

IDC Re-engineering - Geophysical Monitoring System (GMS) Processing Demonstration

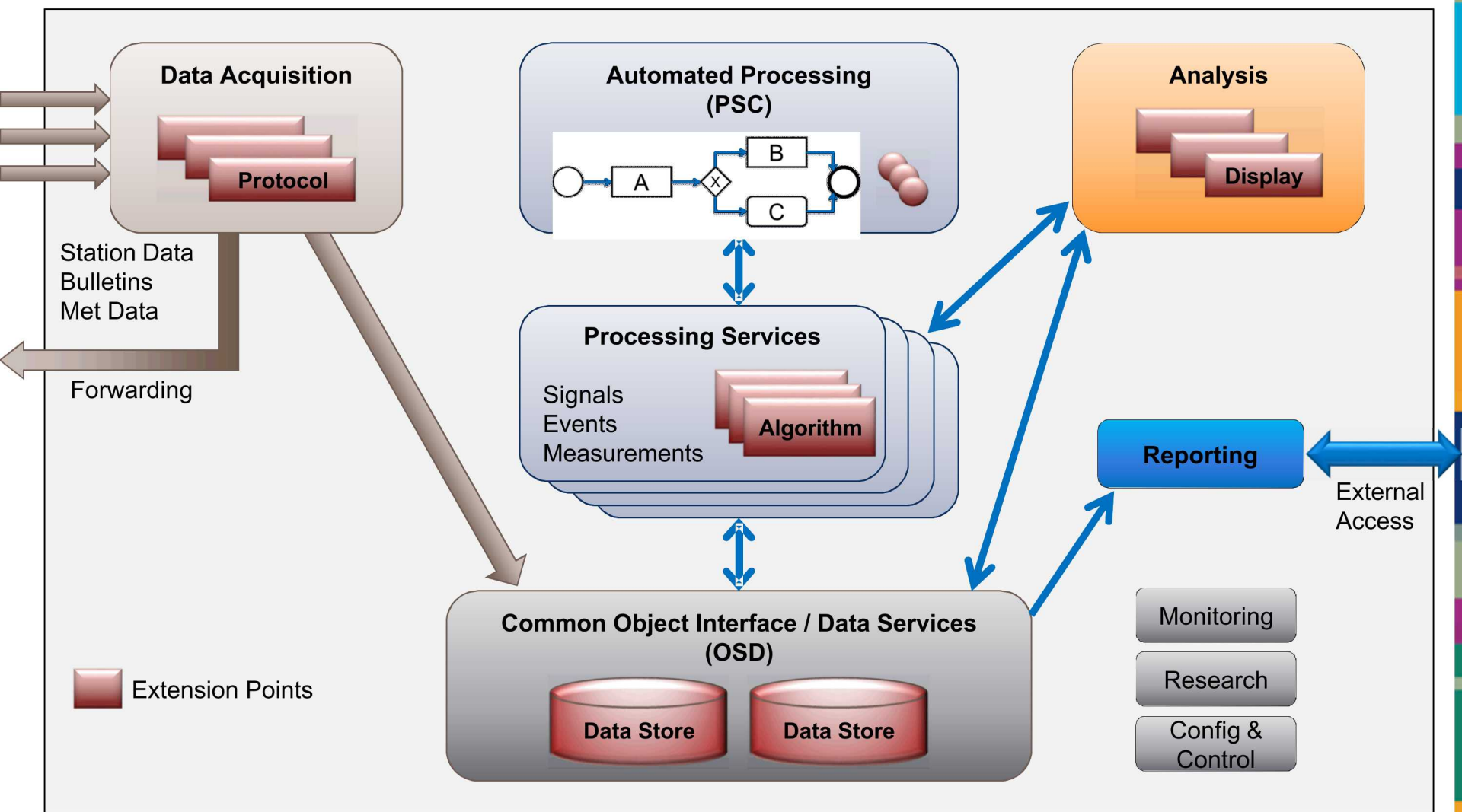


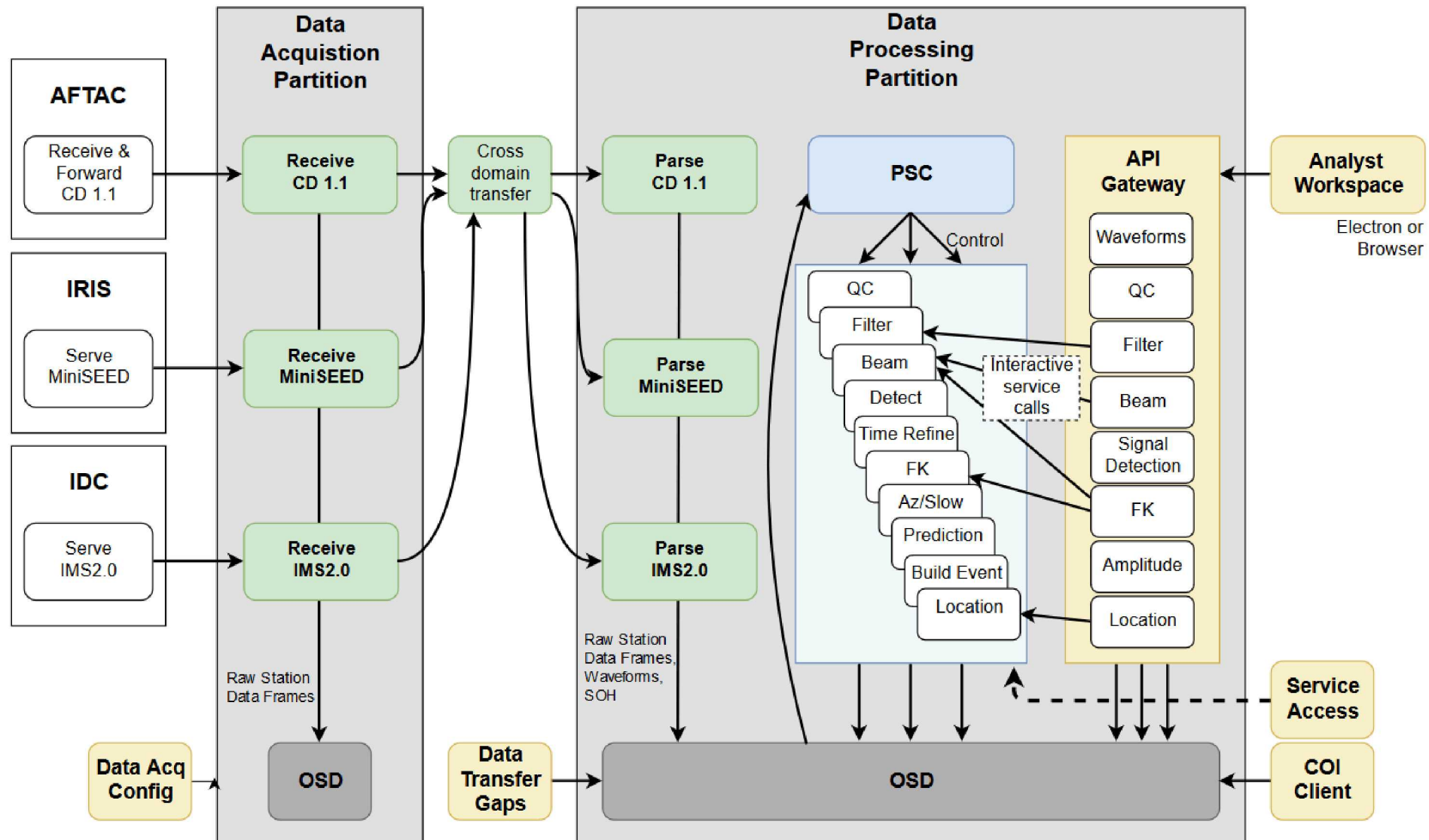
PRESENTED BY

J. Mark Harris

US NDC – IDC Bilateral Meeting
25-26 February 2019

- Processing Sequence Control
- Access to Station Processing Results in the OSD
- Access to Processing Services





