KAFKA SUMMIT
LONDON 2023

Dataflows for machine learning operations

Andrei Paleyes and Alex Rakowski

SELDON

KAFKA SUMMIT
LONDON 2023

Part I – Dataflow and Streams

SELDON

What is ML@CL group?

Machine
Learning
@
Computer
Lab

ML@CL

UNIVERSITY OF CAMBRIDGE

Dataflow research at ML@CL

Towards better data discovery and collection with flow-based programming

Christian Cabrera

RESEARCH-ARTICLE OPEN ACCESS

Causal fault localisation in dataflow systems

Authors: Andrei Paleyes, Neil David Lawrence Authors Info & Claims

EuroMLSys '23: Proceedings of the 3rd Workshop on Machine Learning and Sy

140–147 • https://doi.org/10.1145/3578356.3592593

RESEARCH-ARTICLE OPEN ACCESS

An empirical evaluation of flow based programming in the machine learning deployment context

Authors: Andrei Paleyes, Christian Cabrera, Neil D. Lawrence Authors Info & Claims

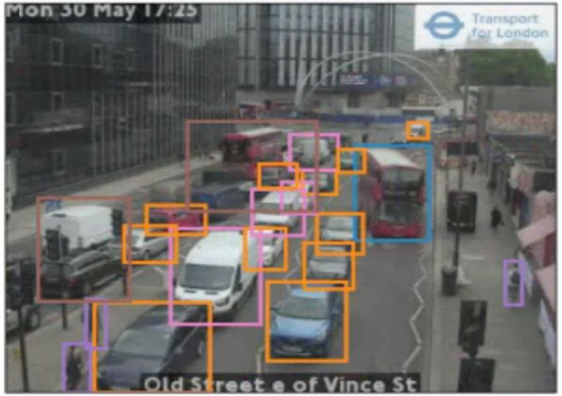
CAIN '22: Proceedings of the 1st International Conference on AI Engineering: Software Engineering for AI • May 2022 • Pages

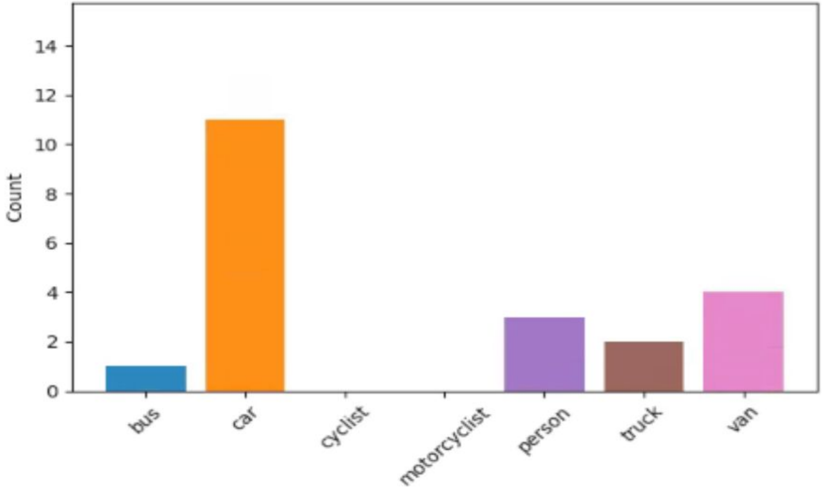
Dataflow graphs as complete causal graphs

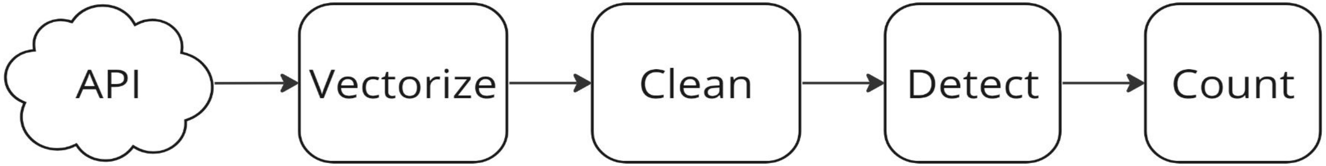
Andrei Paleyes*[1], Siyuan Guo*[12], Bernhard Schölkopf[2], Neil D. Lawrence[1]
[1]Department of Computer Science and Technology, University of Cambridge
[2]Max Planck Institute for Intelligent Systems

