Stateful Functions

戴资力 | Tzu-Li (Gordon) Tai
Software Engineer, Ververica

FLINK FORWARD # ASIA
实时即未来 # Real-time Is The Future

FLINK FORWARD

Contents

- Get the feeling of what kind of applications you can build with Stateful Functions
- Get to know the core concepts
- Understand how it is realized as a Flink streaming graph
- Learn how to get started

Stateful Functions

- A new SDK for building distributed stateful applications

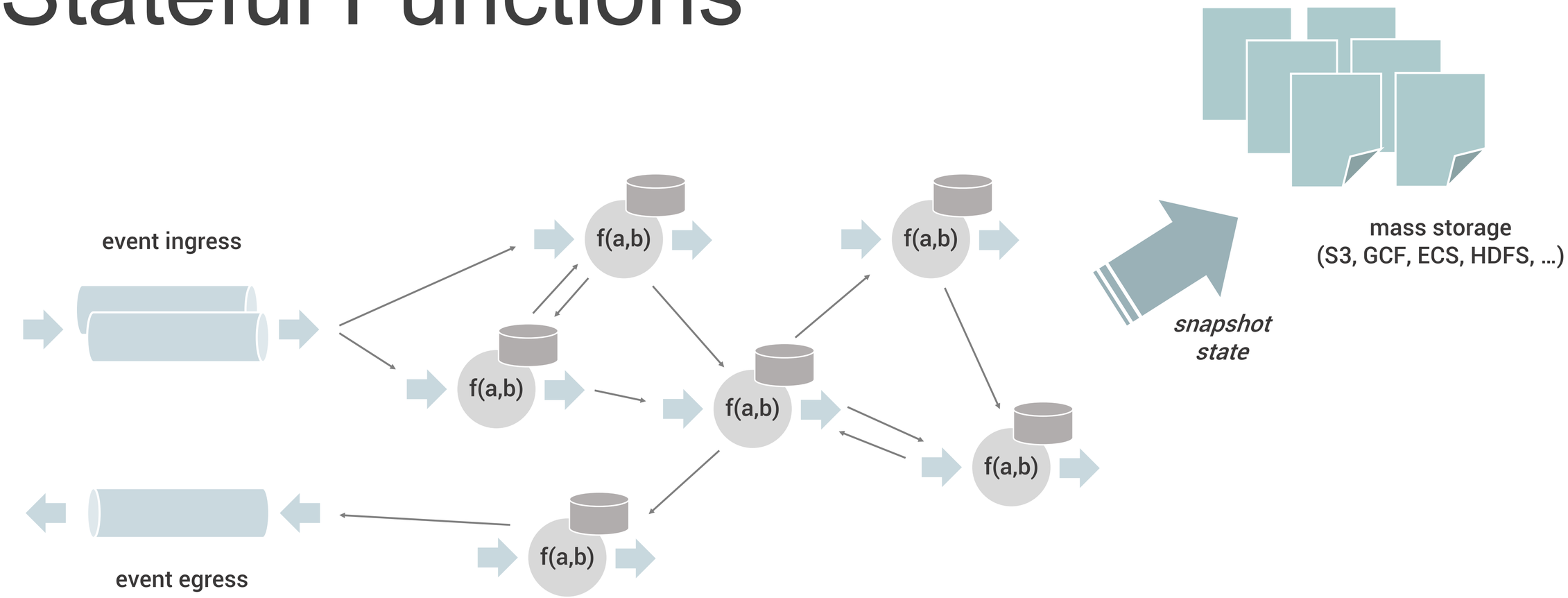
- The SDK aims to simplify some of the challenges in building distributed stateful applications.
  - Scale
  - Reliable communication
  - Consistent state handling

- An implementation on top of Apache Flink, that leverages:
  - Distributed coordination
  - Multi TB state
  - Low latency, and high throughput networking
  - And much more

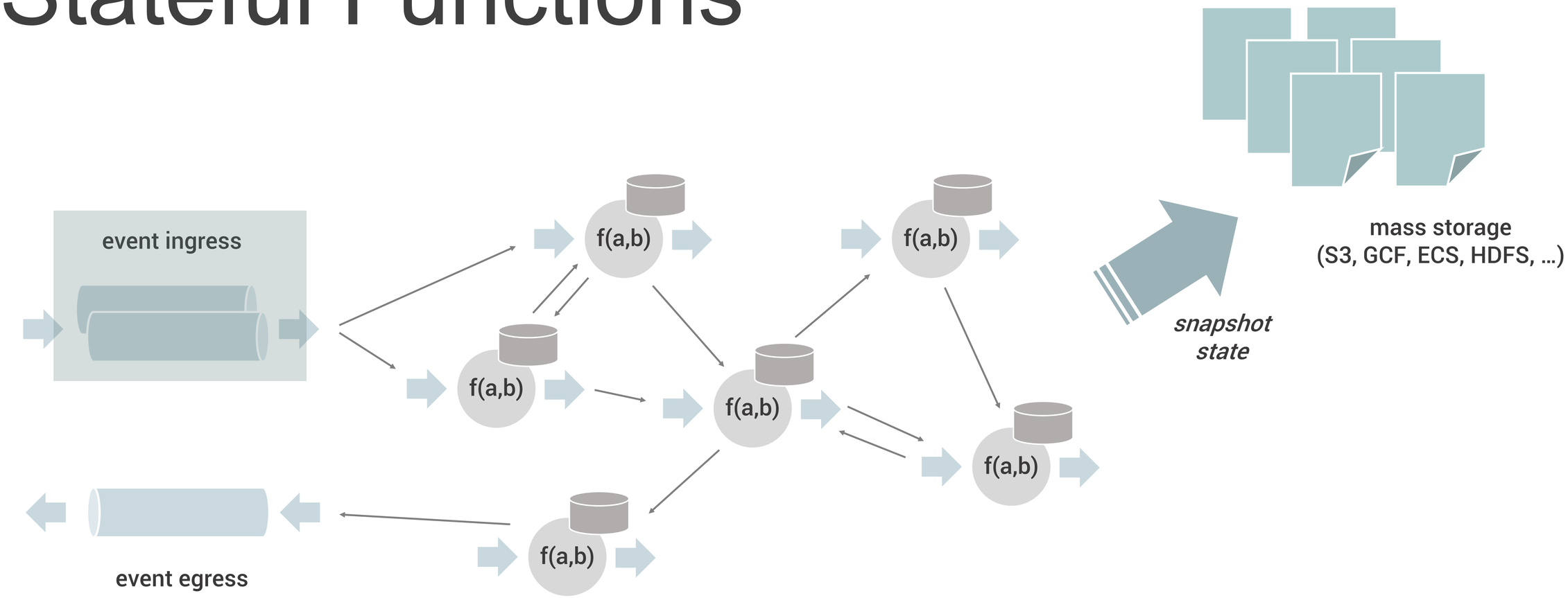
Stateful Functions

- Based on the concept of addressable virtual functions + local state
- Functions send events to each other with arbitrary addressing, no restriction to DAG
- State and messaging are consistent with exactly-once semantics
- Consistently checkpointed to an external storage

