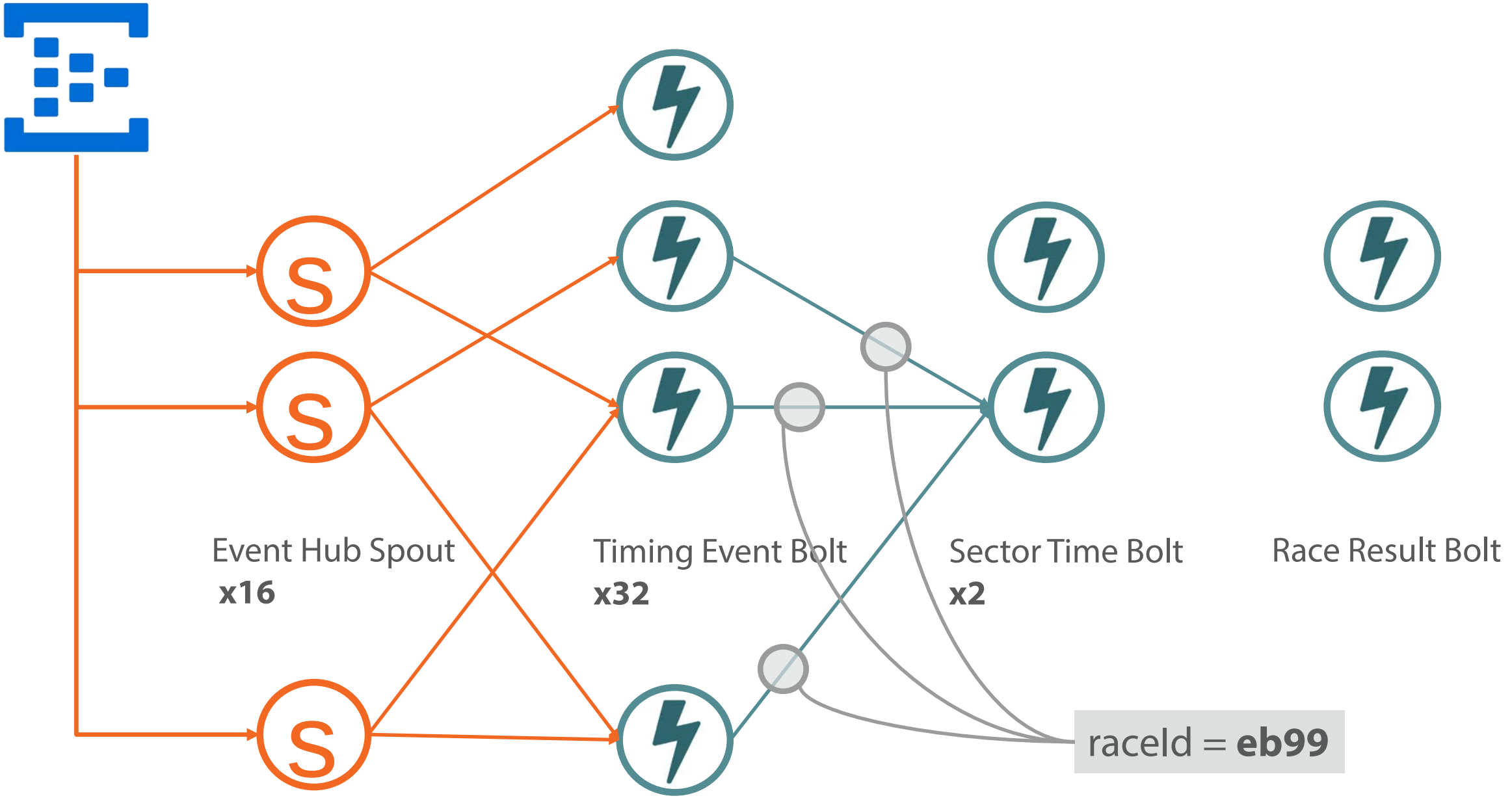


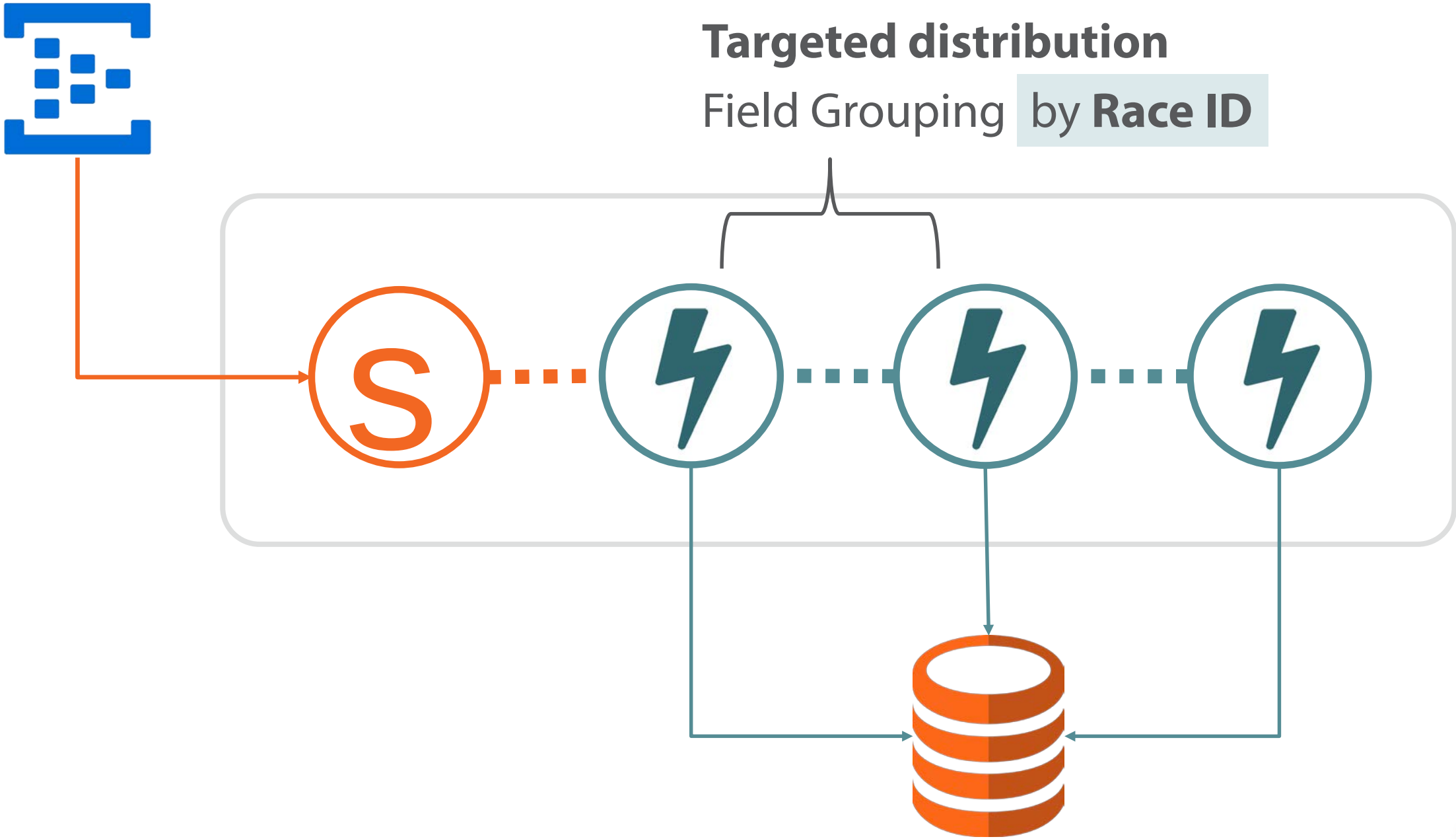
Sector Time Bolt

Stateful buffers



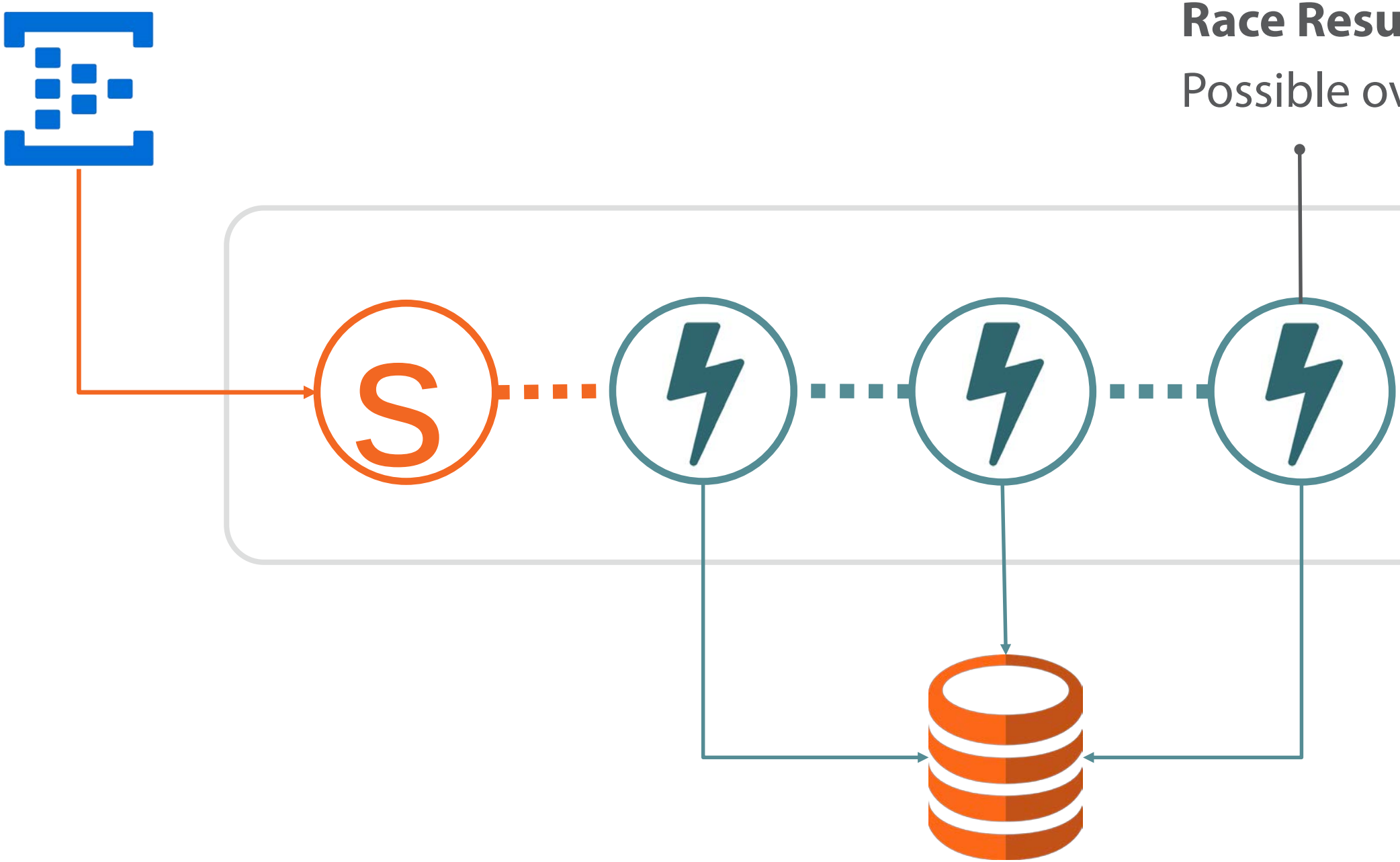

Tick Stream
Trigger flush
S

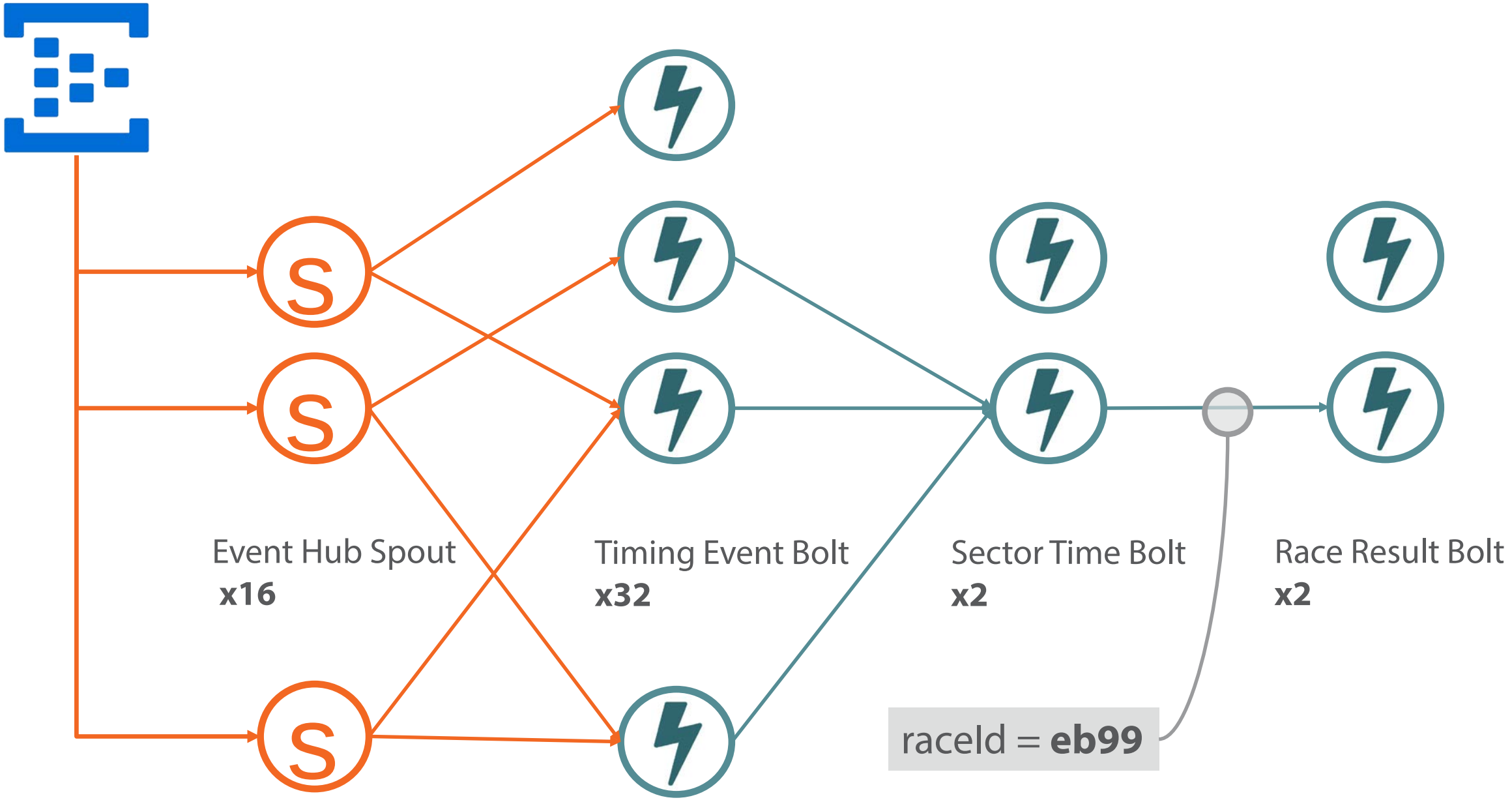




