



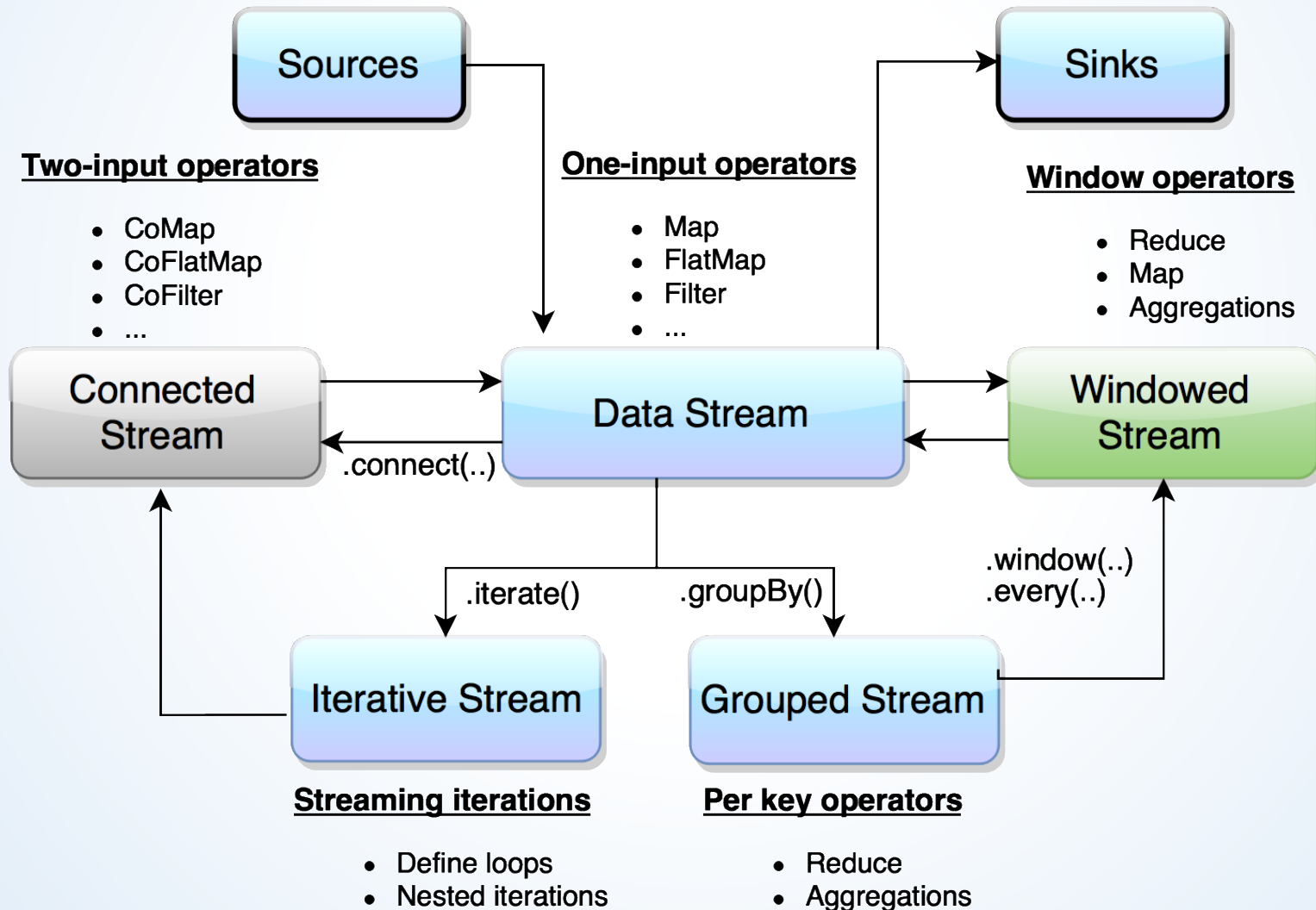
Flink Streaming API

Rolling Word Count

```
DataStream<String> text = env.socketTextStream(host,
                                                port);

DataStream<Tuple2<String, Integer>> result = text
    .flatMap((str, out) -> {
        for (String token : value.split("\\W")) {
            out.collect(new Tuple2<>(token, 1));
        }
    })
    .groupBy(0)
    .sum(1);
```

Streaming programming model



Windowing

- Flexible policy based windowing
- Trigger and Eviction policies
- Built-in policies:
 - Time: ***Time.of***(length, TimeUnit/Custom timestamp)
 - Count: ***Count.of***(windowSize)
 - Delta: ***Delta.of***(Delta function, Start value, threshold)
- Window transformations (on StreamWindows):
 - mapWindow
 - reduceWindow
- Custom trigger and eviction policies can also be implemented easily

Binary stream transformations

- Apply shared transformations on streams of different types.
- Shared state between transformations
- *CoMap, CoFlatMap, CoReduce...*

```
interface CoMapFunction<IN1, IN2, OUT> {  
    OUT map1(IN1 value);  
    OUT map2(IN2 value);  
}
```

State in Flink Streaming

- Two state access patterns
 - Local (Task) state
 - Partitioned (Key) state
- OperatorState accessible from RuntimeContext in Rich functions
 - `ctx.getOperatorState(...)`
- Exactly-once semantics by checkpointing
(See *Paris's demo*)

```
interface OperatorState<T> {  
    T value();  
    void update(T value);  
}
```

State example

```
class CountPerKey extends RichMapFunction<String, Integer> {

    OperatorState<Integer> counter;

    @Override
    public Integer map(String value) throws Exception {
        counter.update(counter.value() + 1);
        return counter.value();
    }

    @Override
    public void open(Configuration conf) throws IOException {
        RuntimeContext ctx = getRuntimeContext();
        counter = ctx.getOperatorState("c", 0, true);
    }
}
```

What is on the USB stick

- flink-boss: source and readmes
- flink-doc: some API docs
- Eclipse if you need on IDE
- Alternatively you can
git clone github.com/mbalassi/boss-tutorial
if you are confident with git, mvn and your own IDE