Stateful Functions

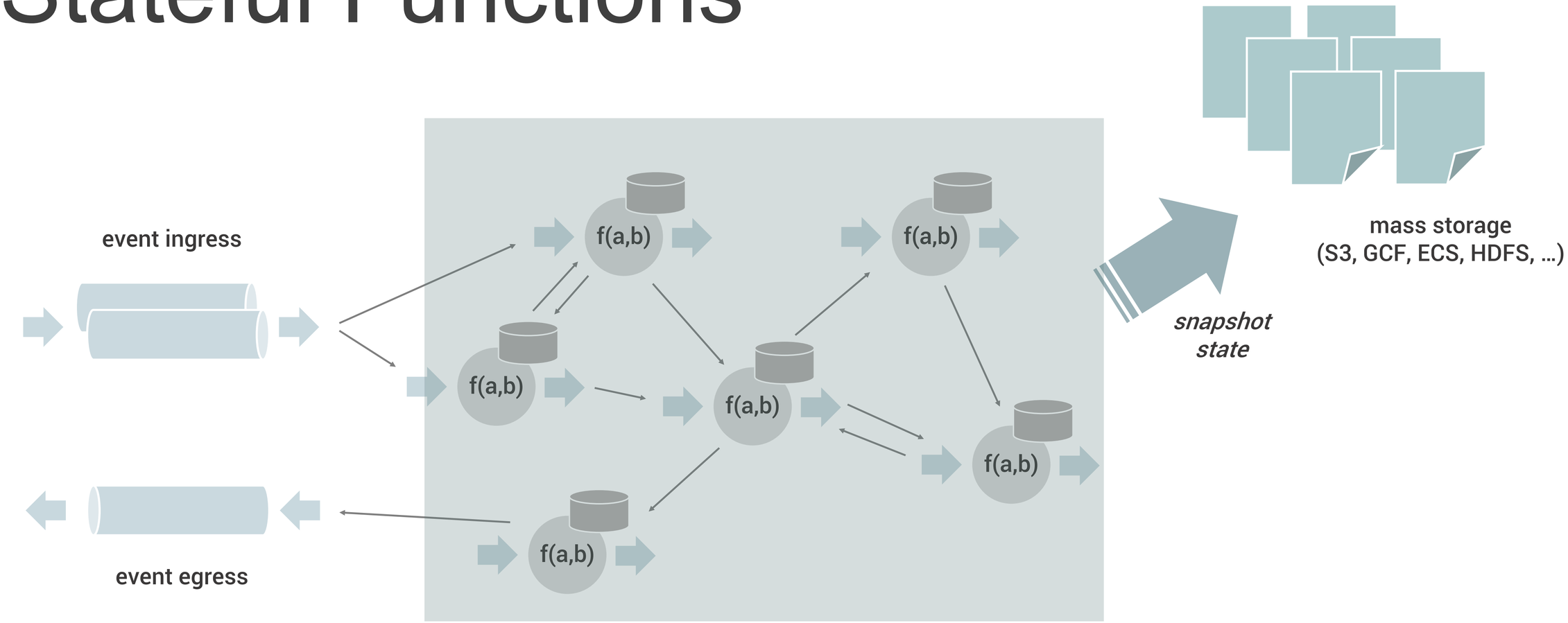


Stateful Functions



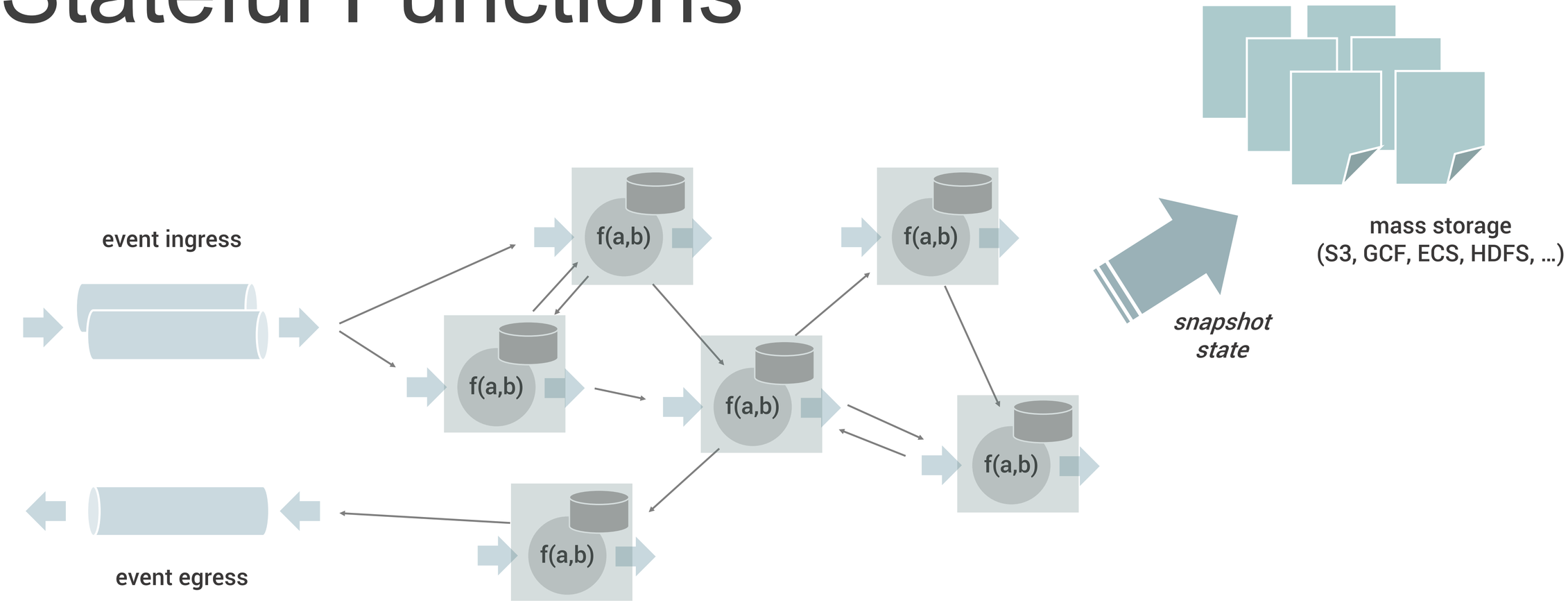
Event ingresses supply events that trigger functions