Open Source: Seldon Core

Inference graph

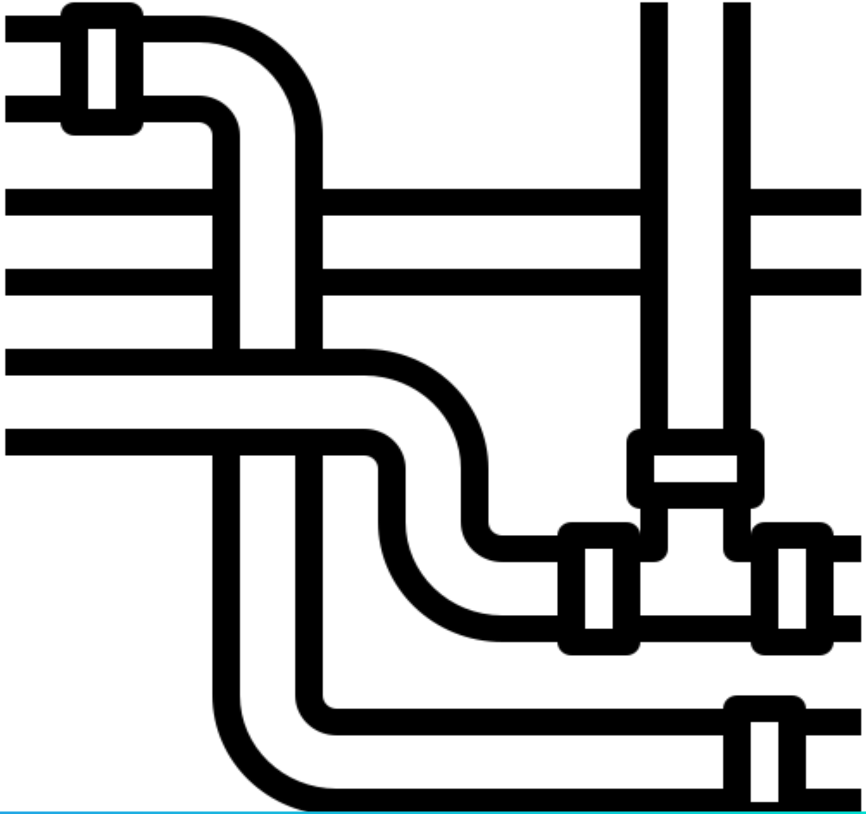




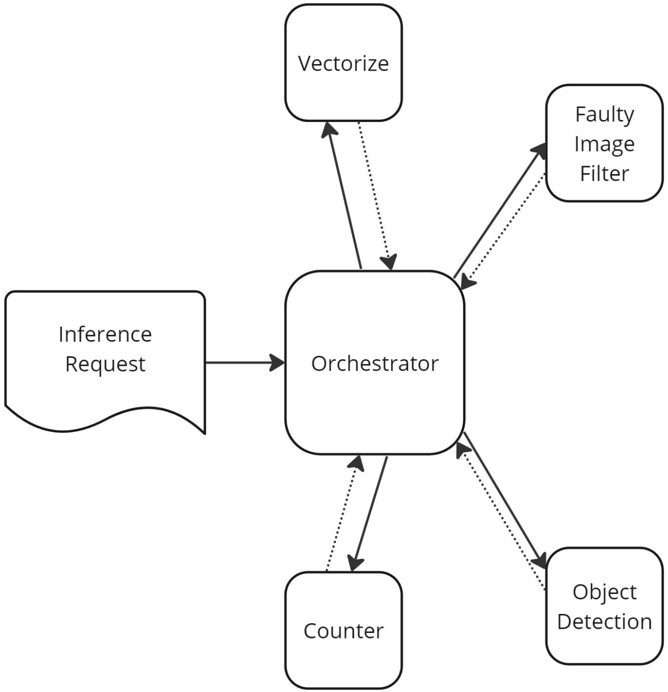


Pipelines

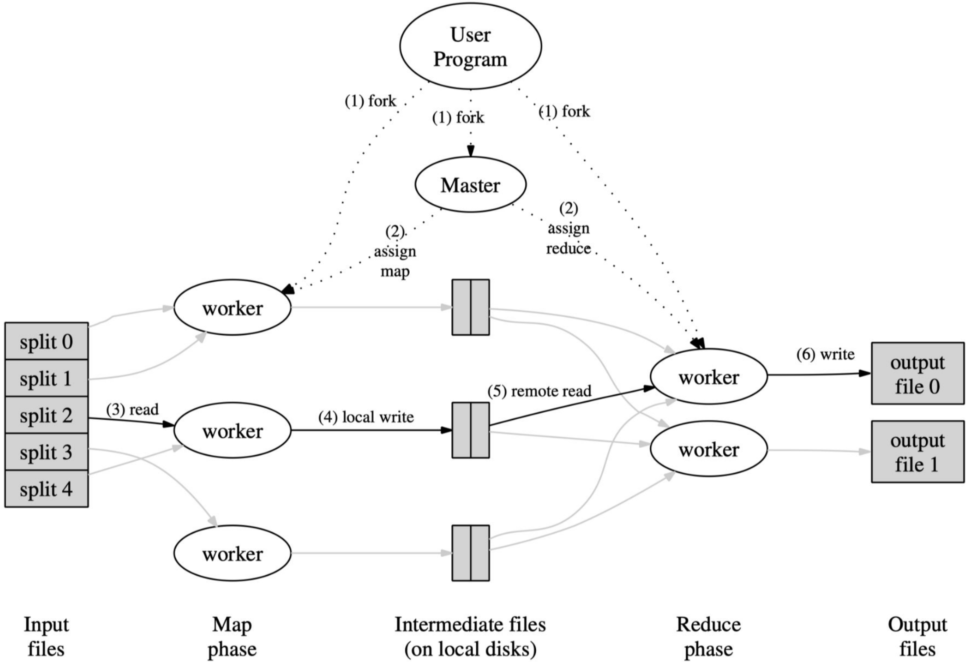
```
apiVersion: mlops.seldon.io/v1alpha1
kind: Pipeline
metadata:
 name: road-counter
spec:
 steps:
   - name: vectorize
   - name: faulty_image_filter
     inputs:
     - vectorize.outputs
   - name: object_detection
     inputs:
     - vectorize.outputs
     - faulty_image_filter.outputs
. . . . .
 output:
   steps:
   - counter
```



Core V1 - Central orchestration



Dataflow architecture

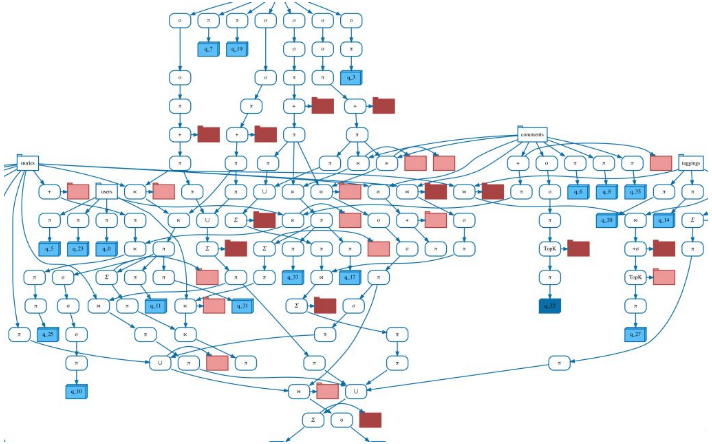


Dean, J. & Ghemawat, S. (2008). MapReduce: simplified data processing on large clusters. Communications of the ACM, 51(1), 107-113.

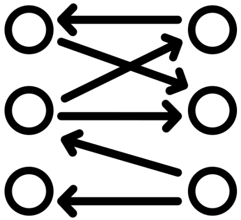
Dataflow vs control flow

“In control flow, the processor follows explicit order, executing instructions one after another. In **dataflow**, by contrast, an instruction is ready to execute as soon as all its inputs are available.”

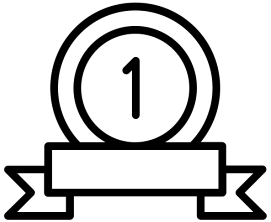
M. Schwarzkopf, *The Remarkable Utility of Dataflow Computing*



