ServiceX

A Distributed, Caching, Columnar Data Delivery Service

B. Galewsky[A], R. Gardner[B], L. Gray[E], M. Neubauer[A],
J. Pivarski[C], M. Proffitt[D], I. Vukotic[B], G. Watts[D], M. Weinberg[B]

A: University of Illinois at Urbana-Champaign, B: The University of Chicago,
C: Princeton University, D: University of Washington, E: Fermilab

ILLINOIS
NCSA | National Center for Supercomputing Applications

iris hep

Problems

- Expensive to transmit intact Root files over WAN
- New generation of grad students don’t want to learn non-transferable software tools
- Grid jobs can require babysitting to complete
- Repeated queries can result in repeated expense
- Too much craft required to extract features for ML Training

ServiceX Features

- Orchestrates containerised experiment approved frameworks to produce columnar outputs.
- REST Server
  - Can be Co-located With Data Lake
  - Easily deployed to local Tier3 cluster
- Simple Select Statements for extracting data
- Stream Awkward Arrays to Analysis Code
  - Use Kafka for streaming
  - Cache results for instant replay
- Save HDF5 or Flat Root Files to Object Store
  - Ntuples as a Service
- Completely Transactional

Sample Transform Request

```
{
"did": "mc15_13TeV:mc15_13TeV.361106.PowhegPythia8EvtGen_AZNLOCTEQ6L1_Zee.merge",
"columns": "Electrons.pt(), Electrons.eta(), Muons.eta(), Muons.phi(), Muons.e()",
"image": "sslhep/servicex-transformer:v0.1",
"result-destination": "kafka",
"kafka":{
        "broker": "servicex-kafka-1.slateci.net:19092"
},
"chunk-size": 9000,
"workers": 17
}
```

Sample Transform Request

```
{
"did": "mc15_13TeV:mc15_13TeV.361106.PowhegPythia8EvtGen_AZNLOCTEQ6L1_Zee.merge",
"columns": "Electrons.pt(), Electrons.eta(), Muons.eta(), Muons.phi(), Muons.e()",
"image": "s[illegible]/servicex-transformer:v0.1",
"result-destination": "[illegible]afka",
"kafka":{
        "b[illegible]                                    19092"
},
"chunk-size": 9000,
"workers": 17
}
```

Rucio Dataset Identifier

Sample Transform Request

```
{
"did": "mc15_13TeV:mc15_13TeV.361106.PowhegPythia8EvtGen_AZNLOCTEQ6L1_Zee.merge",
"columns": "Electrons.pt(), Electrons.eta(), Muons.eta(), Muons.phi(), Muons.e()",
"image": "[illegible]lhep/servicex-transformer:v0.1",
"result-destina[illegible]": "kafka",
"kafka":{
        "brok[illegible]                                    92"
},
"chunk-size": 9000,
"workers": 17
}
```

List of columns to retrieve

Sample Transform Request

```
{
"did": "mc15_13TeV:mc15_13TeV.361106.PowhegPythia8EvtGen_AZNLOCTEQ6L1_Zee.merge",
"columns": "Electrons.pt(), Electrons.eta(), Muons.eta(), Muons.phi(), Muons.e()",
"image": "sslhep/servicex-transformer:v0.1",
"result-destination": "kafka",
"kafka":{
        "broker": "servicex-kafka-1.slate                9092"
},
"chunk-size": 9000,
"workers": 17
}
```

Versioned Docker Image to Use for Transformation

Sample Transform Request

```
{
"did": "mc15_13TeV:mc15_13TeV.361106.Po                                    e",
"columns": "Electrons.pt(), Electrons.e                                    ()",
"image": "sslhep/servicex-transformer:v0.1",
"result-destination": "kafka",
"kafka":{
        "broker": "servicex-kafka-1.slateci.net:19092"
},
"chunk-size": 9000,
"workers": 17
}
```

Send results to Kafka Topic

Sample Transform Request

```
{
"did": "mc15_13TeV:mc15_13TeV.361106.PowhegPythia8EvtGen_AZNLOCTEQ6L1_Zee.merge",
"columns": "Electrons.pt(), Electrons.eta(), Muons.eta(), Muons.phi(), Muons.e()",
"image": "sslhep/servicex-transformer:v0.1",
"result-destination": "kafka",
"kafka":{
        "broker": "servicex-kafka-[illegible].net:19092"
},
"chunk-size": 9000,
"workers": 17
}
```