Processing Sequence Control



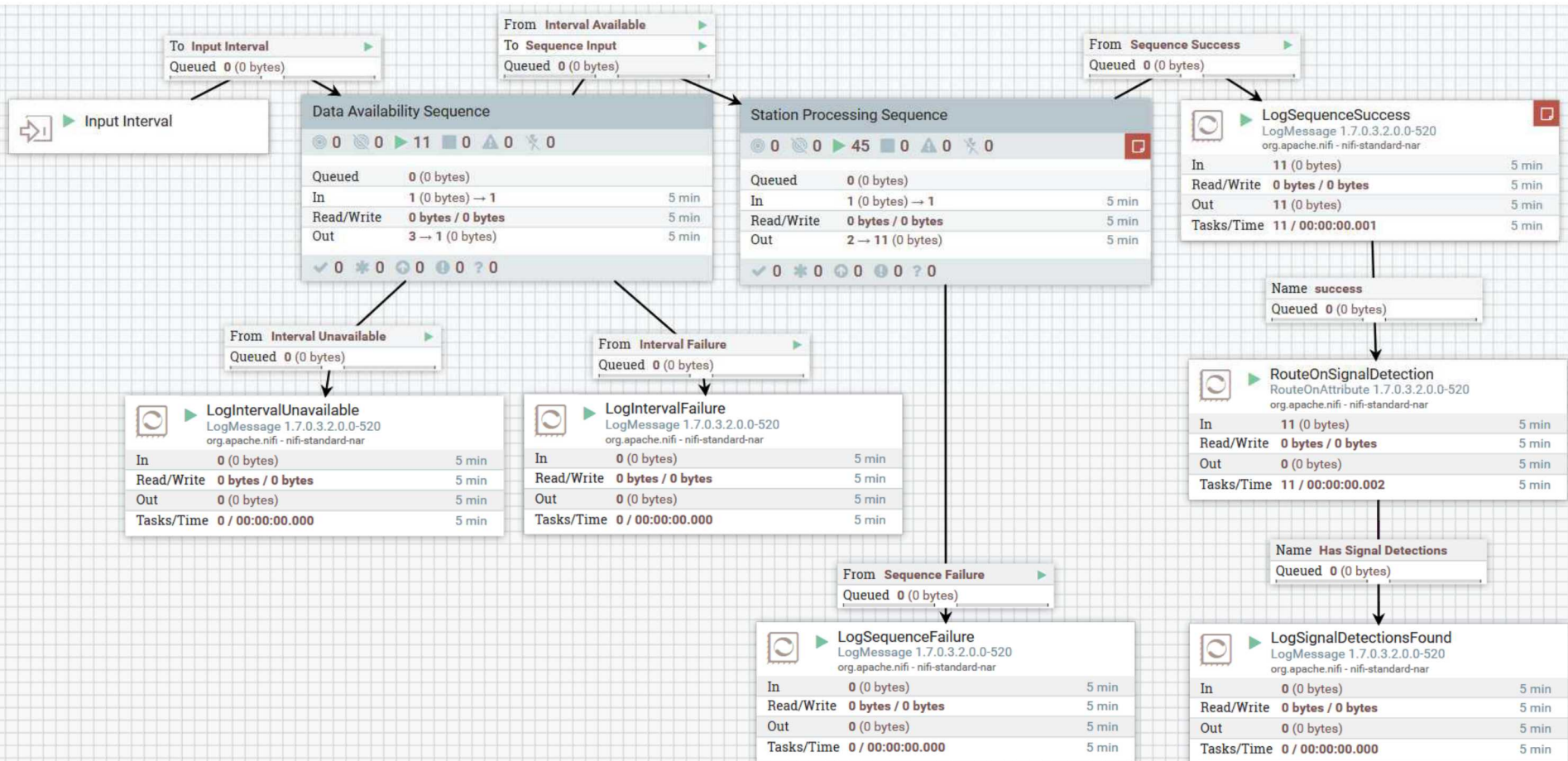
Processing Sequence Control in Apache NiFi