Stateful Functions



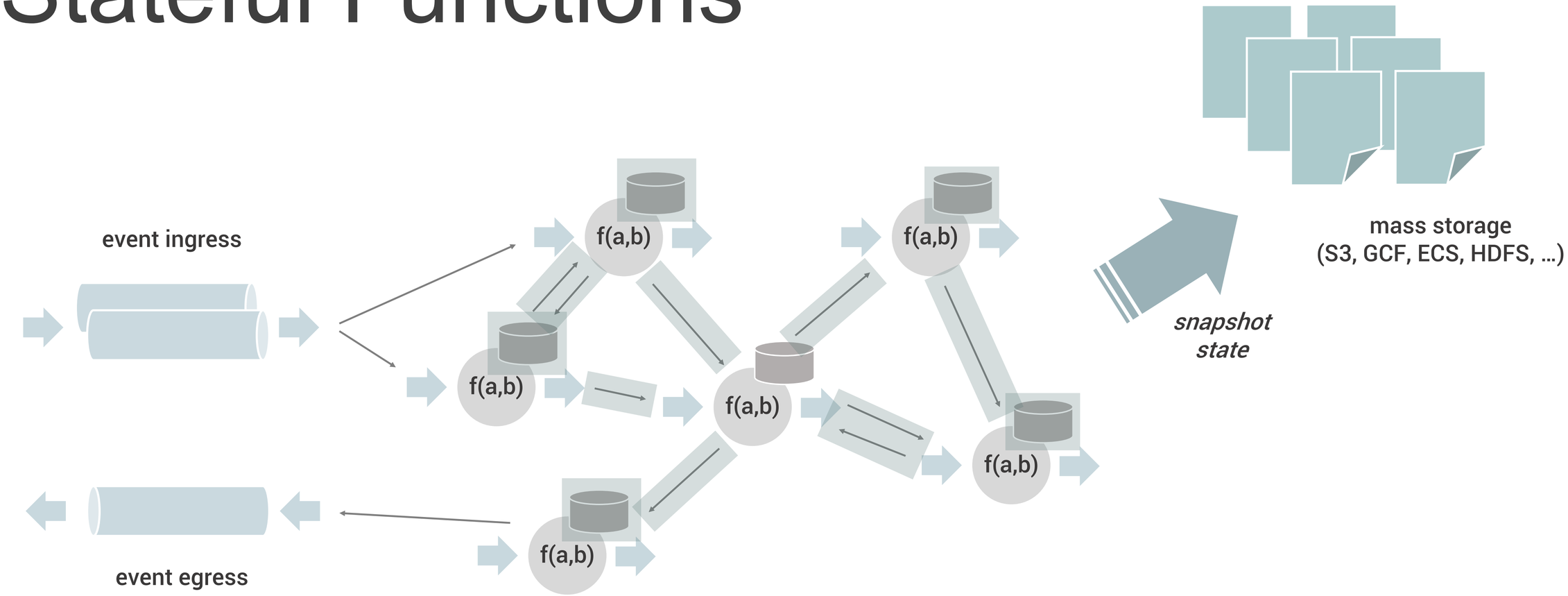
Multiple functions send event to each other
Arbitrary addressing, no restriction to DAG