Each Result Message Will Contain 9,000 Events

Sample Transform Request

```
{
"did": "mc15_13TeV:mc15_13TeV.361106.PowhegPythia8EvtGen_AZNLOCTEQ6L1_Zee.merge",
"columns": "Electrons.pt(), Electrons.eta(), Muons.eta(), Muons.phi(), Muons.e()",
"image": "sslhep/servi
"result-destination": "
"kafka":{
        "broker": "service        1.slateci.net:19092"
},
"chunk-size": 900
"workers": 17
}
```

Spin Up 17 Workers to Process Dataset

Simple Transform Output

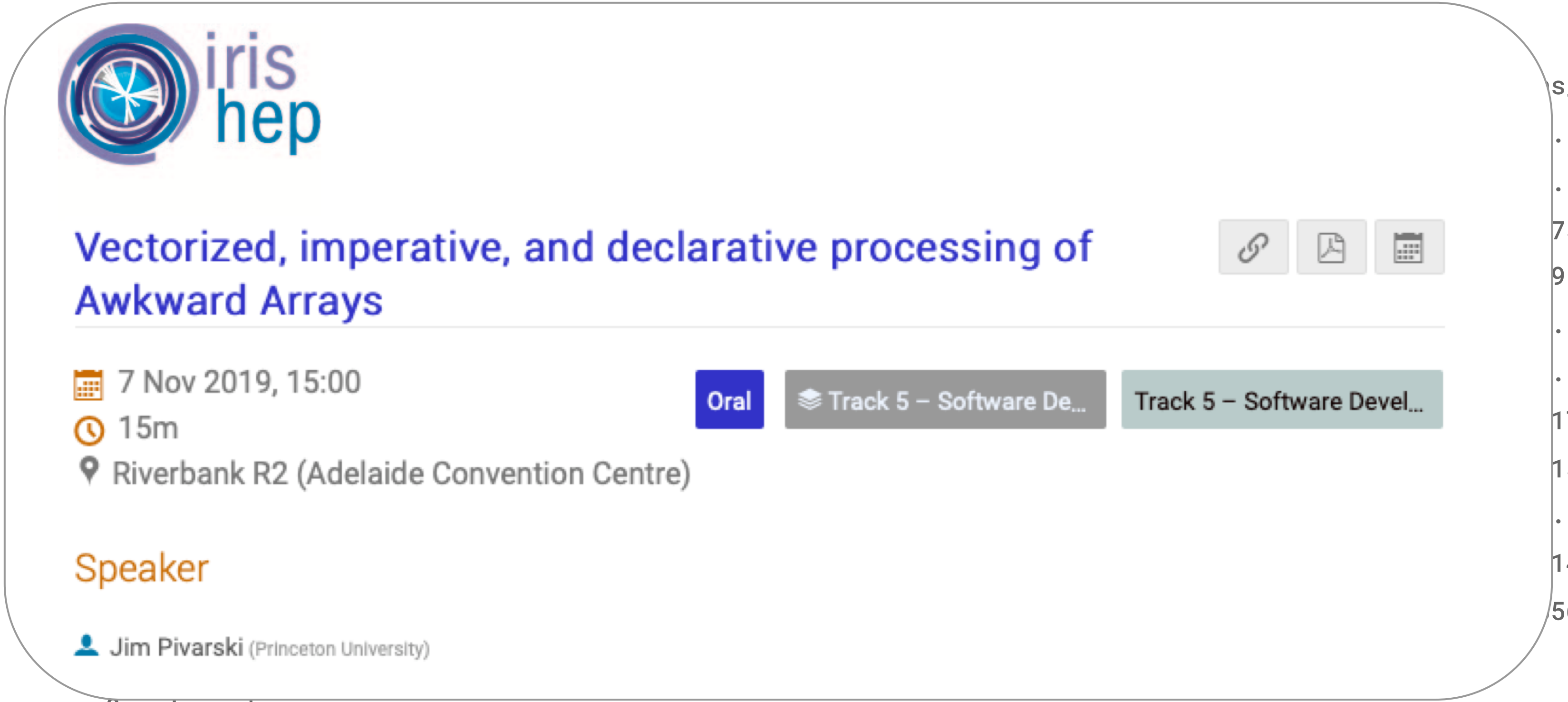
	Electrons_pt	...	Electrons_e
0	[74024.8828125, 30938.857421875, 12550.4033203...	...	[80629.24571240724, 57634.59933983843, 17979.5...
1	[46399.02734375, 44436.625, 4292.2861328125]	...	[131815.2938011208, 142047.43631280548, 23115....
2	[48510.99609375, 43119.9140625]	...	[71525.89772544046, 55259.698605514975]
3	[39687.828125, 4727.27978515625]	...	[78226.37251484161, 6912.516413192391]
4	[47901.21484375, 20389.5703125, 3191.822021484...	...	[92951.5913050441, 43011.55468404851, 3700.502...
...	...	...	...
146995	[48121.5546875, 28690.892578125]	...	[81067.29362621436, 28701.674710654817]
146996	[4180.11767578125]	...	[7611.620870320015]
146997	[23951.232421875, 10593.669921875, 3781.232177...	...	[24488.126431321485, 64412.10937261527, 3798.2...
146998	[42729.1328125]	...	[251197.29821046814]
146999	[41925.36328125, 27975.951171875]	...	[43903.75983279697, 66962.2581786556]