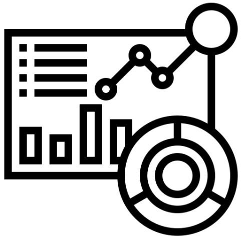
Dataflow features



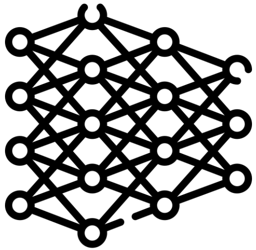
Separation of data and operations



Data as a first class citizen

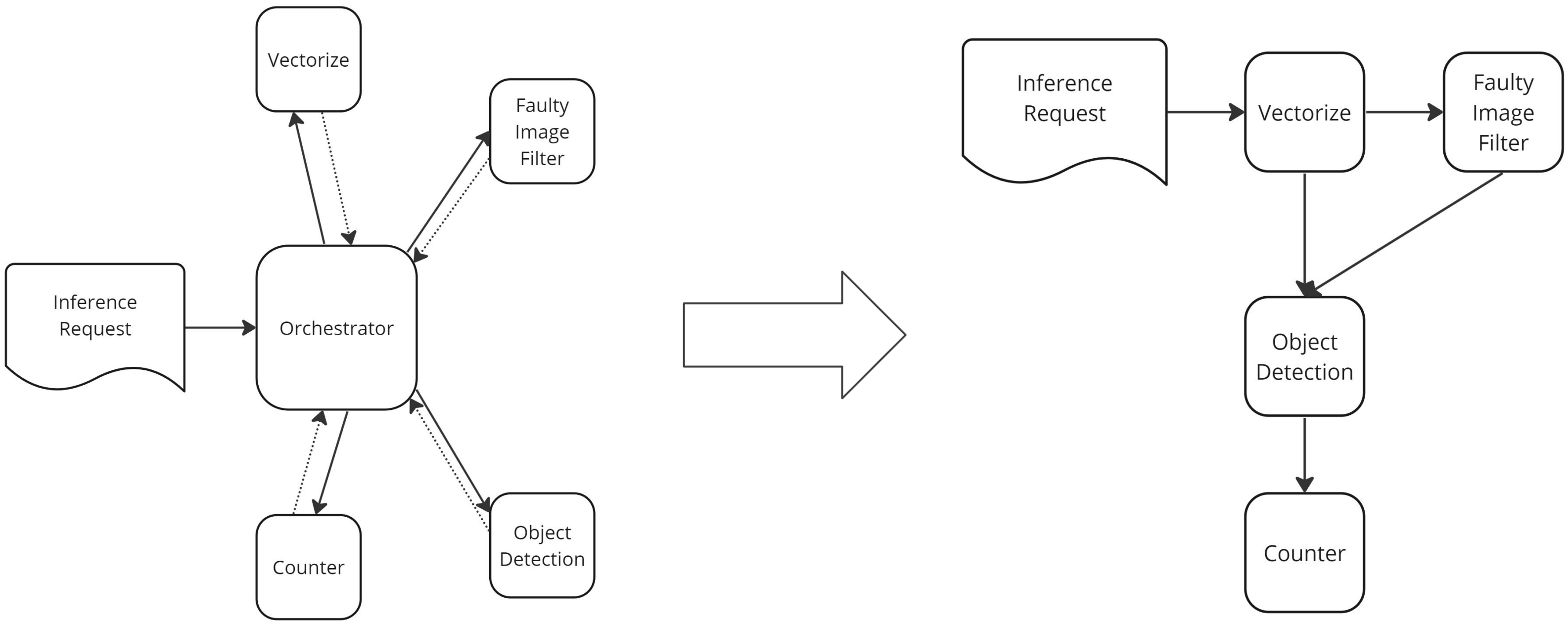


Dataflow graph of the entire system

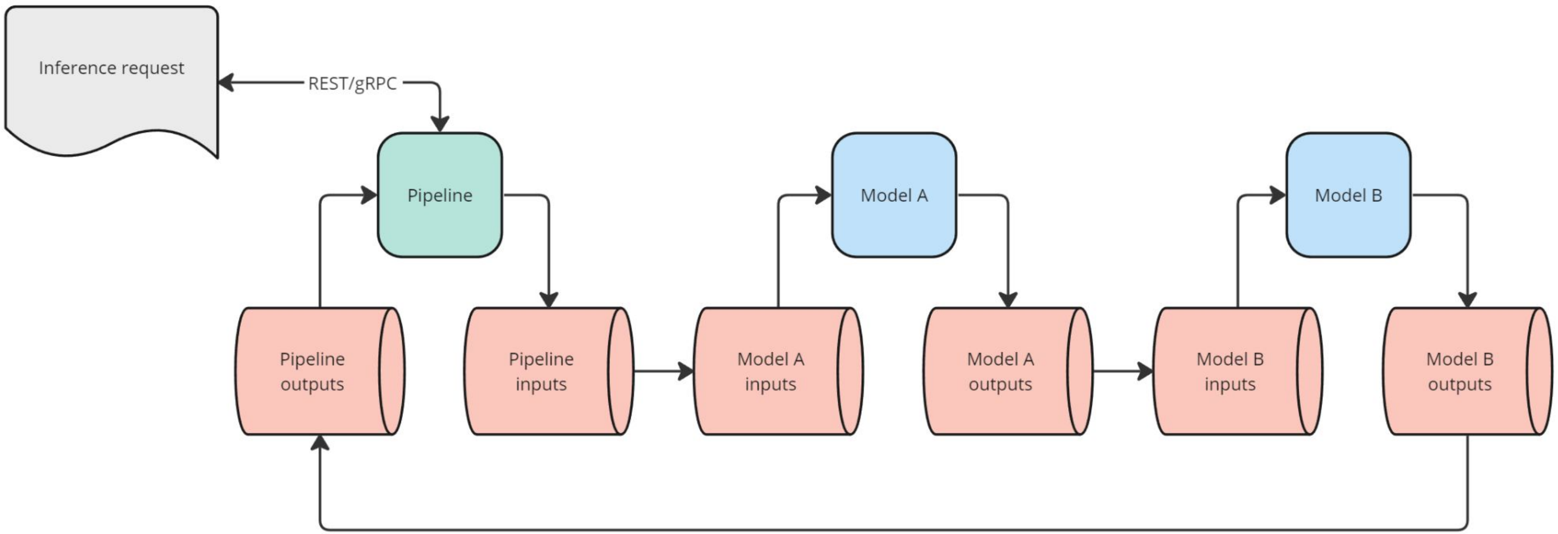


Decentralisation

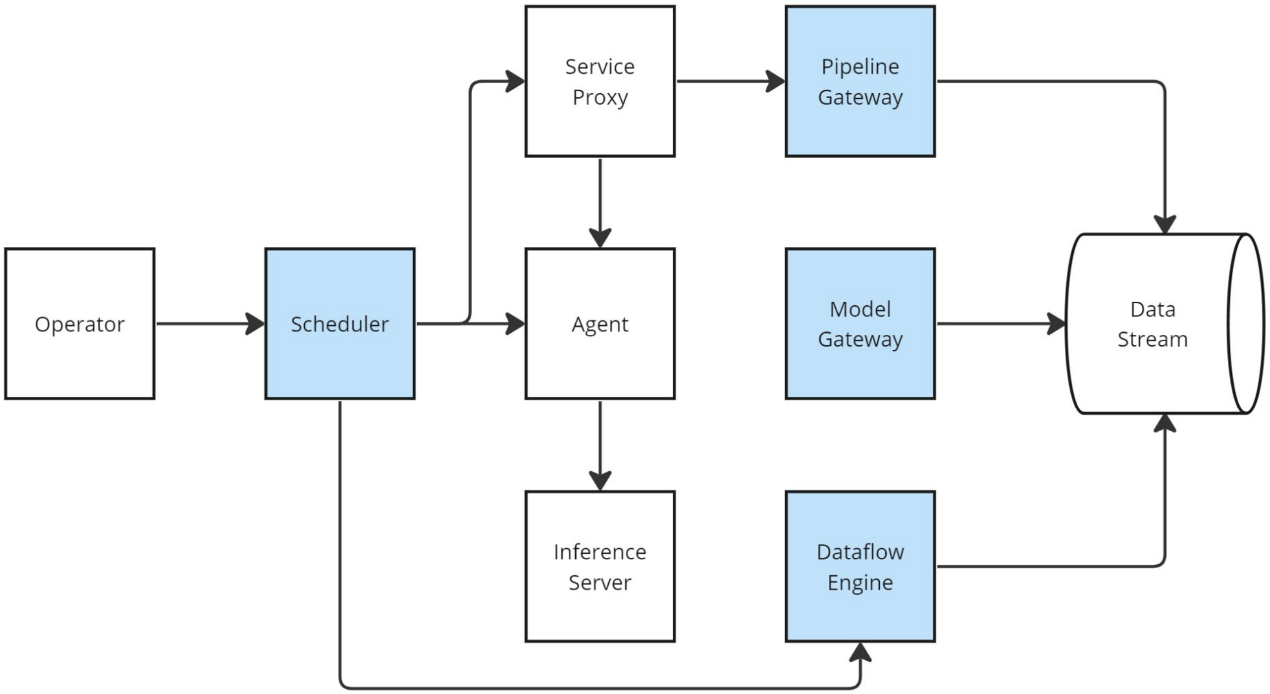
Dataflow and inference graph



Role of streaming in dataflow



Control plane

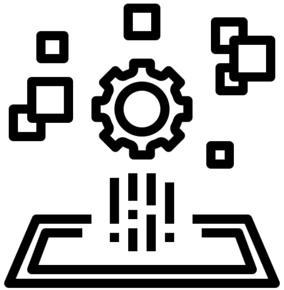


KAFKA SUMMIT
LONDON 2023

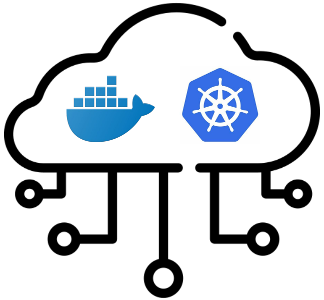
Part II – Implementation

SELDON

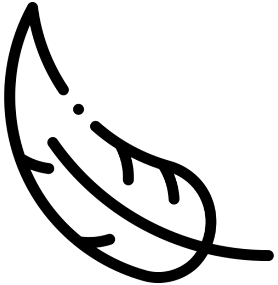