The screenshot displays the Apache NiFi web interface. The top navigation bar shows the URL `dsil.sandia.gov:9101/nifi/?processGroupId=root&componentIds=11d69a02-b443-3521`. The left sidebar contains a 'Navigate' panel and an 'Operate' panel with the message 'Multiple components selected'. The main canvas shows a data flow with an 'Interval Input' component connected to a 'To Input Interval' component, which is then connected to a 'Processing Sequence Control' component. The 'Processing Sequence Control' component's status panel is expanded, showing the following metrics:

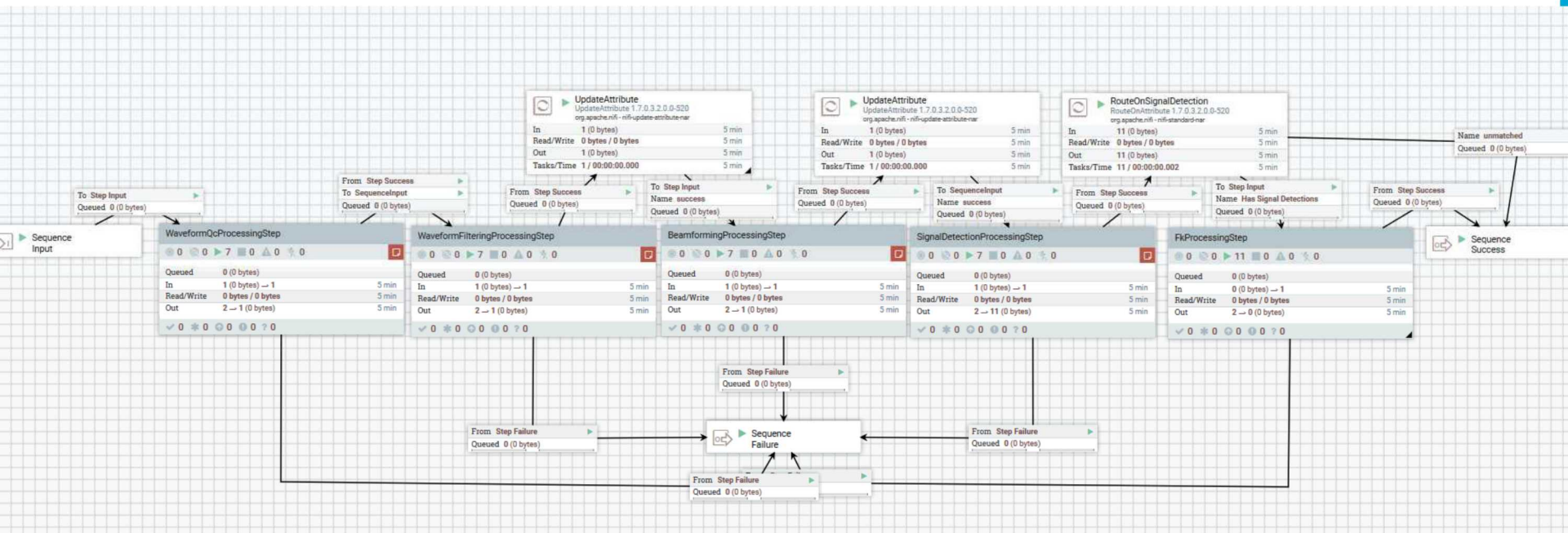
Processing Sequence Control	
Queued	0 (0 bytes)
In	1 (0 bytes) → 1 5 min
Read/Write	0 bytes / 0 bytes 5 min
Out	0 → 0 (0 bytes) 5 min

The status panel also includes a summary row: `✓ 0 * 0 0 0 0 ? 0`.