[147000 rows x 8 columns]

Simple Transform Output

```
                                                s_e
0                                               ...
1                                               ...
2                                               75]
3                                               91]
4                                               ...
...                                             ...
146995                                          17]
146996                                          15]
146997                                          ...
146998                                          14]
146999                                          56]

[147000 rows x 8 columns]
```

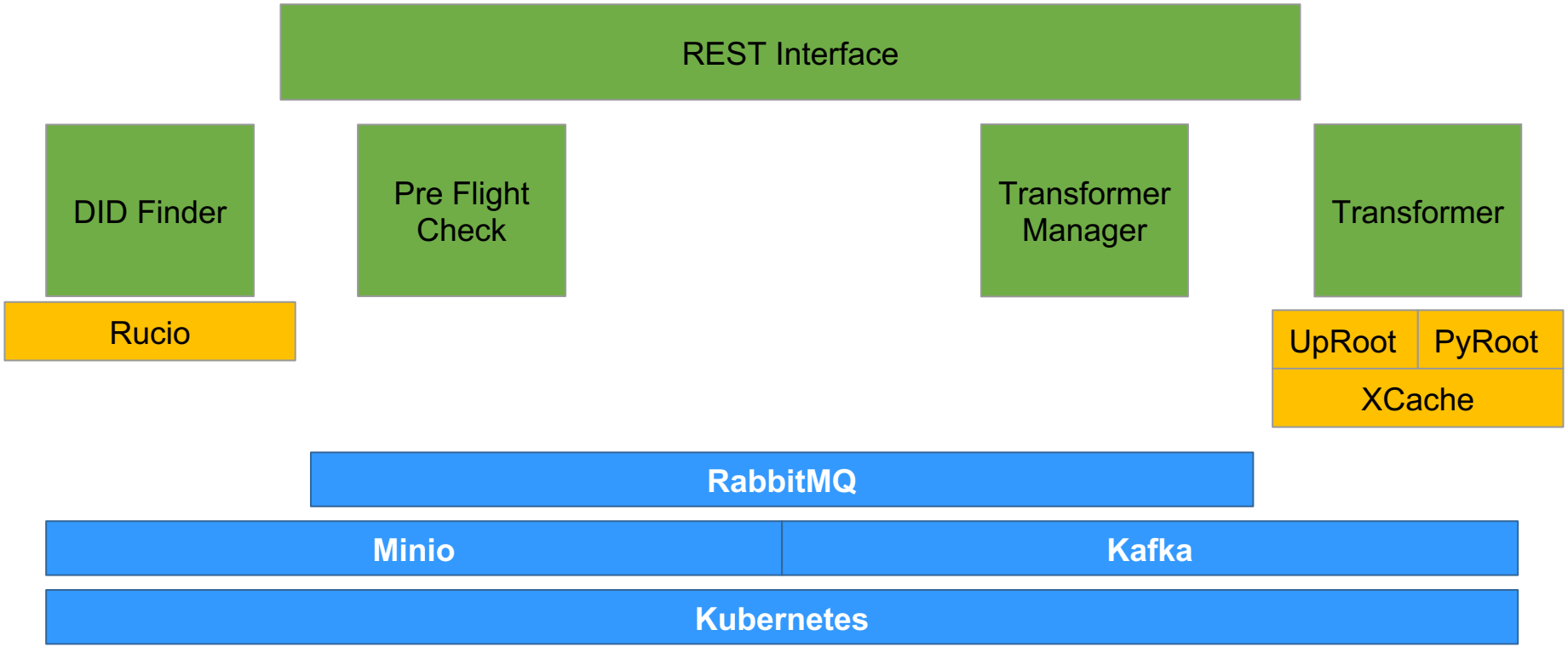


Simple Transform Output

```
                                    s_e
0                                   ...
1                                   ...
2                                   75]
3                                   91]
4                                   ...
...                                 ...
146995                              17]
146996                              15]
146997                              ...
146998                              14]
146999                              56]

[147000 rows x 8 columns]
```

Service Architecture V1



Python Transformers

- Atlas xAOD Transformer
  - Based on Atlas Analysis Base
  - pyROOT for accessing branches
  - Quite Slow
  - No access to calibration libraries
  - No filtering
- Flat Root Transformer
  - Based on upROOT
  - Can read CMS nanoAODs
  - Generated n-tuple files
  - Quite Fast
  - No filtering