Tech Stack — Constraints



OSS



Platform agnostic



Lightweight

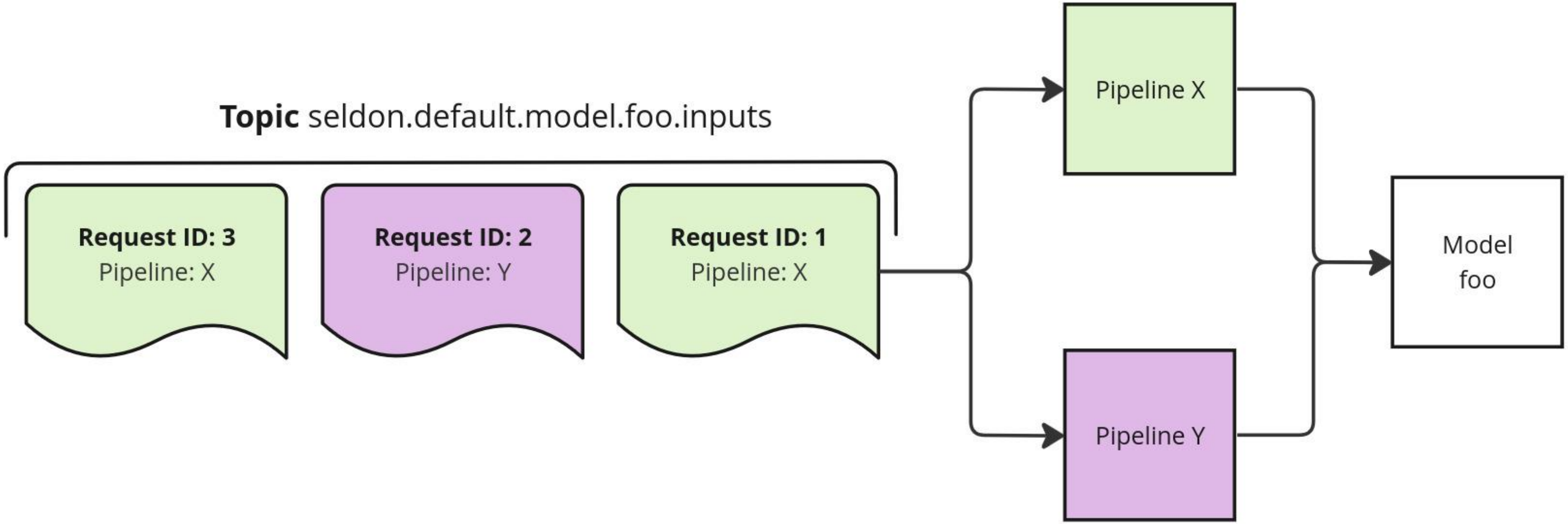
Tech Stack — Contenders

kafka

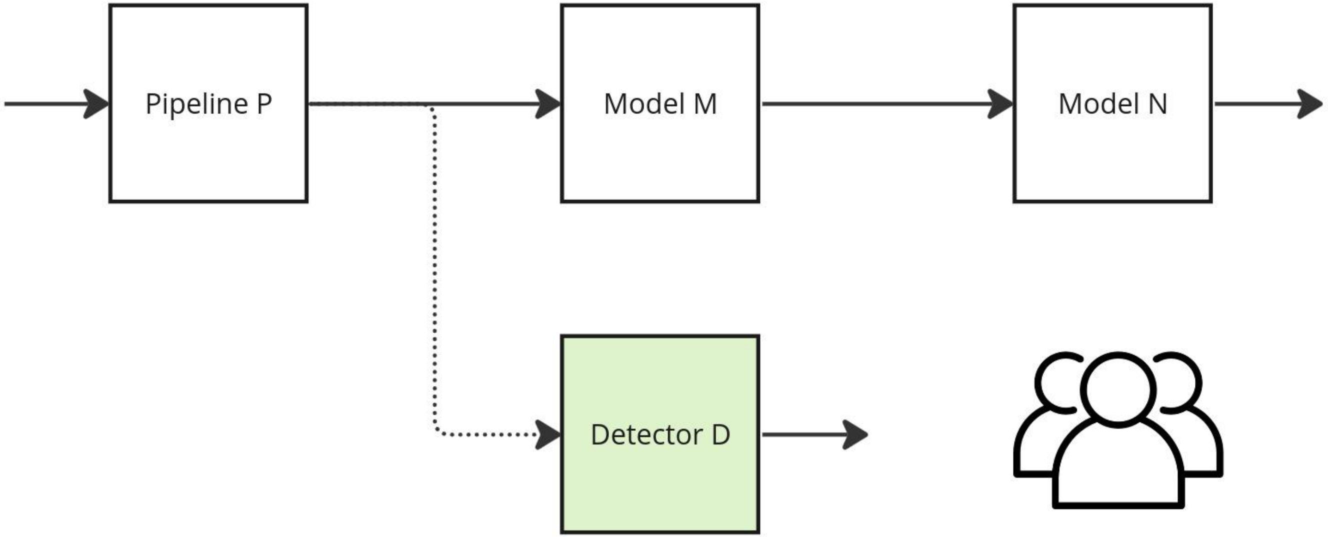
RabbitMQ

Flink

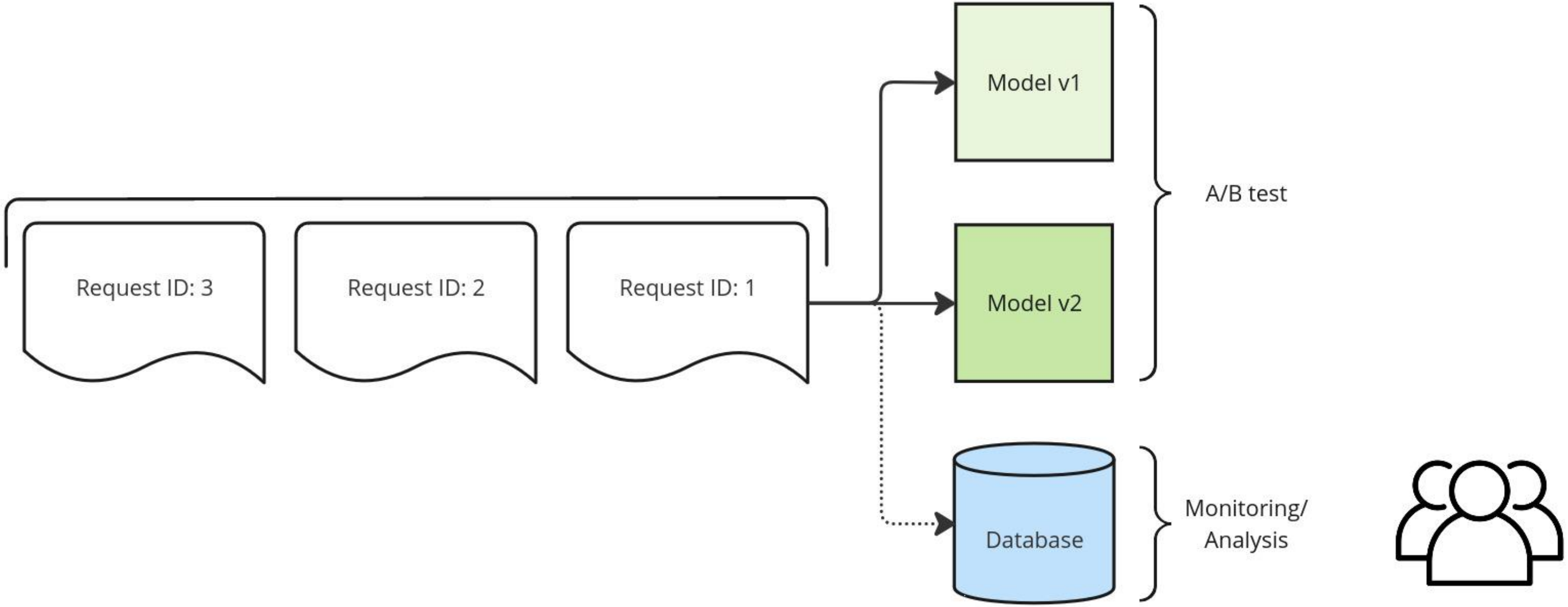
Pub/Sub – Multiplexing Pipelines



Pub/Sub – Extending Pipelines



Pub/Sub – Repeatable Streams



Tech Stack — Choices

```
fun <T> KStream<T, TRecord>.filterForPipeline(pipelineName: String): KStream<T, TRecord> {
    return this
        .transformValues(ValueTransformerSupplier { PipelineNameFilter(pipelineName) })
        .filterNot { _, value -> value == null }
}
```