Stateful Functions



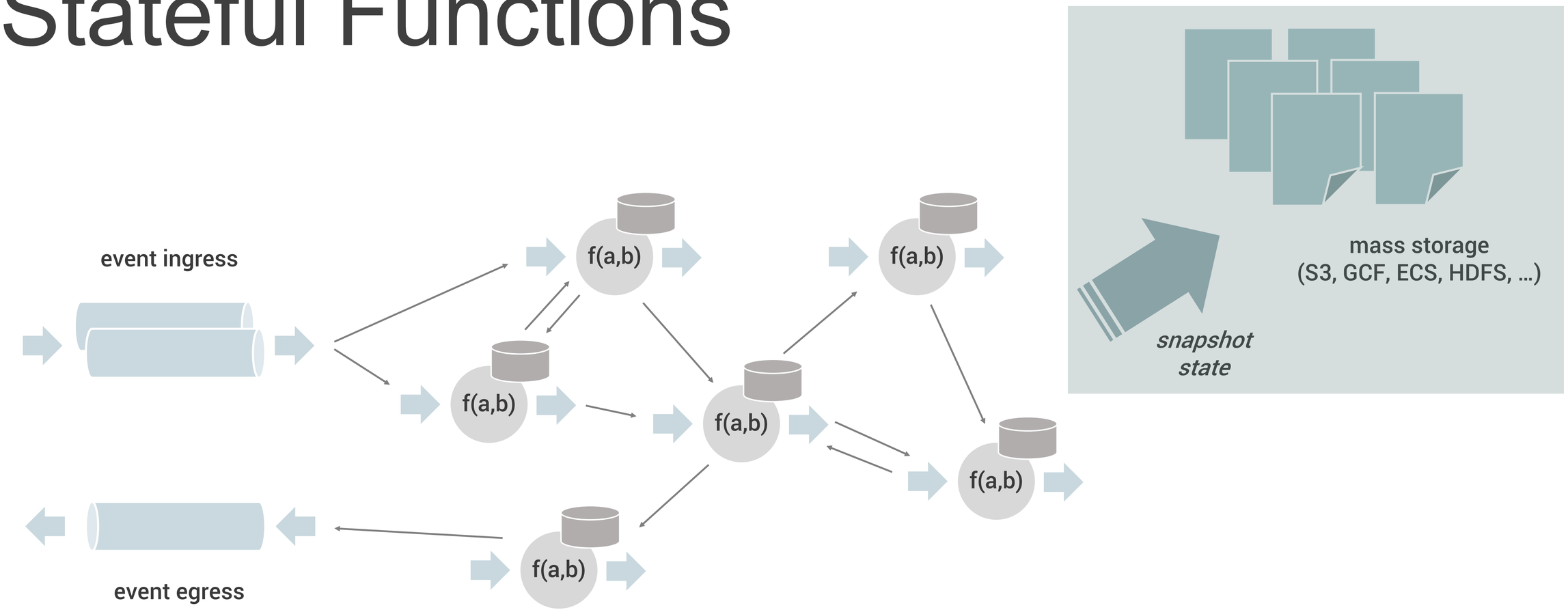
Functions have locally embedded state

Stateful Functions



State and messaging are consistent
with exactly-once semantics

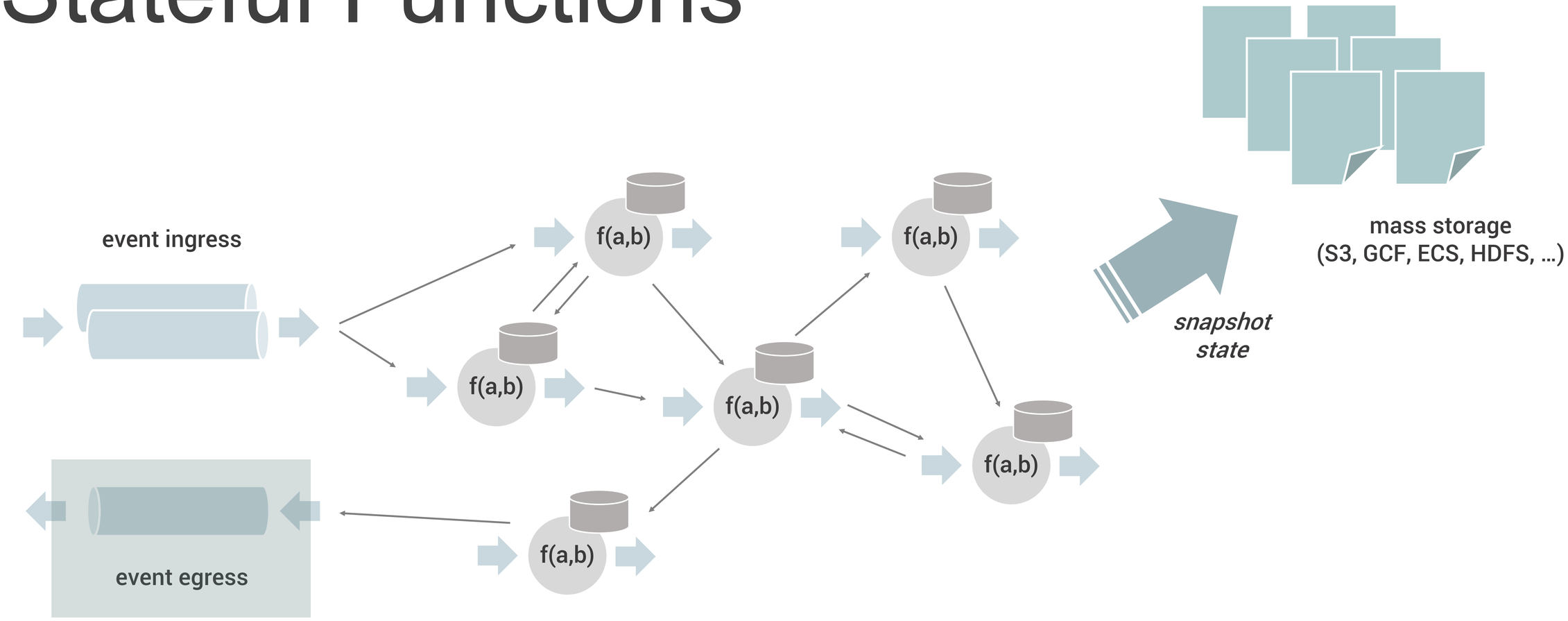
Stateful Functions



No database required

All persistence goes directly to blob storage

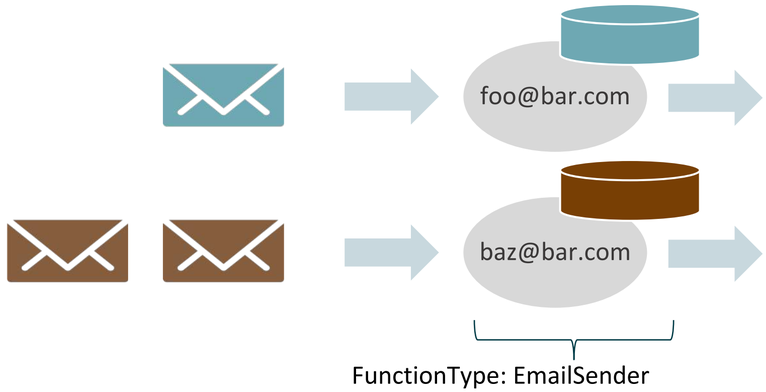
Stateful Functions



Event egresses to respond via event streams

SDK Concepts: Address

- Each function is associated with a **FunctionType** (~ class) and an **id** (~ instance)
- A combination of the two defines a unique stateful function instance ⇒ **Address**
- An **Address** is a logical and not physical (no service discovery needed, just use the logical address to send messages)
- The stateful function instances are partitioned by an **Address** across all the available compute cores
- For example
  FunctionType = EmailSender, **id** = foo@bar.com



SDK Concepts: Router

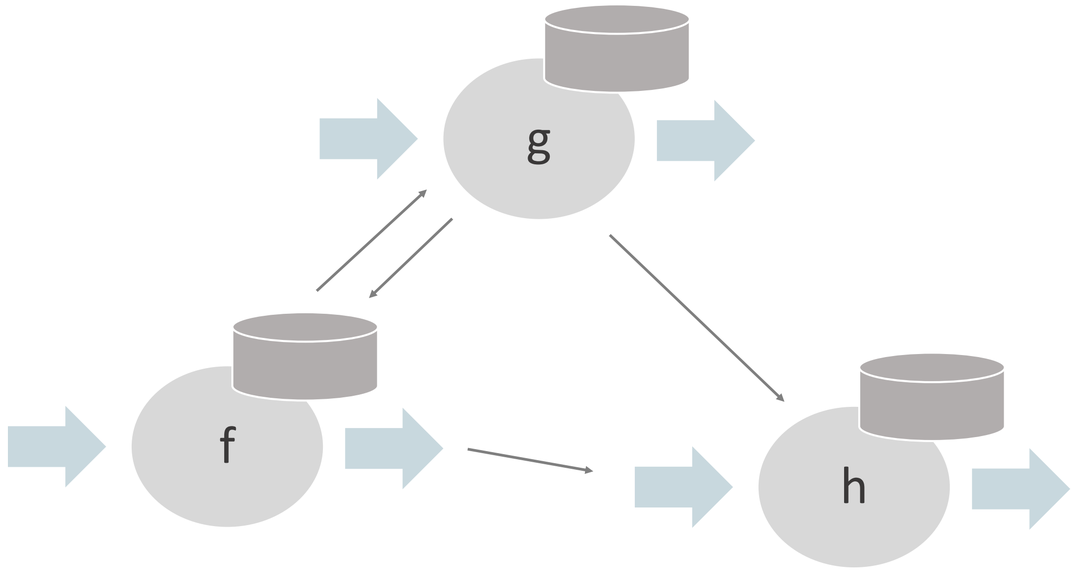
- A **Router** determines what functions to invoke for a given ingress message
- It is being invoked for each incoming ingress message
- It may chose to drop messages, forward them, or fanout a message to multiple functions.
- Multiple routers can be attached to a single **Ingress**

```
class GreetRouter implements Router<GreetRequest> {

    @Override
    public void route(GreetRequest message, Downstream<GreetRequest> downstream) {
        Address address = new Address(GreetStatefulFunction.TYPE, message.getUserId());
        downstream.forward(address, message);
    }
}
```