Available Today

```
% helm install ssl-hep/servicex
```

https://github.com/ssl-hep/ServiceX

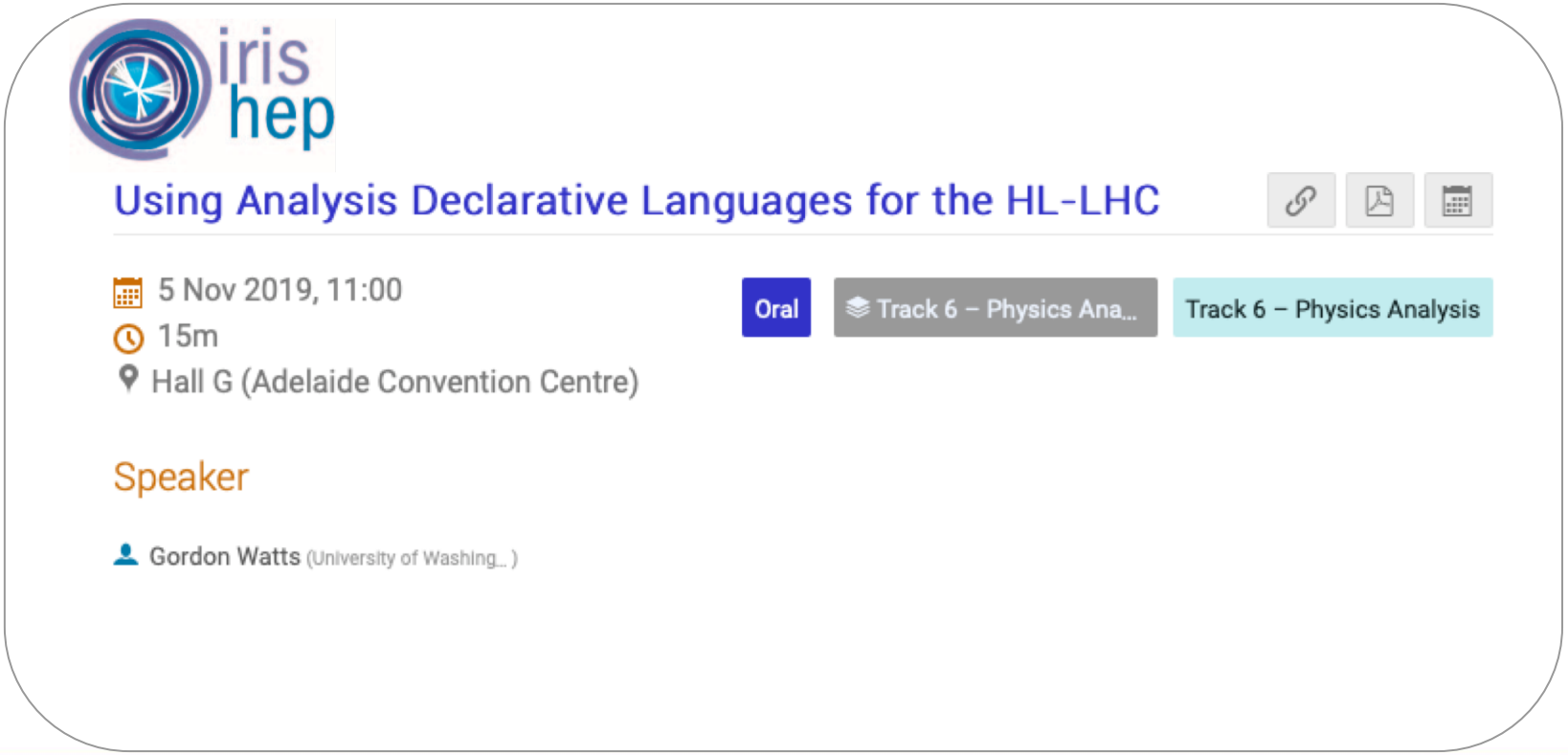
Version 2 - With C++ Transformers

- SQL-Like Select Statements
  - Specify branches to return
  - Filtering
  - Simple Mathematical Operations
  - Experiment Approved Calibrations
- C++ Code Generated from Select Statements
- ATLAS code operates in EventLoop Framework
- Very Fast
- Has access to all Experiment Approved Libraries

Example Select Code

```
xAOD.Where(
    lambda e:
        e.jet_pT.Where(lambda pT:
        pT > 1000).Count() > 0)
    .Select('lambda e:
        [e.eventNumber, e.CalibJet_pT]'
)
```

Example Select Code



Example Select Code

Two Posters in Hall F

A Functional Declarative Analysis Language in Python
Emma Torro Pastor

HEP Data Query Challenges
Mason Proffitt

What Next?

- Try out the ServiceX Demo
- Talk to us about providing data for your Analysis
- Join the Discussion on Designing the Select Language
- Help Build ServiceX!