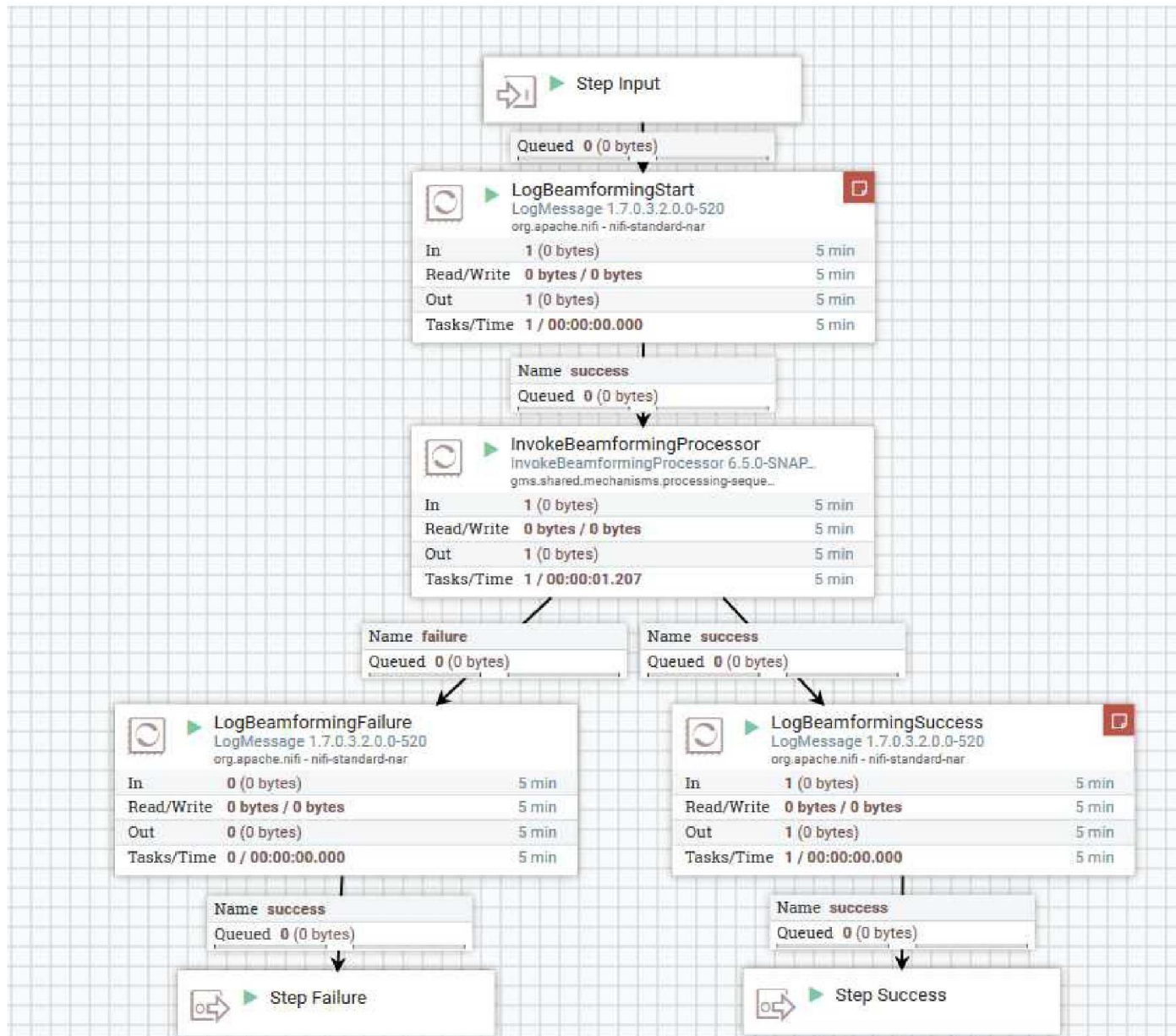
7 Processing Sequence Control



8 Station Processing Sequence



9 Beamforming Processing Step



InvokeBeamformingProcessor Configuration

Step Input

Processor Details

SETTINGS	SCHEDULING	PROPERTIES	COMMENTS
Name InvokeBeamformingProcessor		Automatically Terminate Relationships ? failure	
Id 600e464e-cf31-35ca-b1bb-4dc78d8f178e		Unsuccessful execution of QC Control Service success	
Type InvokeBeamformingProcessor 6.5.0-SNAPSHOT		Successful execution of QC Control Service	
Bundle gms.shared.mechanisms.processing-sequence-controller.pro...			
Penalty Duration ? 30 sec	Yield Duration ? 1 sec		
Bulletin Level ? WARN			

Name success
Queued 0 (0 bytes)

Processor Details

SETTINGS	SCHEDULING	PROPERTIES	COMMENTS
Required field			
Property		Value	
Service URL		http://beam-control-service:8080/signal-enhancement/bea...	