Race Result Bolt

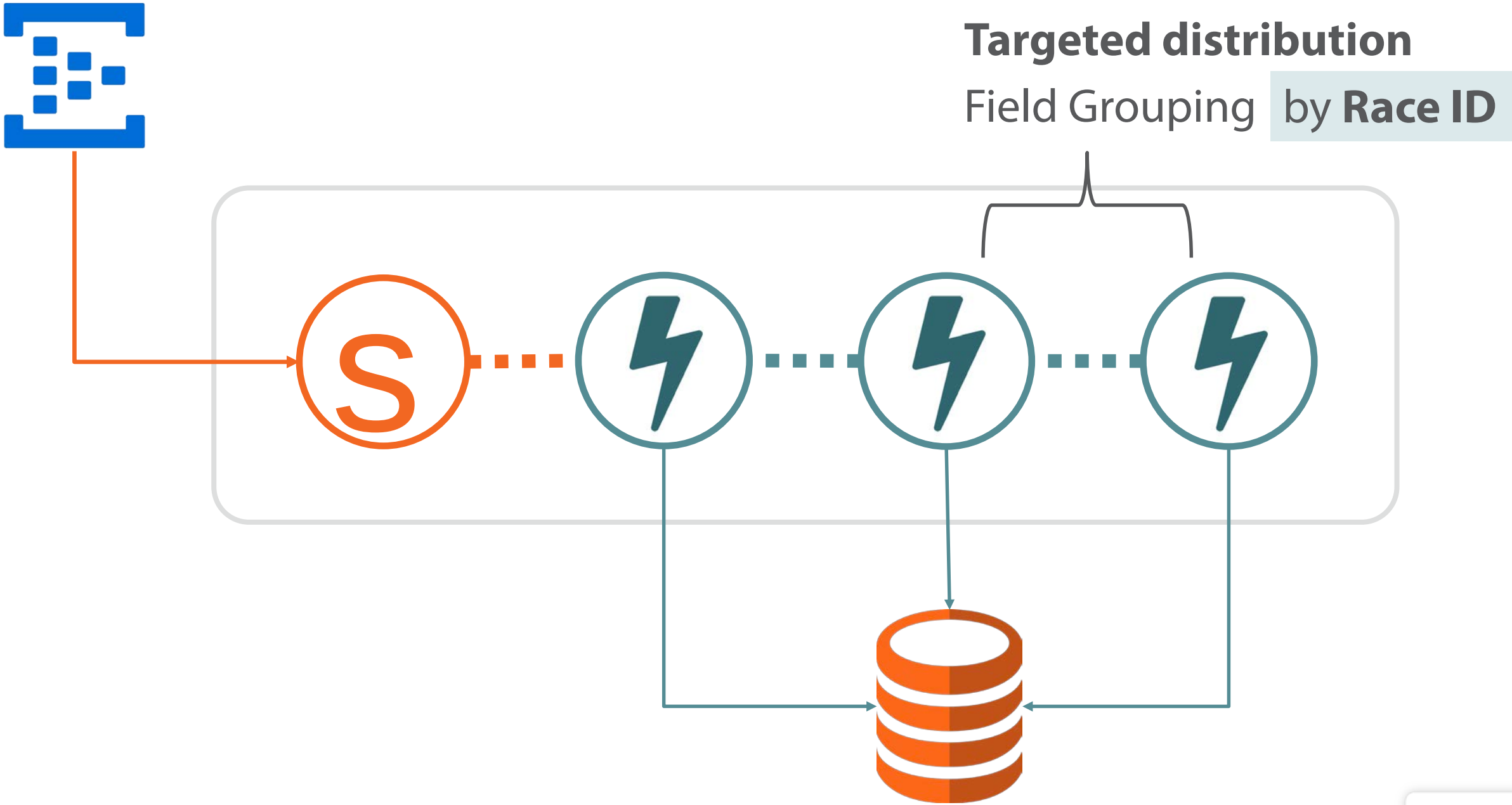
Possible overwrite

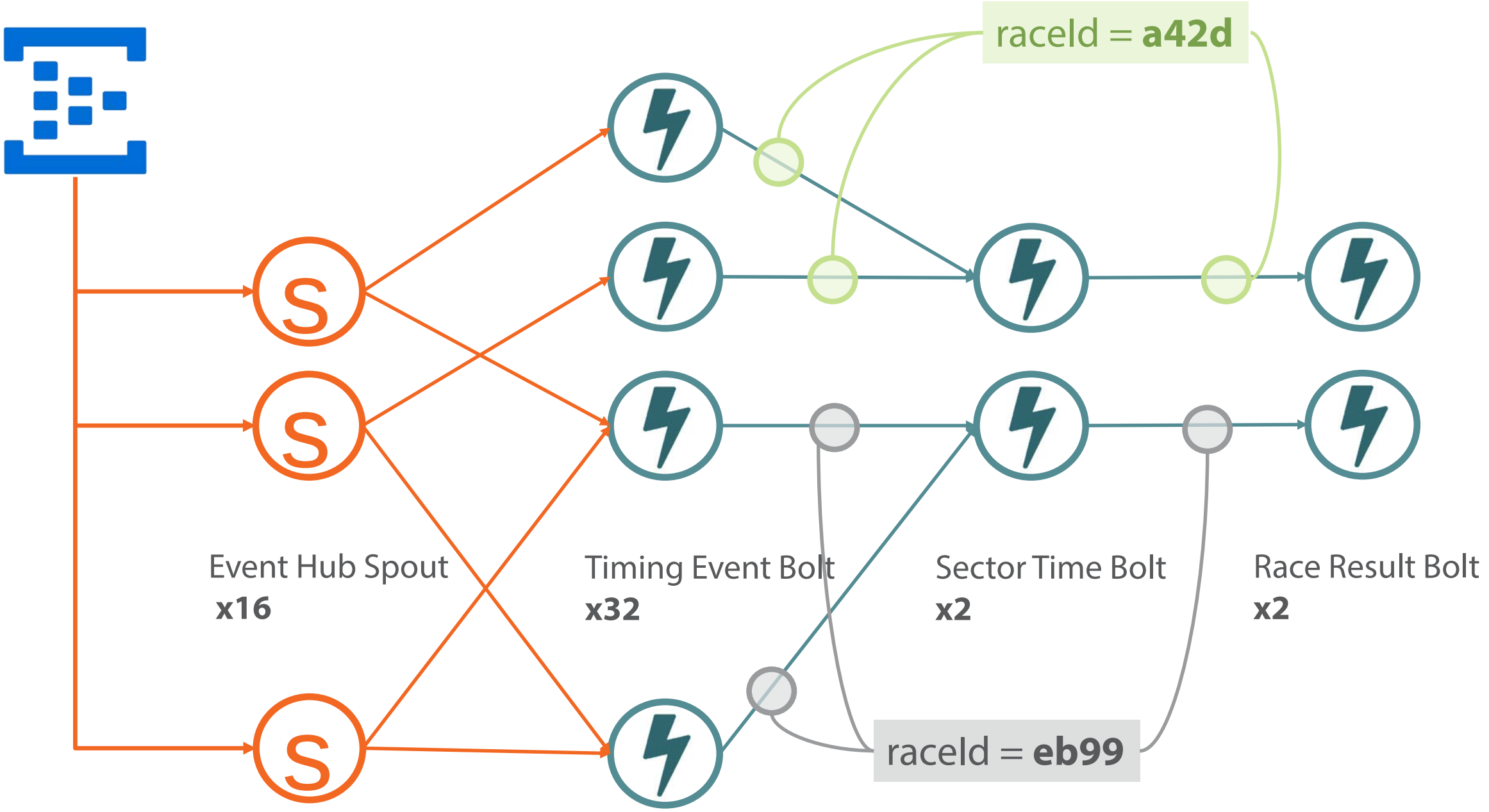




Targeted distribution

Field Grouping by **Race ID**





Race Topology Spec

```
{:name "RaceTiming-201510051039"
 :topology
  (topology
   {
     "EventHubSpout"
     (spout-spec
      (scp-load-component
            "com.microsoft.eventhubs.spout.EventHubSpout"
```

Race Topology Spec

```
(
 topology
   "EventHubSpout"   (spout-spec ...
   "TimingEventBolt" (bolt-spec ...
   "SectorTimeBolt"  (bolt-spec ...
   "RaceResultBolt"  (bolt-spec ...
)
:config { "topology.max.spout.pending" 1000 }
```

Race Topology Spec

```
"TimingEventBolt"
  (bolt-spec
    {
      ["EventHubSpout" "default"] :shuffle
    }
```

Race Topology Spec

```
(scp-bolt
 {
  "plugin.name" "SCPHost.exe"
  "plugin.args"
    [
      "Sixeyed.RaceTiming.EventProcessor.dll"
      "Sixeyed.RaceTiming.EventProcessor.ComponentFactory"
      "GetTimingEventBolt"
```