https://github.com/ssl-hep/ServiceX

bengal1@illinois.edu

https://github.com/ssl-hep/ServiceX

ILLINOIS
NCSA | National Center for Supercomputing Applications

iris hep

Open Source Technologies Used

Flask

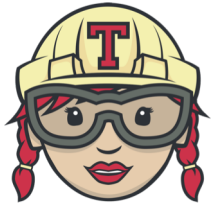
APACHE ARROW

kafka

HELM

kubernetes

docker



RabbitMQ

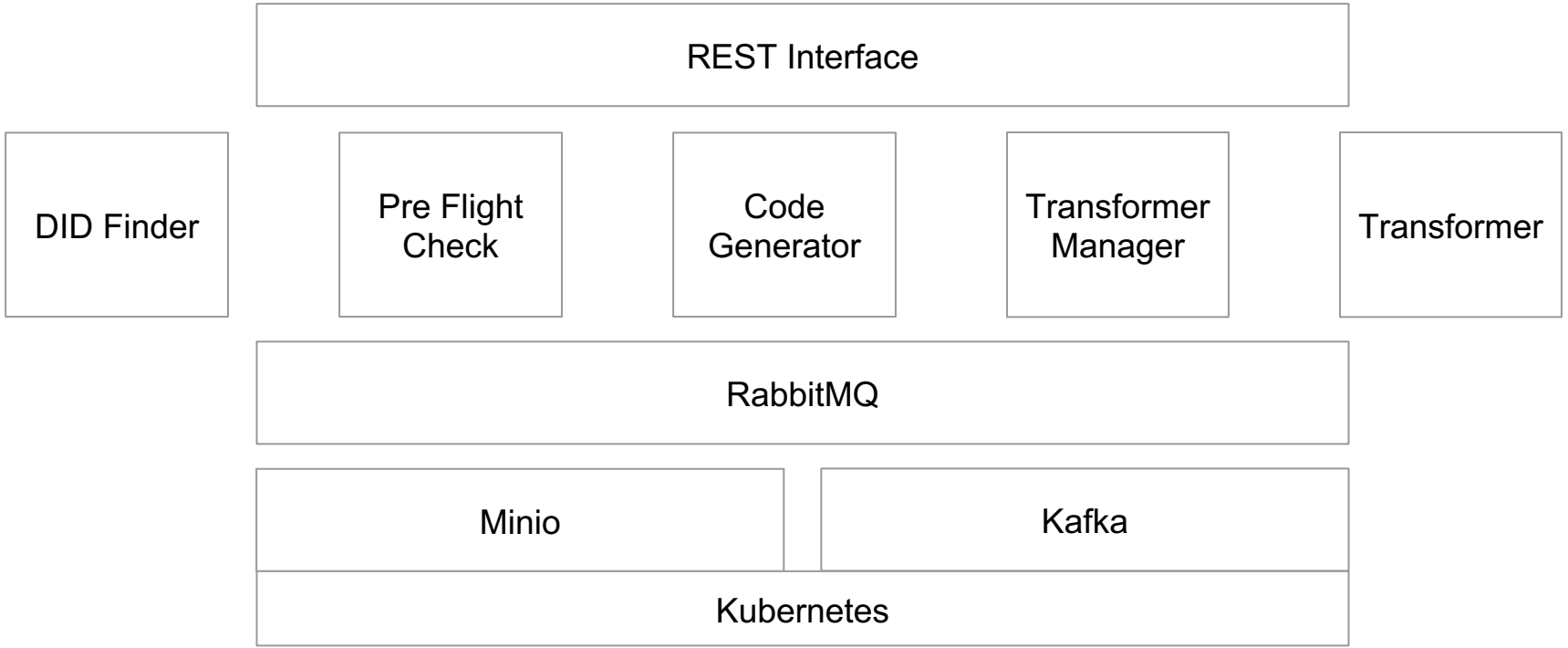
MINIO

SQLite

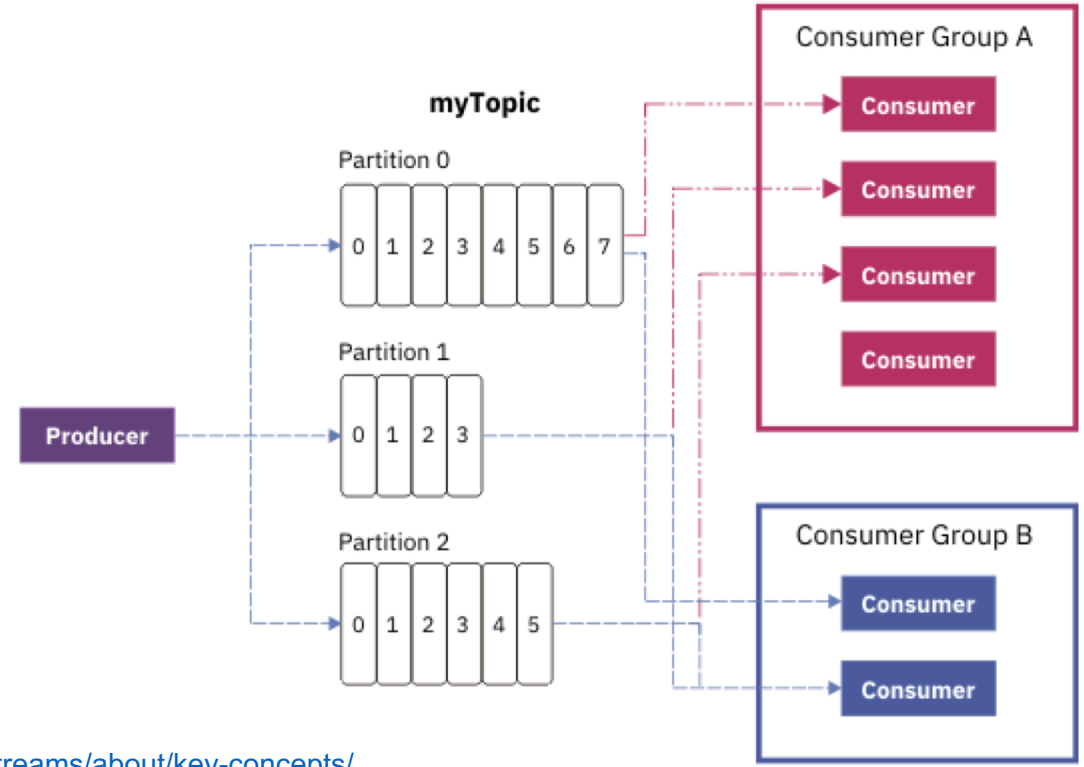
PostgreSQL

pytest

Service Architecture V2: Available in January



Kafka



https://ibm.github.io/event-streams/about/key-concepts/

ServiceX Sequence Diagram