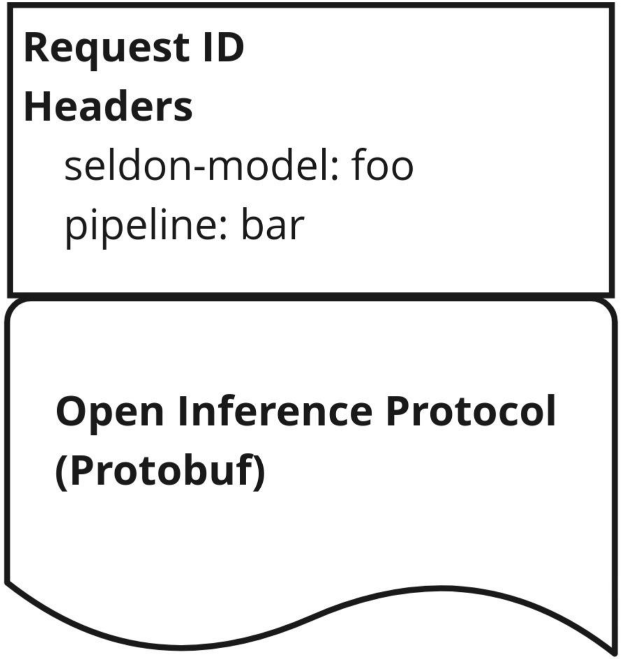
```
private fun buildInputInputStream(builder: StreamsBuilder) {
    val s1 = builder
        .stream(inputTopic.topicName, consumerSerde)
        .filterForPipeline(inputTopic.pipelineName)
        .unmarshallInferenceV2Request()
        .filterRequests(inputTopic.pipelineName, inputTopic.topicName, tensors, tensorRenaming)
        // handle cases where there are no tensors we want
        .filter { _, value -> value.inputsList.size != 0 }
        .batchMessages(batchProperties)
        .marshallInferenceV2Request()
```

Anatomy of a Topic Name

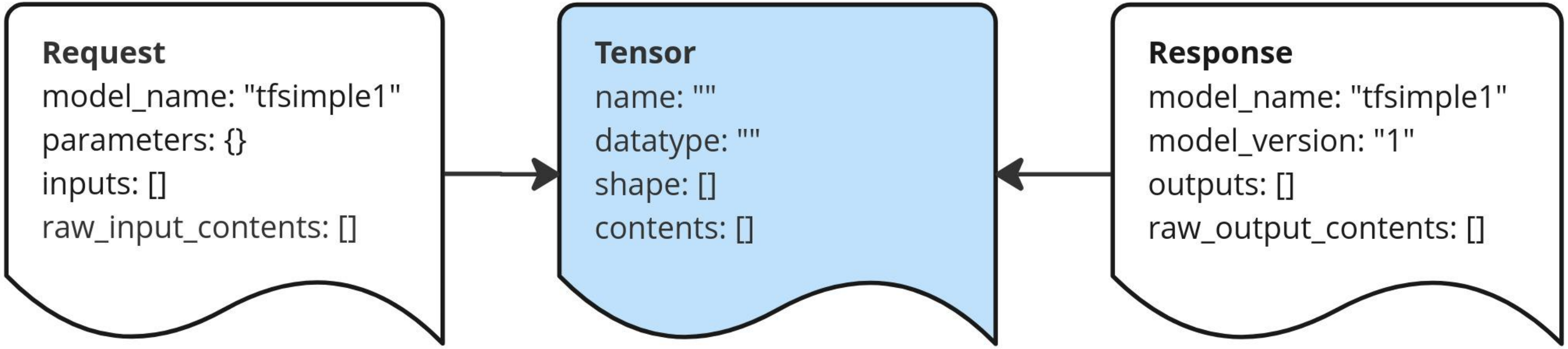
```
<prefix>.<namespace>.<pipeline|model>.<name>.<inputs|outputs>
    seldon.default.pipeline.foo.inputs
    seldon.recommendations.model.bar.outputs

<prefix>.<namespace>.errors.errors
    ml.default.errors.errors
```

Anatomy of an Inference Message



Open Inference Protocol



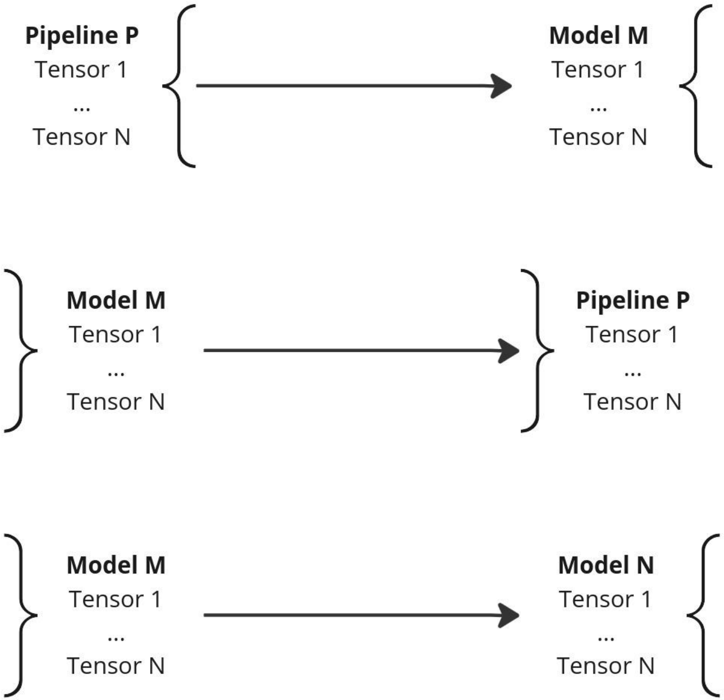
Open Inference Protocol – Batching

Tensor

shape: [BATCH_SIZE, ...]

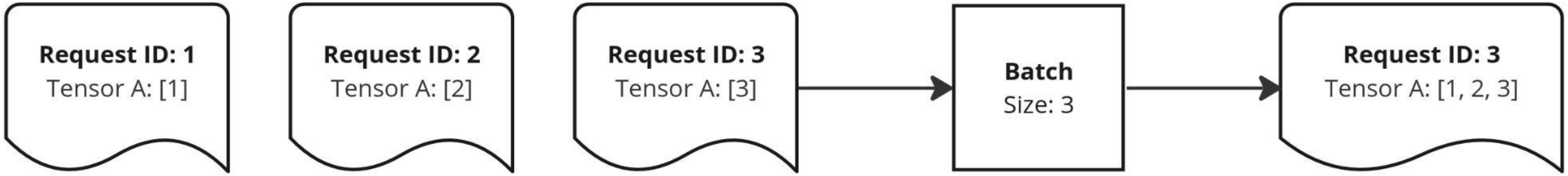
...