Demo: Race Timing Topology

Topology Builder

Bolt Specs

Dependency Injection

Demo: Storm on HDInsight

Create Cluster

Package Topology

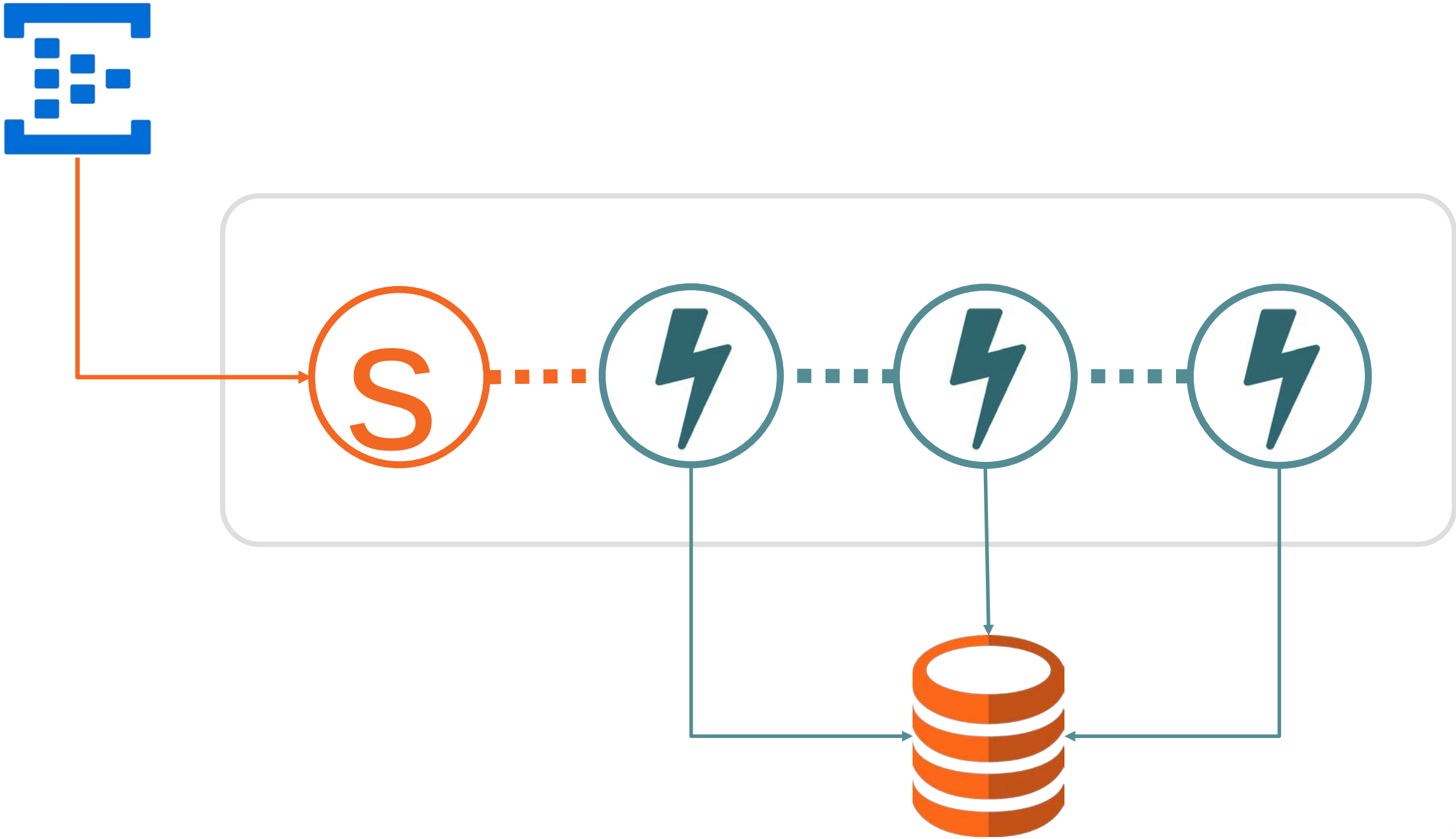
Deploy Topology

Demo: Race Simulator

Generate Event Hub Load