Name success
Queued 0 (0 bytes)

OK

InvokeBeamformingProcessor NiFi Data Provenance

NiFi Data Provenance

Displaying 510 of 510

Oldest event available: 02/20/2019 07:00:14 UTC

Filter

by component name ▼

Date/Time ▼	Type	FlowFile Uuid	Size	Component Name	Component
02/21/2019 06:59:21.094 UTC	ATTRIBUTES_MODIFIED				
02/21/2019 06:59:21.094 UTC	SEND				
02/21/2019 06:54:22.028 UTC	ATTRIBUTES_MODIFIED				
02/21/2019 06:54:22.028 UTC	SEND				
02/21/2019 06:49:21.317 UTC	ATTRIBUTES_MODIFIED				
02/21/2019 06:49:21.317 UTC	SEND				
02/21/2019 06:44:21.028 UTC	ATTRIBUTES_MODIFIED				
02/21/2019 06:44:21.028 UTC	SEND				
02/21/2019 06:39:21.356 UTC	ATTRIBUTES_MODIFIED				
02/21/2019 06:39:21.355 UTC	SEND				
02/21/2019 06:34:21.538 UTC	ATTRIBUTES_MODIFIED				
02/21/2019 06:34:21.538 UTC	SEND				
02/21/2019 06:29:21.413 UTC	ATTRIBUTES_MODIFIED				
02/21/2019 06:29:21.413 UTC	SEND				
02/21/2019 06:24:20.915 UTC	ATTRIBUTES_MODIFIED				
02/21/2019 06:24:20.914 UTC	SEND				
02/21/2019 06:19:20.818 UTC	ATTRIBUTES_MODIFIED				
02/21/2019 06:19:20.818 UTC	SEND				
02/21/2019 06:14:20.759 UTC	ATTRIBUTES_MODIFIED				
02/21/2019 06:14:20.758 UTC	SEND				
02/21/2019 06:09:21.486 UTC	ATTRIBUTES_MODIFIED				
02/21/2019 06:09:21.486 UTC	SEND				
02/21/2019 06:04:20.732 UTC	ATTRIBUTES_MODIFIED				
02/21/2019 06:04:20.732 UTC	SEND	1f5493fc-aa38-40db-8437-ea60042375a0	0 bytes	InvokeBeamformingProcessor	InvokeB
02/21/2019 06:00:21.097 UTC	ATTRIBUTES_MODIFIED	70500045-6ab0-40a4-a66b-264a55af0dad	0 bytes	InvokeBeamformingProcessor	InvokeB

Provenance Event

DETAILS

ATTRIBUTES

CONTENT

Time

02/21/2019 06:54:22.028 UTC

Event Duration

No value set

Lineage Duration

00:00:50.024

Type

ATTRIBUTES_MODIFIED

FlowFile Uuid

ee095cd3-ee6f-4ce7-954c-61ff8a5e6027

File Size

0 bytes

Component Id

600e464e-cf31-35ca-b1bb-4dc78d8f178e

Component Name

InvokeBeamformingProcessor

Component Type

Parent FlowFiles (0)

No parents

Child FlowFiles (0)

No children

OK

Access to Station Processing Results in the OSD



- OSD Services
 - Expose http endpoints to store and retrieve data
 - Station Reference COI Service
 - Waveforms COI Service
 - Signal Detection COI Service
 - etc.
- Master COI Data Client
 - Python-based
 - Allow researchers to access Common Object Interface (COI) objects within the system via a command line interface (CLI) that accepts input arguments for data queries
 - Connects to COI web service interfaces of the OSD (which are returned as JSON objects)
 - Invokes GET endpoints of the Station Reference COI Service, the Waveforms COI Service, and the Signal Detection COI Service, and POST endpoints of Signal Detection COI Service
 - Prints and/or stores the parsed JSON object into a Python COI object
 - Writes associated CSS flat files and performs some limited visualization
 - Meant to be extensible by Researchers