SDK Concepts: Stateful Functions

- Applications are a collection of **StatefulFunction**s bundled together

- A stateful function reacts to incoming events and can
  - Perform a local computation
  - Access & modify its local state
  - Send messages to any other stateful function
  - Send messages to external systems



SDK Concepts: Stateful Functions

```java
class GreetStatefulFunction implements StatefulFunction {

  static final FunctionType TYPE = new FunctionType( namespace: "ververica",  type: "greeter");

  @Override
  public void invoke(Context context, Object input) {
    GreetRequest request = (GreetRequest) input;
    PersonalizedGreeting response = computePersonalizedGreeting(request);

    Address address = new Address(EmailSenderFn.TYPE, response.getUserEmail());

    context.send(address, response);
  }
```

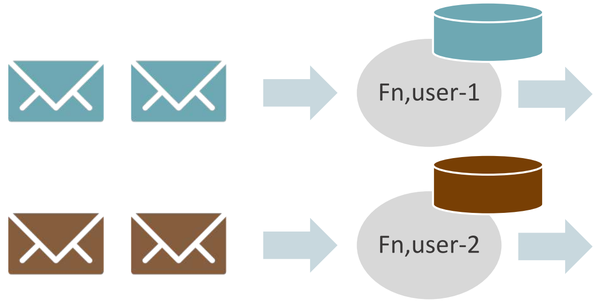
SDK Concepts: State

- All stateful functions may contain ... well state.
- A persistent, and fault tolerant state is declared by adding a **PersistedValue** class field
- The data can be accessed as if it was a regular class field

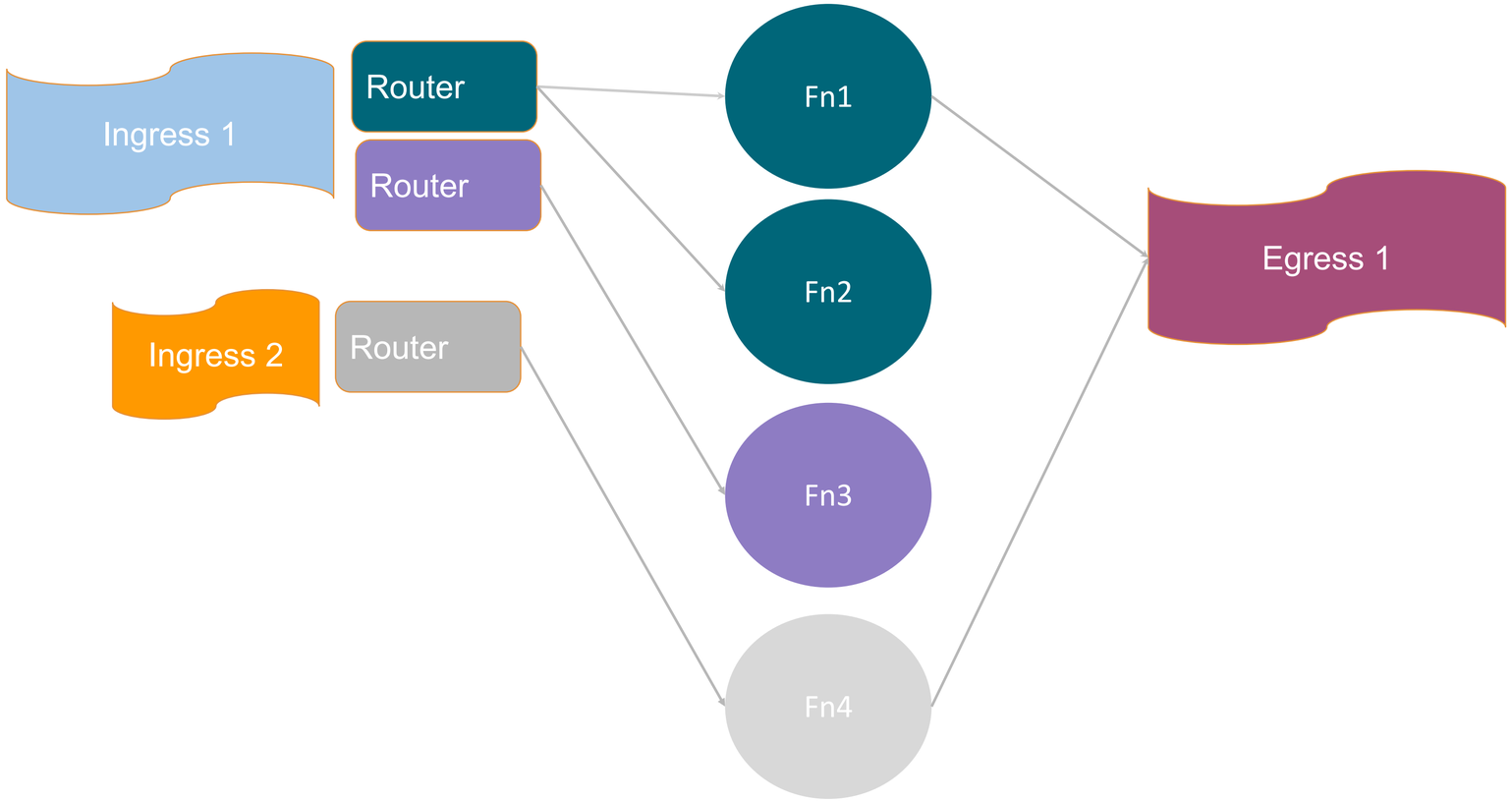
```java
@Persisted
private final PersistedValue<Integer> seenCount = PersistedValue.of( name: "seen-count", Integer.class);

private PersonalizedGreeting computePersonalizedGreeting(GreetRequest request) {
  final String name = request.getFirstName();
  final int seen = seenCount.getOrDefault( orElse: 0);
  seenCount.set(seen + 1);

  String greeting = greetText(name, seen);

  return new PersonalizedGreeting(request, greeting);
}
```