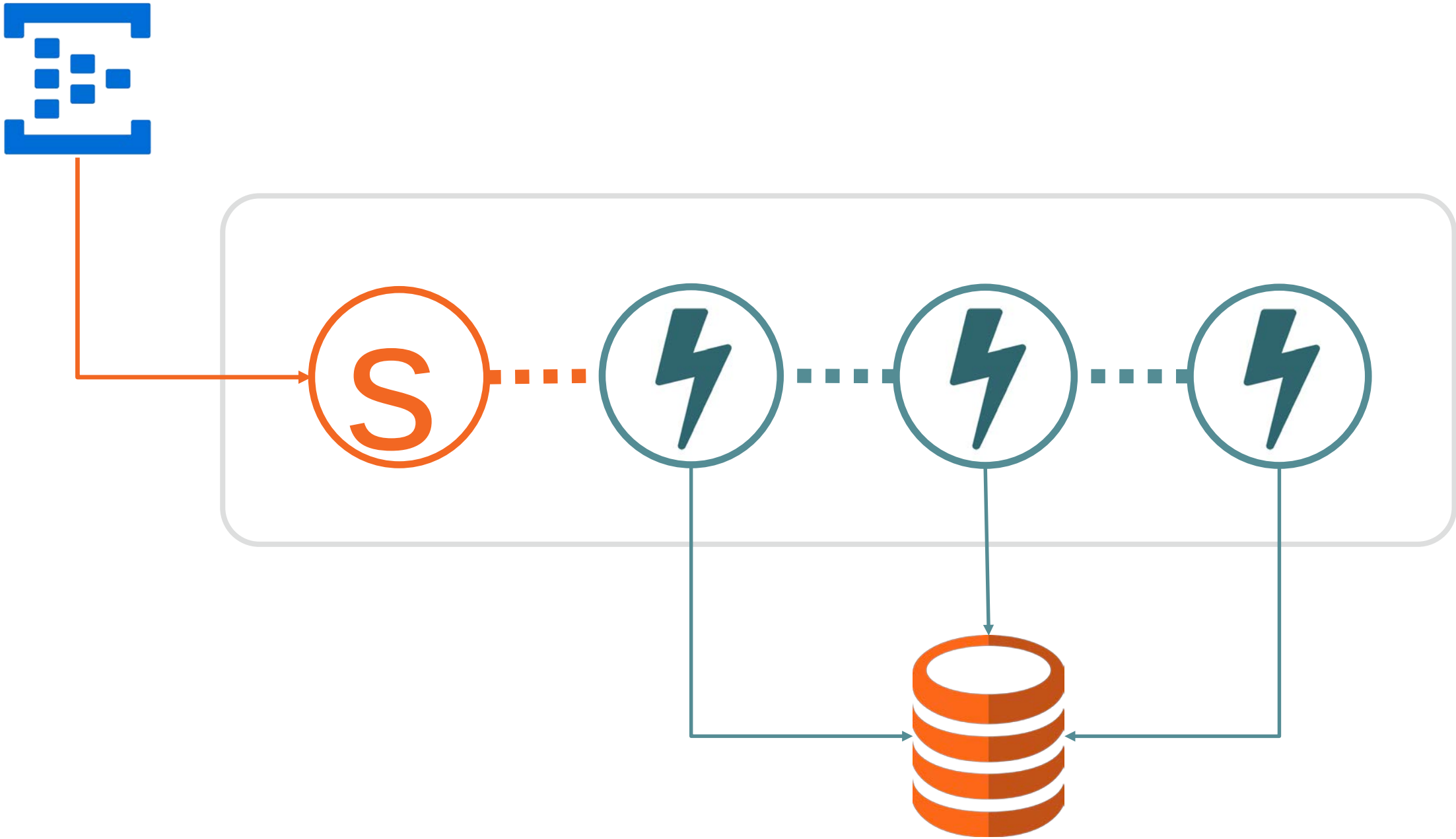
Monitor Topology

Verify Data in HBase

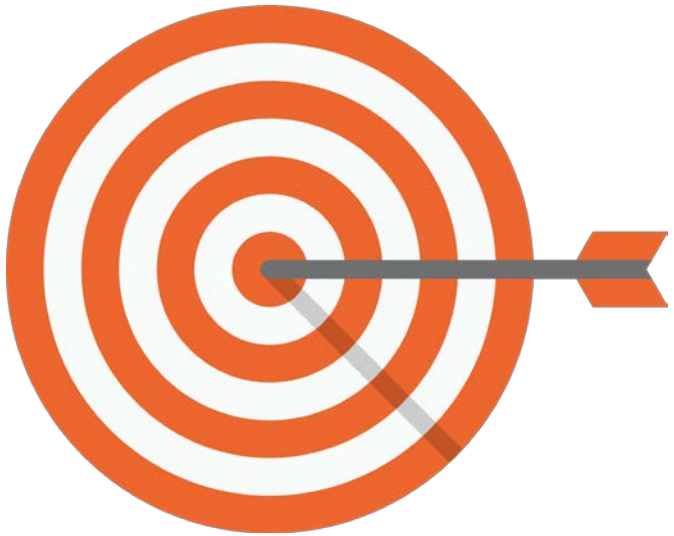


```
c:\RaceSimulator.exe
```

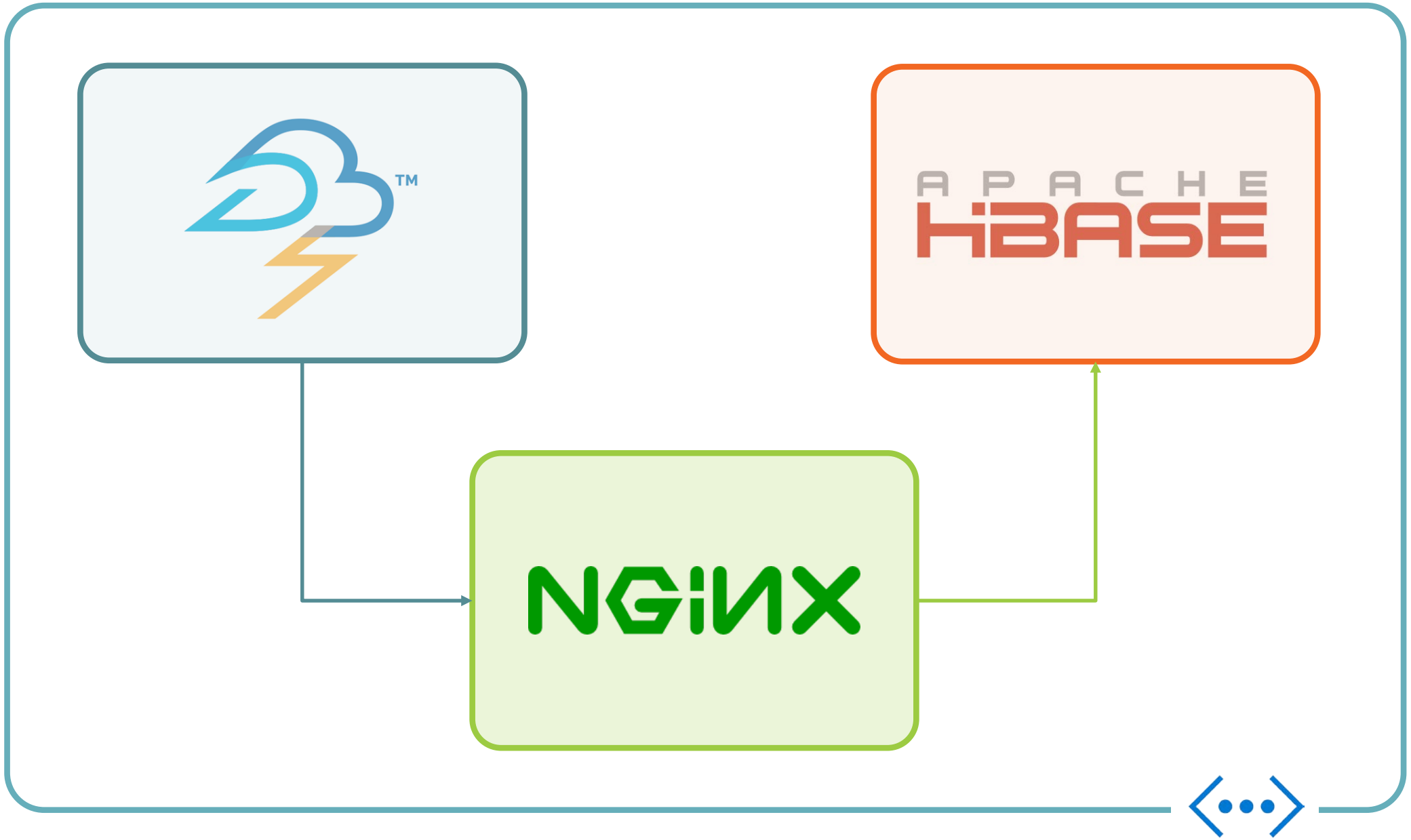
Timer ID	Racer ID	Timestamp
eb9a5	2134e13	1444052588000
eb9a5	a545abf	1444052591740