Dataflow Operations – Topic Chaining

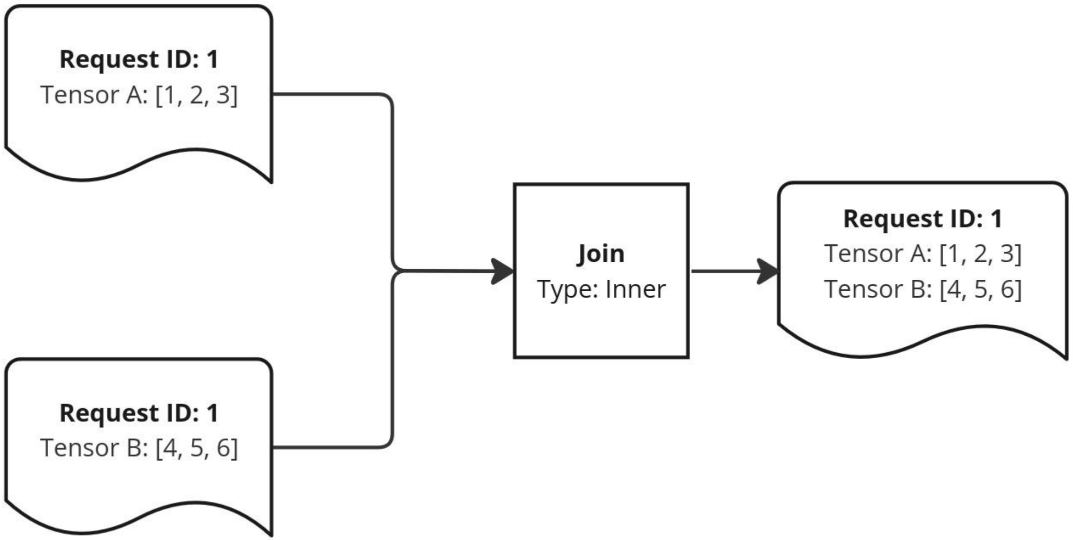


Dataflow Operations – Tensor Projection

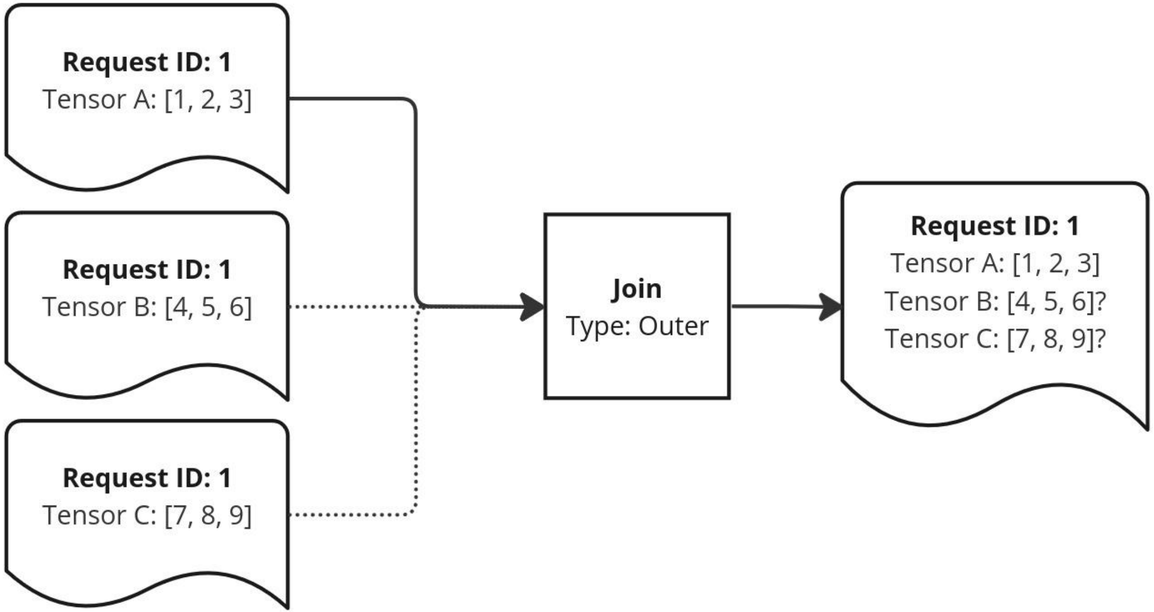
Dataflow Operations – Batching



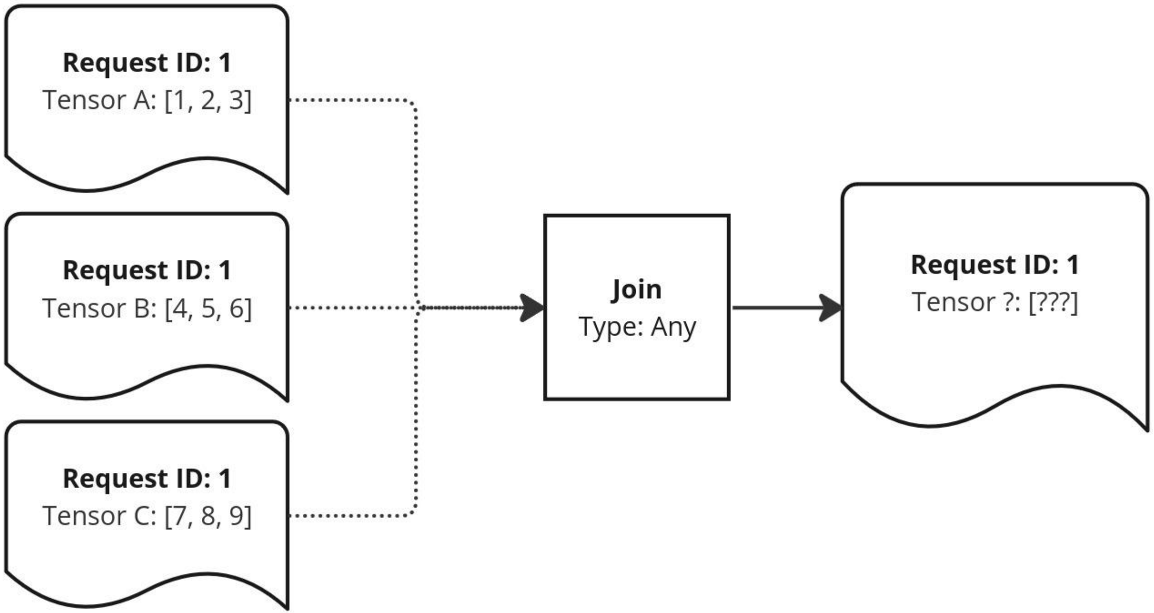
Dataflow Operations – Inner Join



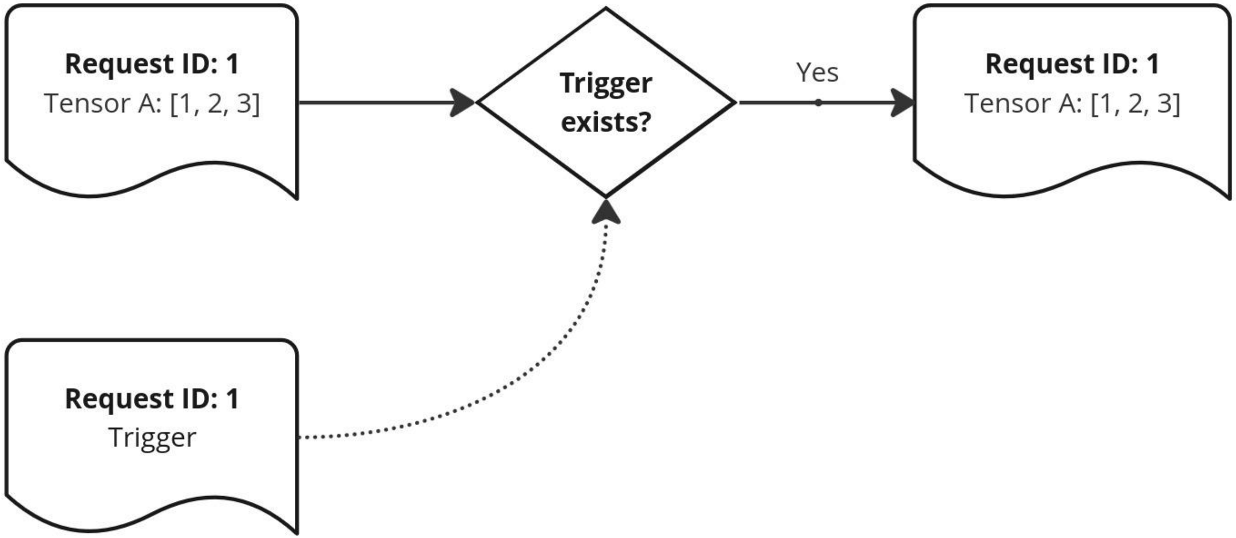
Dataflow Operations – Outer Join



Dataflow Operations – Any Join



Dataflow Operations – Triggers



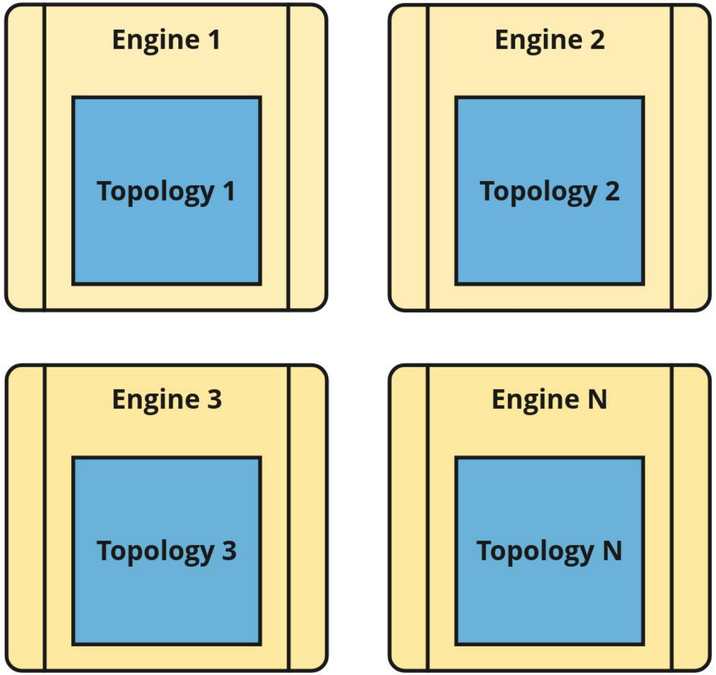
Topologies

Core v2 Pipeline = KStream Topology

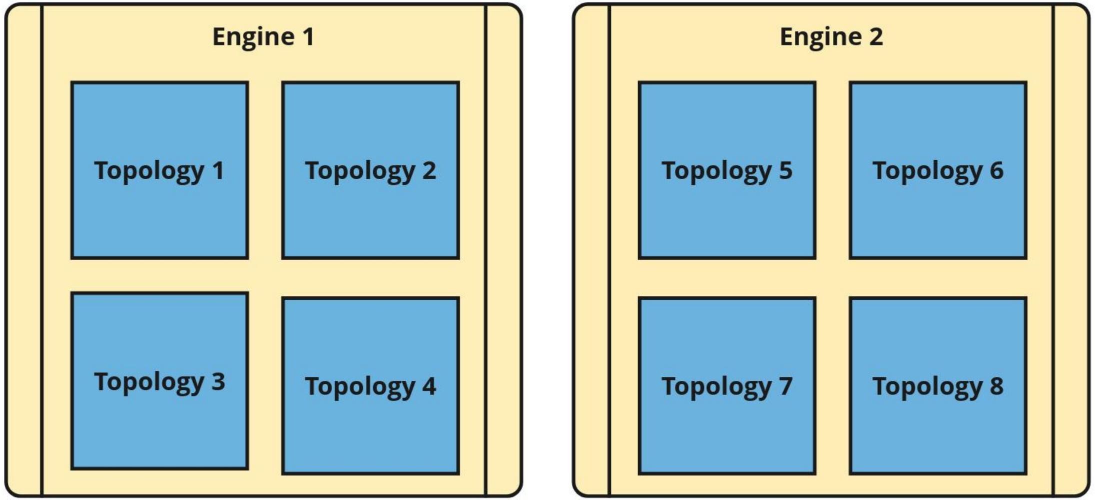
```yaml
apiVersion: mlops.seldon.io/v1alpha1
kind: Pipeline
metadata:
 name: road-counter
spec:
 steps:
   - name: vectorize
   - name: faulty_image_filter
     inputs:
     - vectorize.outputs
   - name: object_detection
     inputs:
     - vectorize.outputs
     - faulty_image_filter.outputs
...
```

```kotlin
val builder = StreamsBuilder()
builder
       .stream(inputTopic.topicName, consumerSerde)
       .filterForPipeline(inputTopic.pipelineName)
...
val topology = builder.build()
```

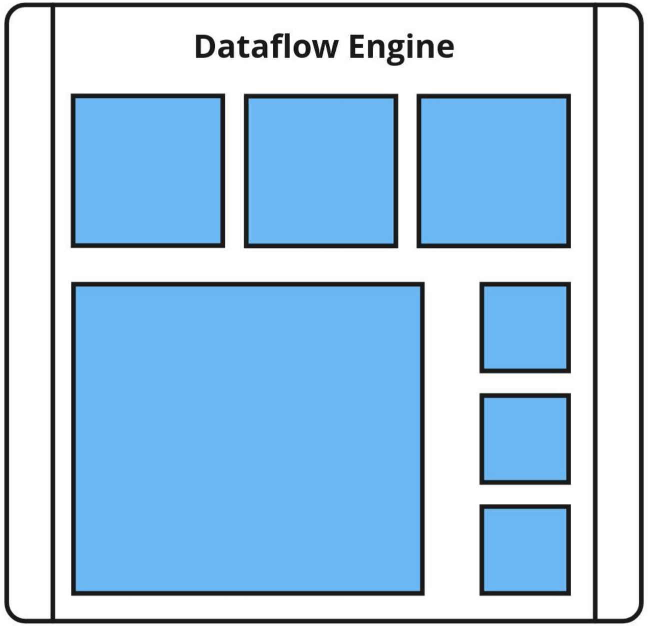
Pipeline Allocation – 1 Topology = 1 Engine



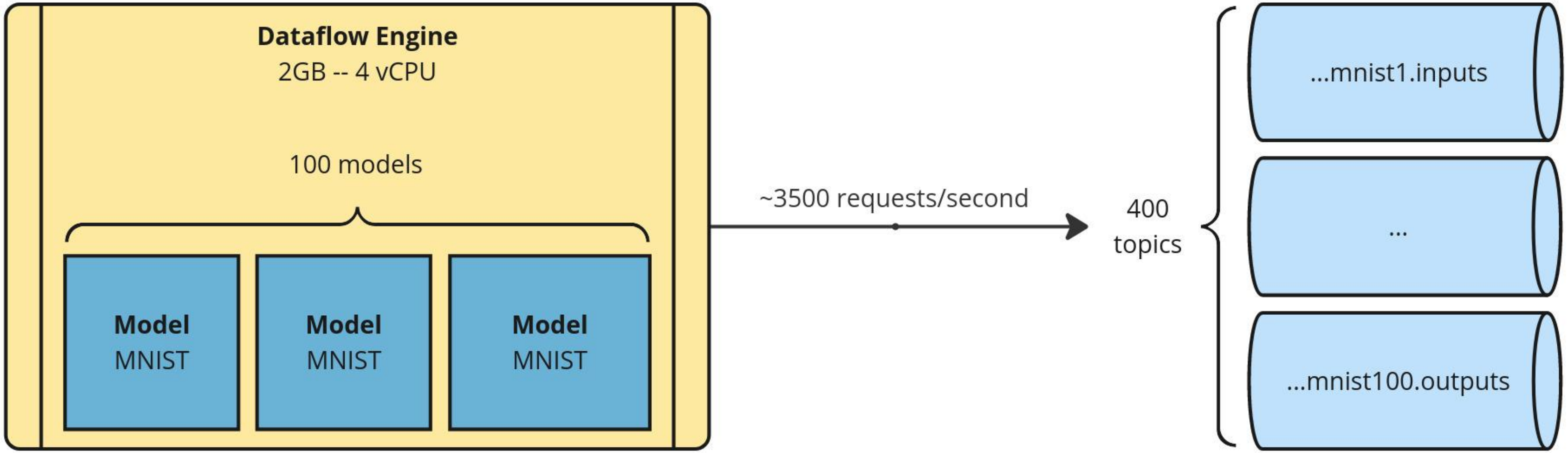
Pipeline Allocation – Shared Engines



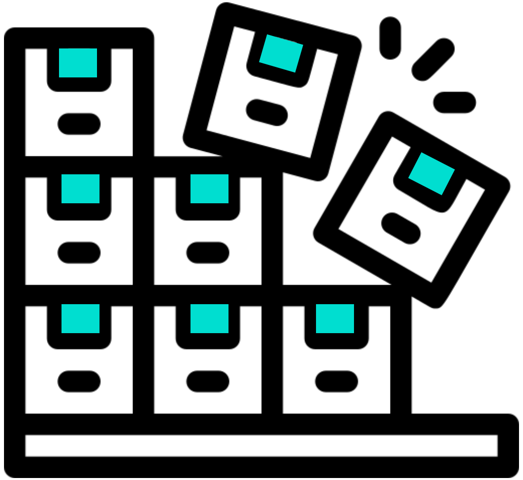
Pipeline Allocation — Threads



How Many Is Too Many?