SDK Concepts: Logical Instances

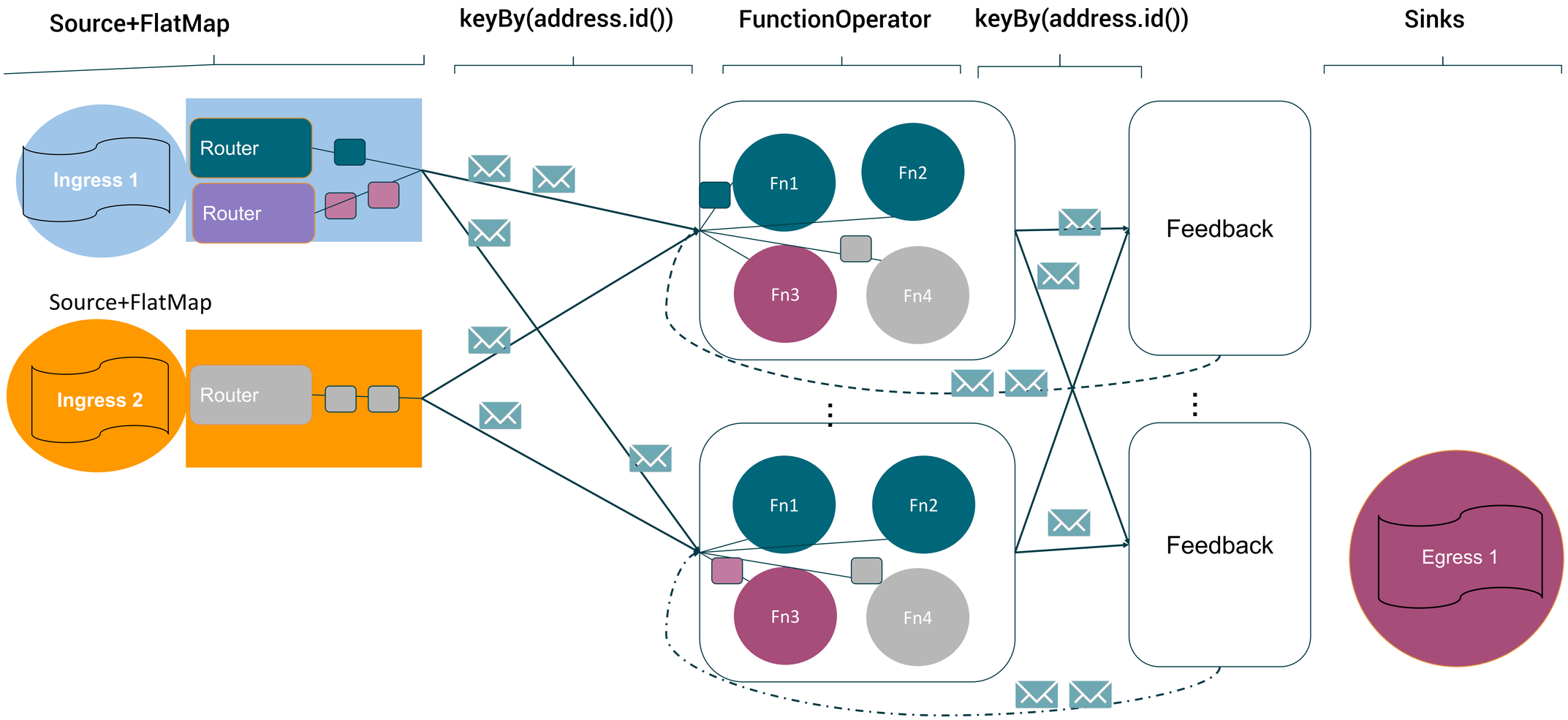
- There is a single (logical) instance of a function per an **Address**
- An instance of a **StatefulFunction** is created, on demand, transparently, by the runtime upon receiving a message addressed to it.
- Since there is a single instance per Address.
  The messages sent to an **Address** are processed sequentially by a single thread.
- Messages sent from function A to function B are always received in FIFO order