Example: Retrieve MK01 Waveforms

Use Python CLI to find recently acquired MK01 Channel Segments

```
In [10]: %%bash -s "$MASTERCOI" "$time1HourAgo" "$timeNow" "$mk01_shz" "$waveformRepo"
python $1/coidataclient.py -mode waveforms -start_time $2 -end_time $3 -id $4 -hostname $5 -plot True
```

Use the COI Client



The OSD Request

```
GET: Requesting http://osd-waveforms-repository-service:8080/mechanisms/object-storage-distribution/waveforms/channel-segment?channel-id=3f174558-cff8-3aba-bea2-39f9b2b28b30&start-time=2019-02-21T06%3A38%3A20Z&end-time=2019-02-21T07%3A38%3A20Z&with-waveforms=true
st:2019-02-21T06:38:20Z et:None freq:25.0ms
Figure(800x550)
```

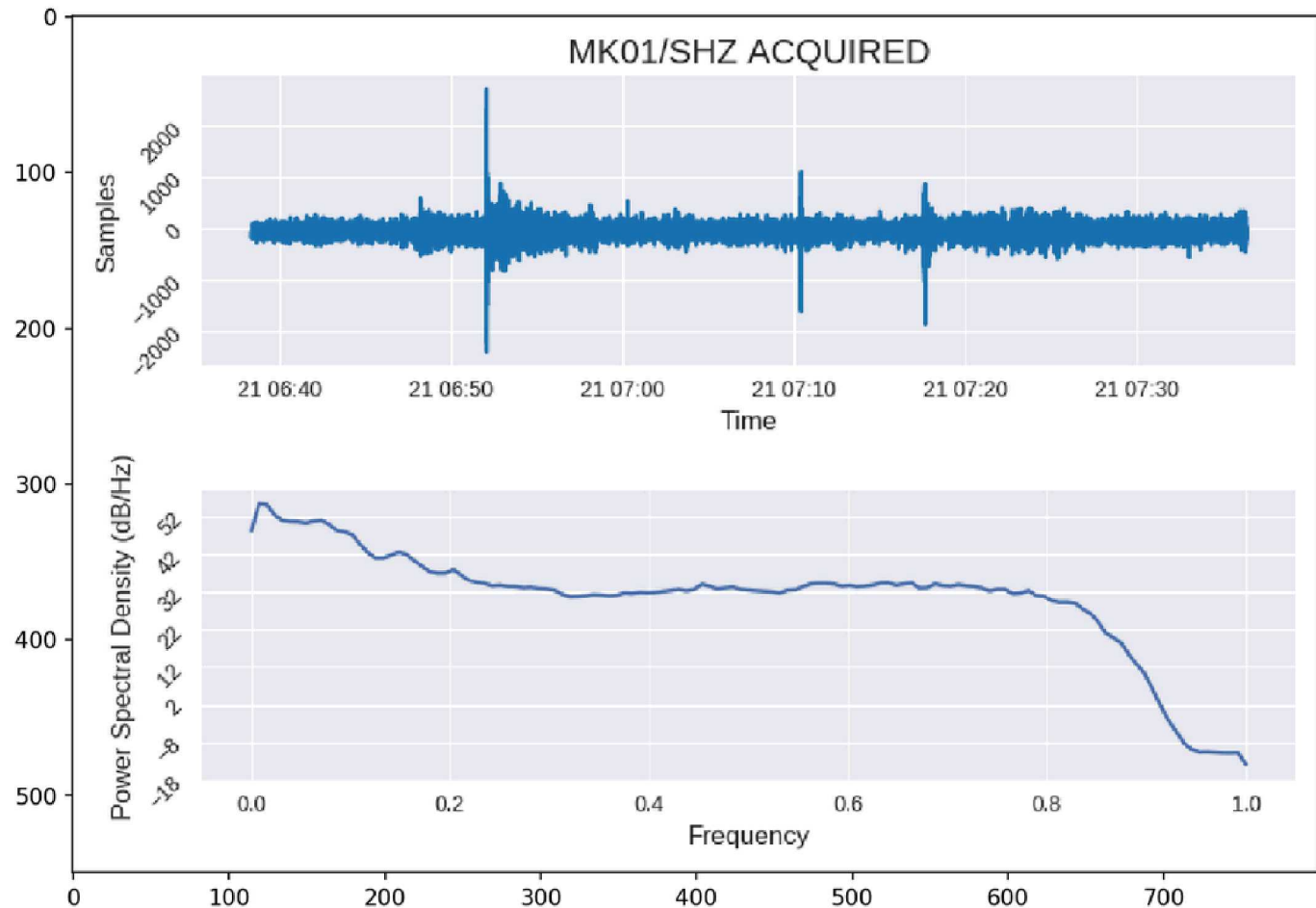
Summary of Results

```

waveforms
channelId: 3f174558-cff8-3aba-bea2-39f9b2b28b30
creationInfo.creationTime: 2019-02-21T07:38:28.807530Z
creationInfo.creatorName: waveforms-repository
creationInfo.softwareInfo.name: waveforms-repository
creationInfo.softwareInfo.version: 0.0.1
endTime: 2019-02-21T07:36:19.975Z
id: 56a41b61-1e54-4c65-bc37-a63e67466516
name: MK01/SHZ ACQUIRED
startTime: 2019-02-21T06:38:20Z
timeseries: Data was too long to display.
timeseriesType: WAVEFORM
type: ACQUIRED
```

