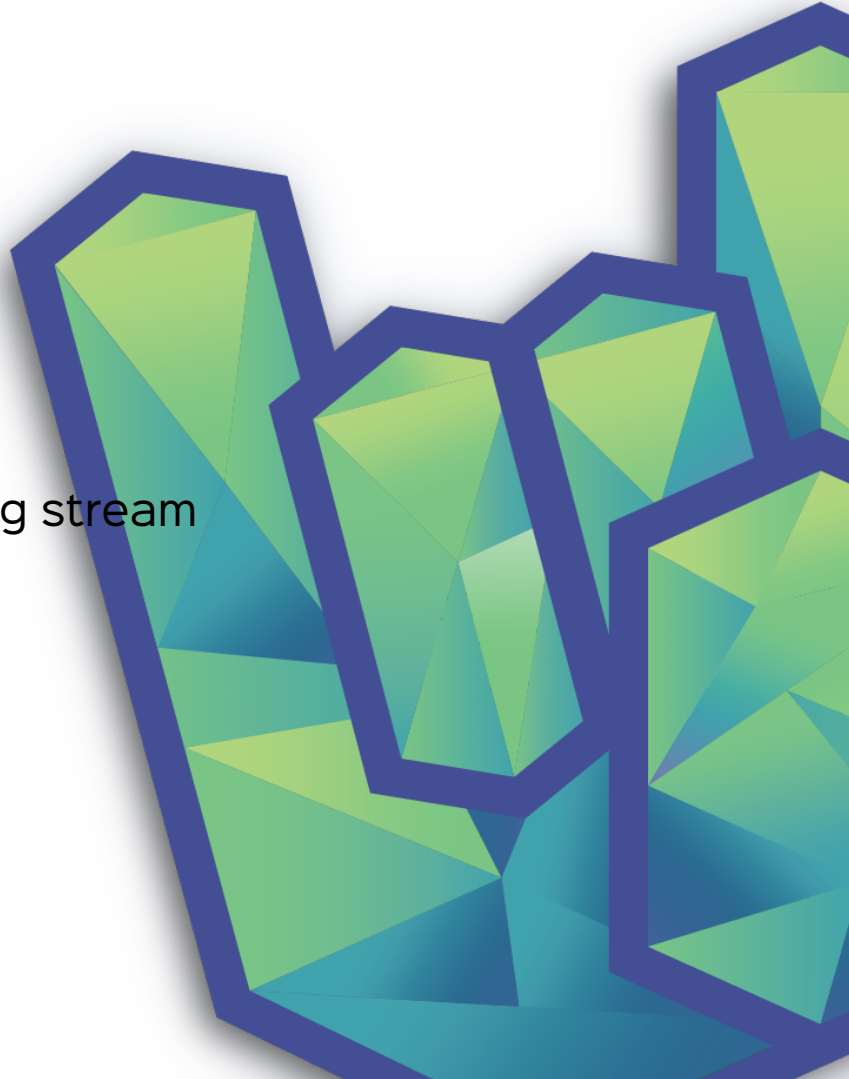


Materializing Streams



Goal

getting a meaningful value out of a running stream



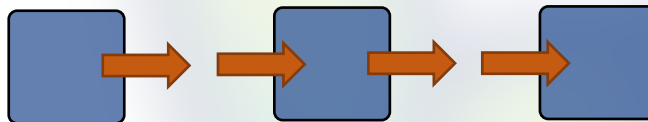
Materializing

Components are static until they run

```
val graph = source.via(flow).to(sink)
```

```
val result = graph.run()
```

↑
materialized value



A graph is a "blueprint" for a stream

Running a graph allocates the right resources

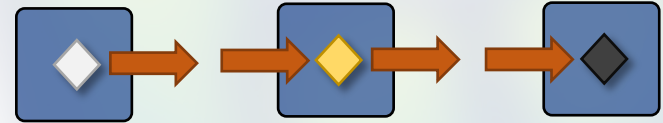
- actors, thread pools
- sockets, connections
- etc – everything is transparent

Running a graph = materializing

Materialized Values

Materializing a graph = materializing *all* components

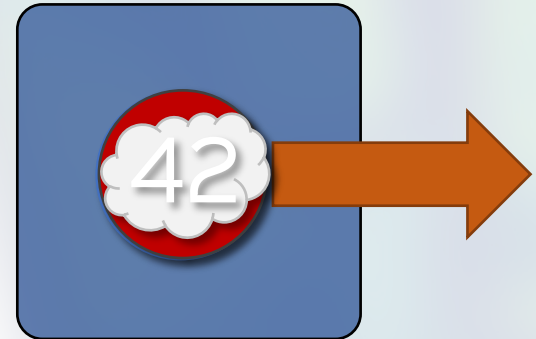
- each component produces a materialized value when run
- the graph produces a single materialized value
- our job to choose which one to pick



A component can materialize multiple times

- you can reuse the same component in different graphs
- different runs = different materializations!

A materialized value can be ANYTHING.



Recap

Materializing a graph = materializing *all* components

- each component produces a materialized value when run
- the graph produces a single materialized value
- our job to choose which one to pick

A component can materialize multiple times

A materialized value can be ANYTHING

Akka rocks