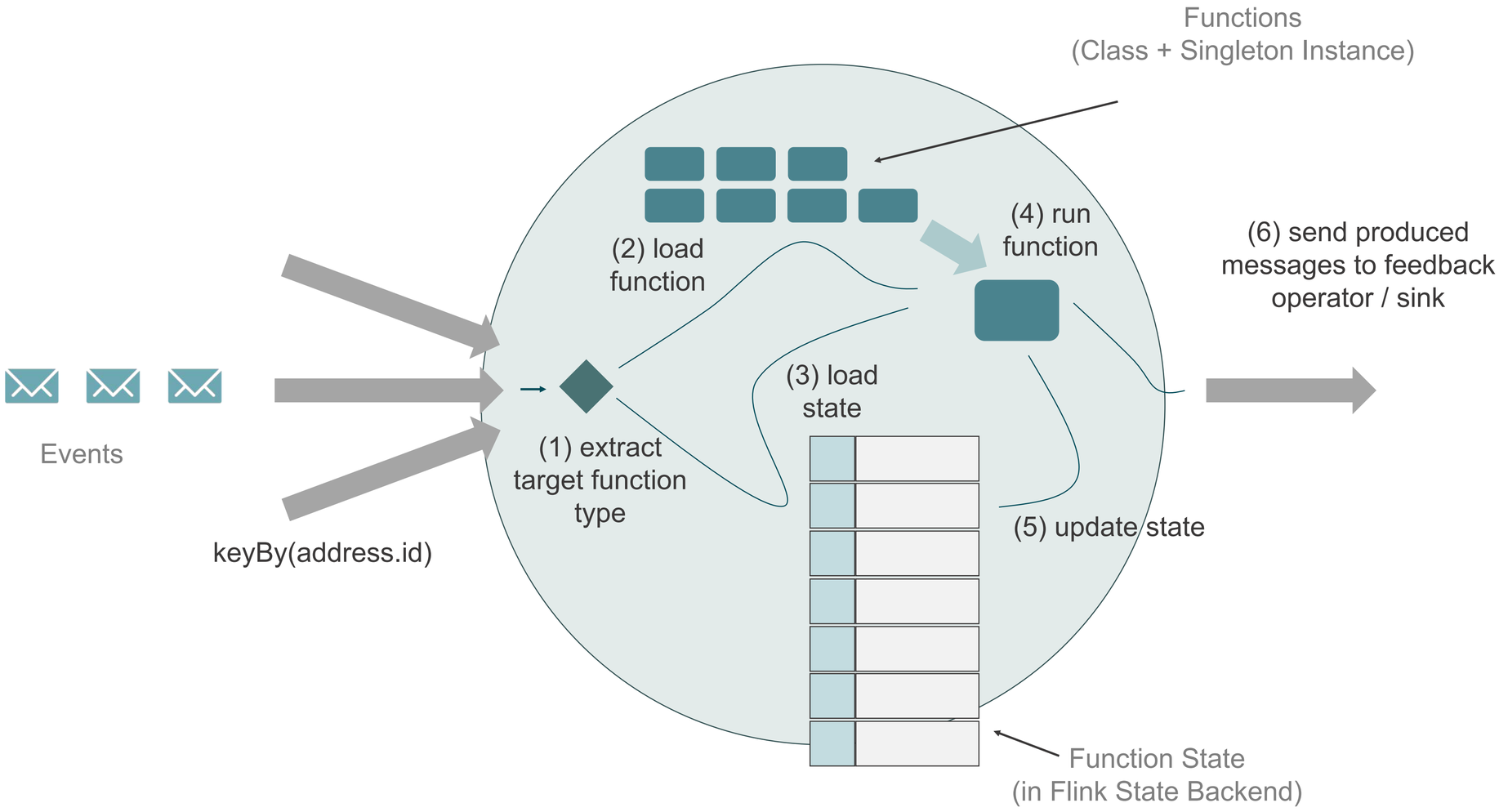
A Peek Under the Hood



Translated Flink Stream Graph



Function Operator



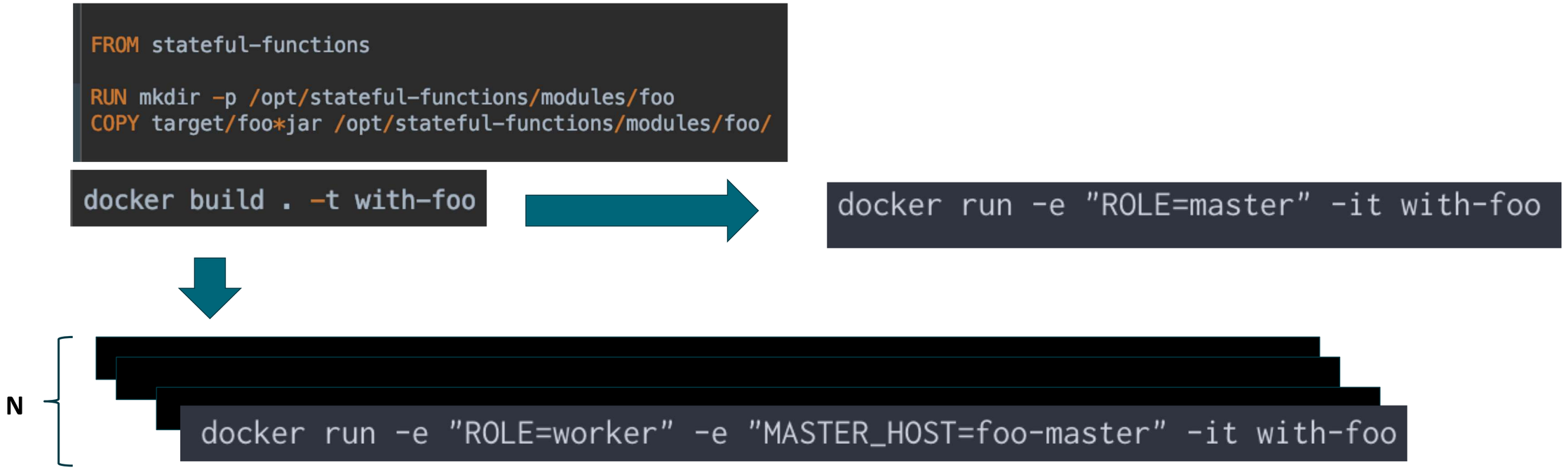
Deployment: Modules

- Modules define an entry point.
- Multiple modules can exists within an Application
- They allow configuring:
  - Ingress/Egress definitions
  - Routers
  - Stateful function factory methods

```
class GreetingModule implements StatefulFunctionModule {

  @Override
  public void configure(Map<String, String> globalConfiguration, Binder binder) {
    configureGreetingIngress(globalConfiguration, binder);
    configureGreetingEgress(globalConfiguration, binder);
    configureGreetingFunctions(globalConfiguration, binder);
    configureGreetingRouters(globalConfiguration, binder);
  }
```

Deployment: Dockerized

- Stateful functions, ships with a self contained Docker image



Deployment: Existing Flink Cluster

- Add a dependency to your stateful functions application

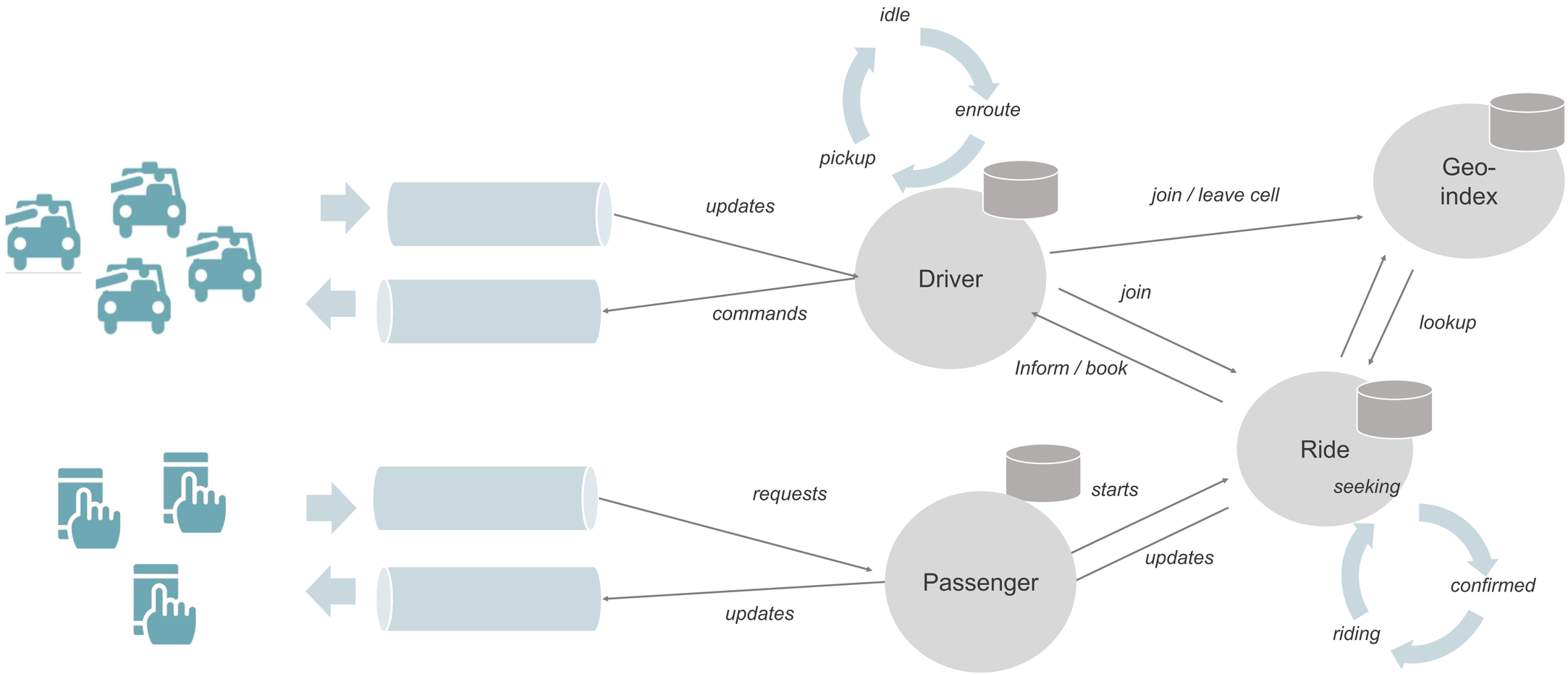
```
<dependency>
    <groupId>com.ververica</groupId>
    <artifactId>stateful-functions-flink-distribution</artifactId>
    <version>${version}</version>
</dependency>
```