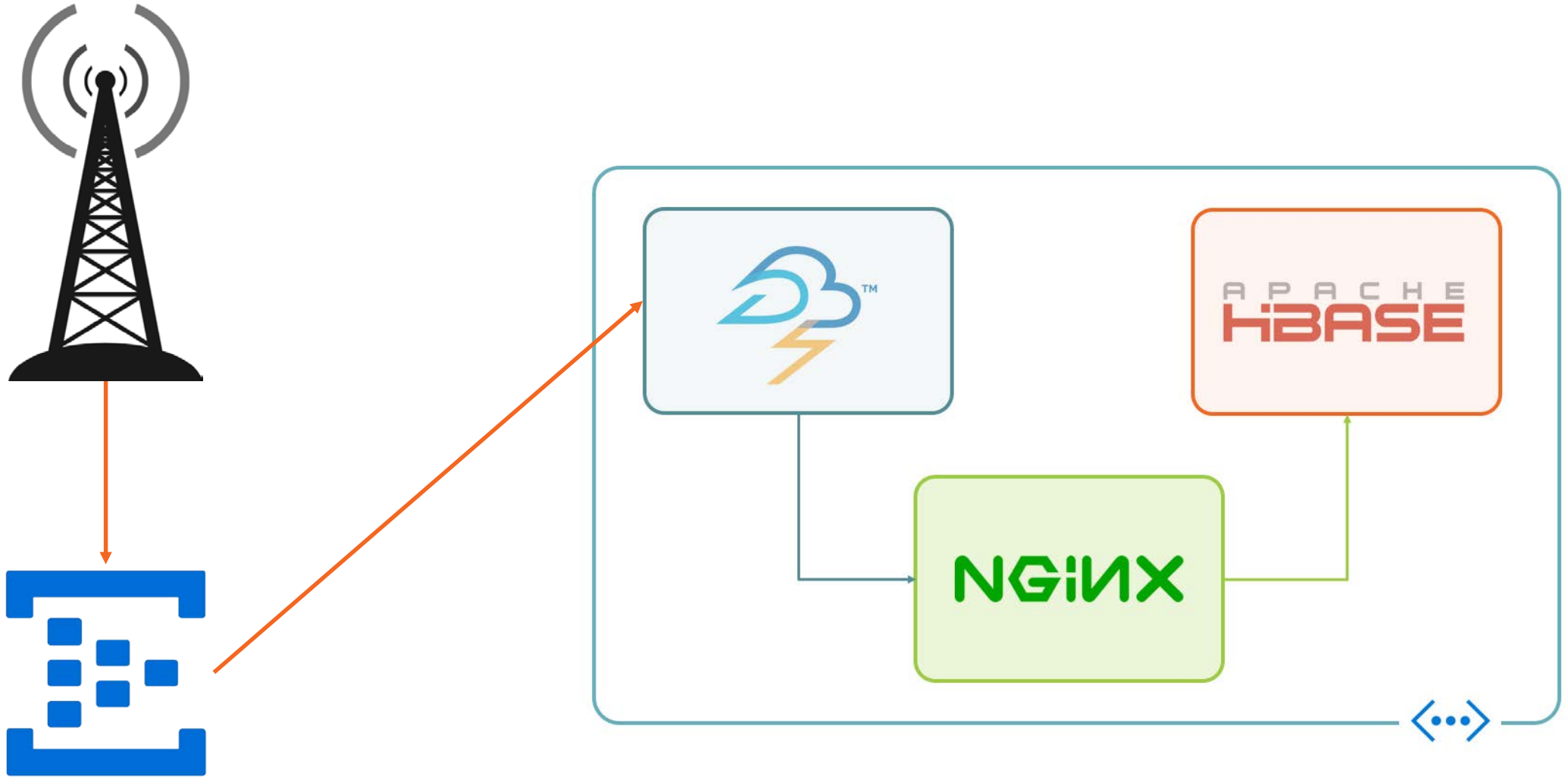


Module Goals

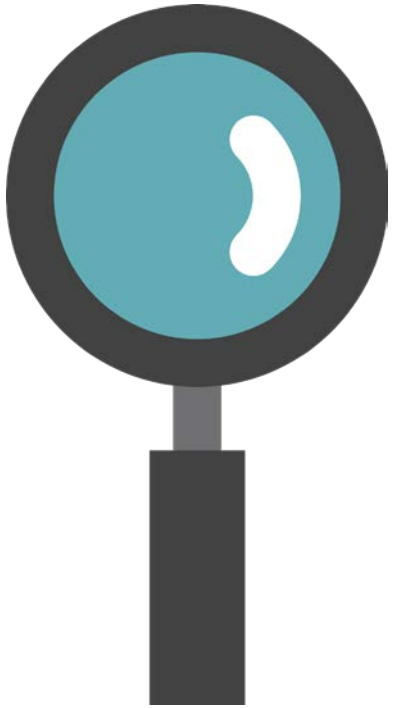


- Build Race Timing Components
- Configure Topology
- Deploy to HDInsight
- Verify End-to-End





Coming Next



- Storm Performance Tuning
- HDInsight Storm Clusters
- Unit Testing Storm
- Handling Failure