How Many Is Too Many?

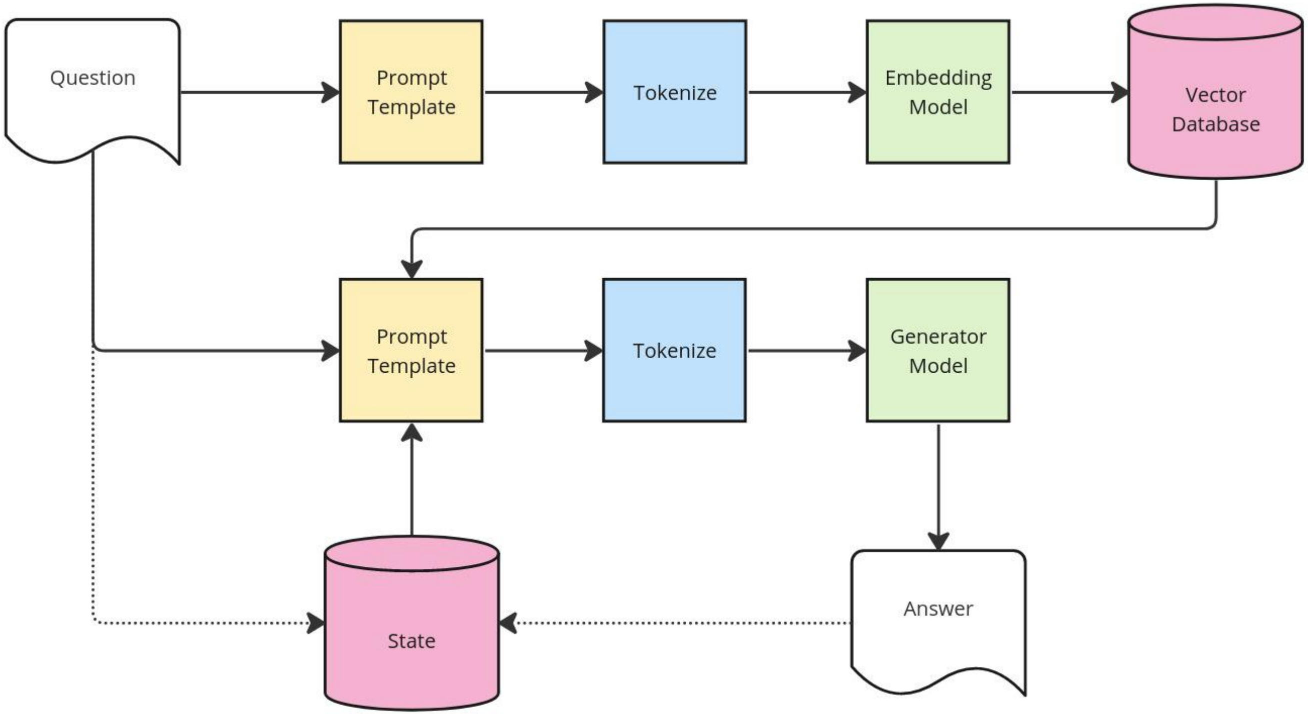


[KSQL Limitations in Confluent Cloud]

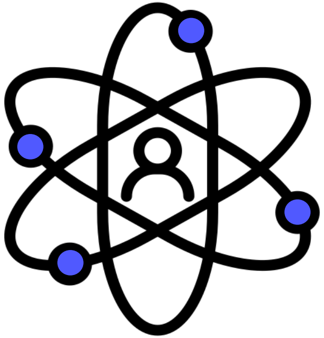
[Kafka Summit 2022 - Modular Topologies]

[KIP-809 (Modular Topologies)]

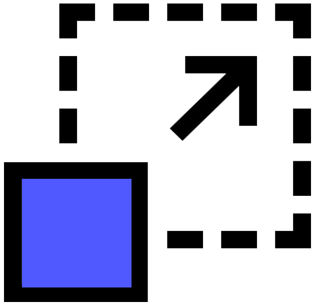
Advanced ML Applications



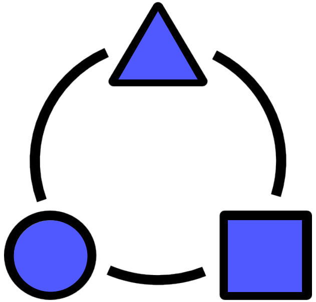
Challenges



Dynamism

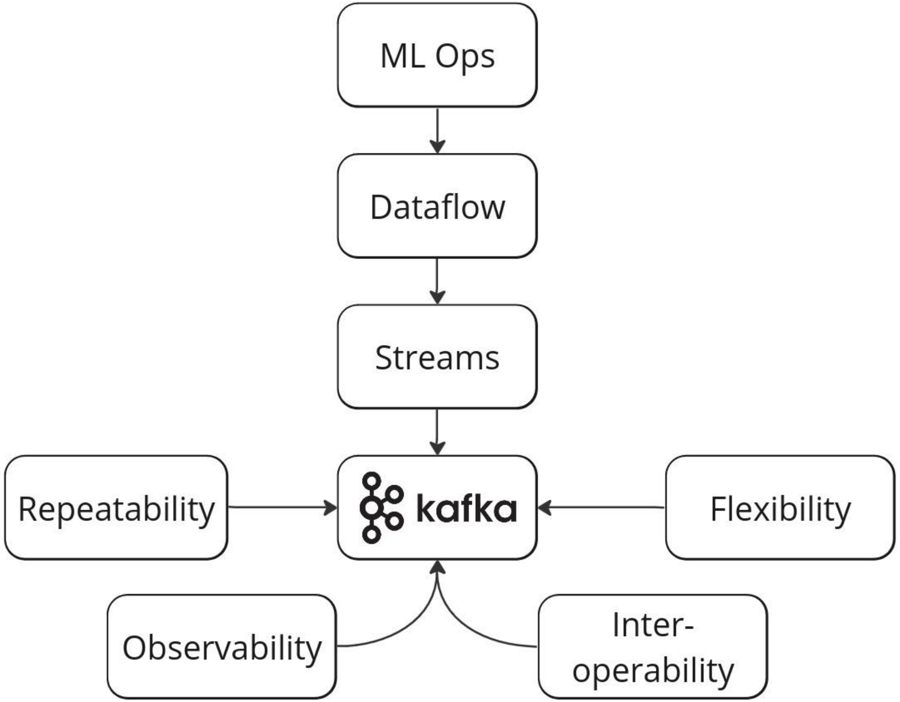


Scalability



Variety

Thanks for listening!



The Seldon Core v2 Team

Core v2 on GitHub

SCAN ME

Team

Clive Cox
CTO

Sherif Akoush
MLOps Engineer

Rafal Skolasinski
Machine Learning Engineer

Adrian Gonzalez-Martin
Machine Learning Engineer

Alex Rakowski
Software Engineer

Andrei Paleyes
MLOps Researcher

Contact us: hello@seldon.io