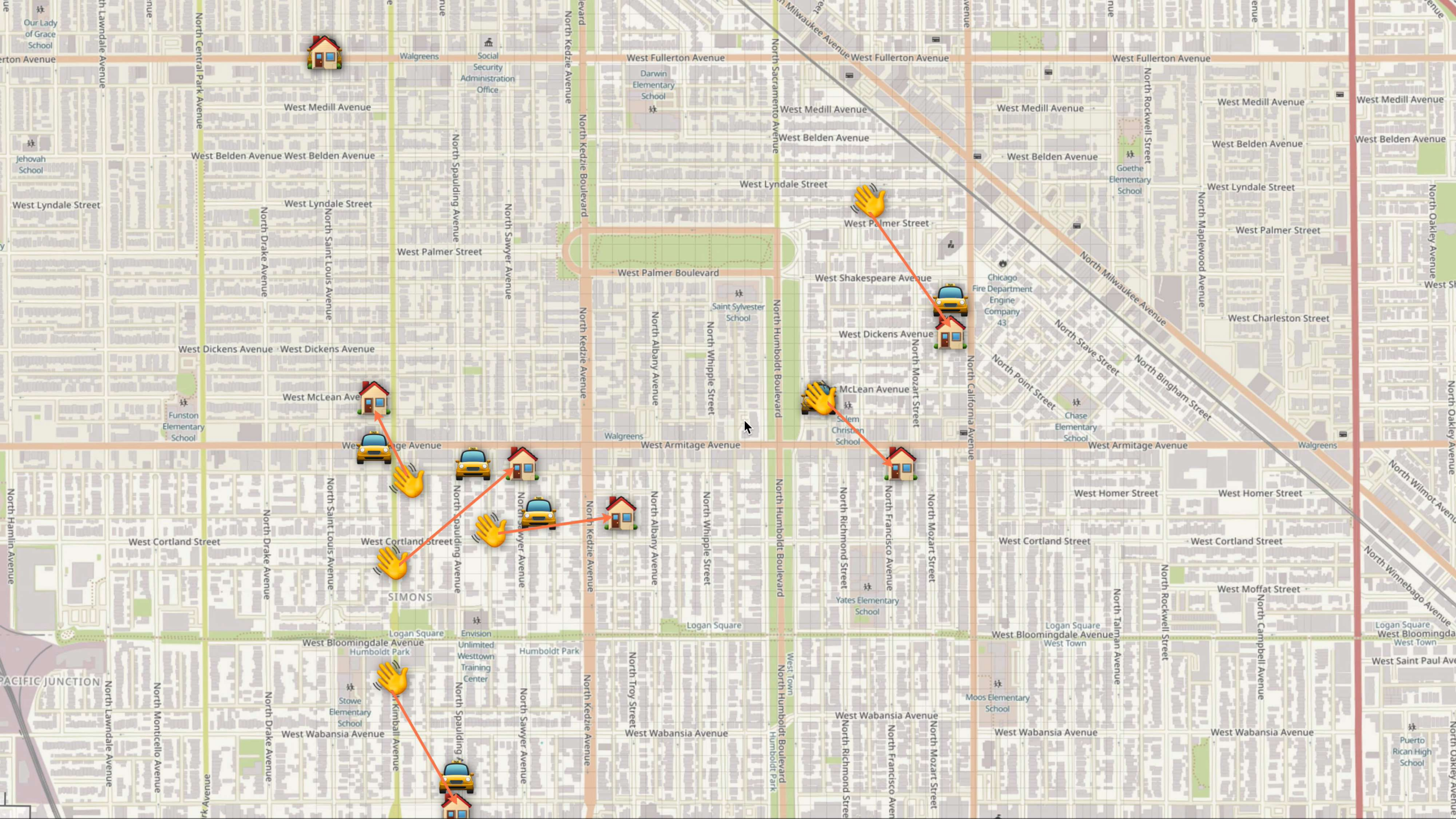
- Create a jar-with-dependency and submit

```
bin/flink run -d -jar with-foo.jar
```

Example: Ride Sharing App

Example: Ride Sharing App




West Fullerton Avenue
West Medill Avenue
West Belden Avenue
West Lyndale Street
West Palmer Street
West Palmer Boulevard
West Shakespeare Avenue
West Dickens Avenue
West McLean Avenue
West Armitage Avenue
West Cortland Street
West Homer Street
West Moffat Street
West Bloomingdale Avenue
West Wabansia Avenue
West Charleston Street
North Milwaukee Avenue
North Kedzie Boulevard
North Kedzie Avenue
North Sacramento Avenue
North Humboldt Boulevard
North California Avenue
North Central Park Avenue
North Lawndale Avenue
North Drake Avenue
North Saint Louis Avenue
North Spaulding Avenue
North Sawyer Avenue
North Albany Avenue
North Whipple Street
North Mozart Street
North Richmond Street
North Francisco Avenue
North Rockwell Street
North Maplewood Avenue
North Talman Avenue
North Campbell Avenue
North Stave Street
North Point Street
North Bingham Street
North Wilmot Avenue
North Winnebago Avenue
North Oakley Avenue
North Hamlin Avenue
North Monticello Avenue
North Troy Street
Our Lady of Grace School
Walgreens
Social Security Administration Office
Darwin Elementary School
Jehovah School
Goethe Elementary School
Saint Sylvester School
Chicago Fire Department Engine Company 43
Funston Elementary School
Salem Christian School
Chase Elementary School
Yates Elementary School
Moos Elementary School
Stowe Elementary School
Envision Unlimited Westtown Training Center
Puerto Rican High School
SIMONS
PACIFIC JUNCTION
Logan Square
Humboldt Park
West Town

FLINK FORWARD

THANKS