Plot MK01 acquired waveform and psd

```
In [12]: #show the plot  
img = mpimg.imread('waveforms.png')  
plt.figure(figsize=(10, 12),dpi=150)  
imgplt = plt.imshow(img)  
plt.show()
```



Example: Retrieve Signal Detections

```
In [13]: %%bash -s "$MASTERCOI" "$time1HourAgo" "$timeNow" "$signalRepo"
python $1/coidataclient.py -mode signal_detections -start_time $2 -end_time $3 -hostname $4
```



```
GET: Requesting http://osd-signal-detection-repository-service:8080/mechanisms/object-storage-distribution/signal-detection/signal-detections?start-time=2019-02-21T06%3A38%3A20Z&end-time=2019-02-21T07%3A38%3A20Z
```

```
__ __ __ __ __
signal_detections
creationInfoId: 00000000-0000-0000-0000-000000000000
id: e91c2c6a-7f93-4726-9ea5-843d057dec26
monitoringOrganization: Organization
stationId: fcb09e96-003a-4e88-8e13-4ba9f2fc993f
__ __ __ __ __
signal_detections
creationInfoId: 00000000-0000-0000-0000-000000000000
id: d76d7625-b30d-43a5-8a16-24885b819bdd
monitoringOrganization: Organization
stationId: 4bbf9e76-ca75-4540-8a70-6dd0bdb25a1a
__ __ __ __ __
signal_detections
creationInfoId: 00000000-0000-0000-0000-000000000000
id: 381282b3-4862-4ad4-a824-1c283d1df809
monitoringOrganization: Organization
stationId: 6b160154-8b59-4820-84a5-f196b72ffe6f
__ __ __ __ __
signal_detections
creationInfoId: 00000000-0000-0000-0000-000000000000
```

Show Signal Detection Hypotheses

```
In [14]: #Use a Signal Detection ID from the results of the previous cell's query
signal_detection_id = "e91c2c6a-7f93-4726-9ea5-843d057dec26"
```

```
In [15]: %%bash -s "$MASTERC0I" "$signal_detection_id" "$signalRepo"
python $1/coidataclient.py -mode signal_detection_hypotheses -id $2 -hostname $3
```



```
GET: Requesting http://osd-signal-detection-repository-service:8080/mechanisms/object-storage-distribution/si
gnal-detection/signal-detections/e91c2c6a-7f93-4726-9ea5-843d057dec26
```

```
_____ signal_detection_hypotheses _____
creationInfoId: 00000000-0000-0000-0000-000000000000
featureMeasurements:
{
  "id": "e2a8dcc2-5fd0-477e-991d-81ca8f9d88e8",
  "channelSegmentId": "a6e9decd-d62a-4bfa-86a4-86edb65362b8",
  "measurementValue": {
    "value": "UNKNOWN",
    "confidence": 1.0
  },
  "featureMeasurementType": "PHASE"
},
{
  "id": "a57b9c9f-6012-4991-909c-0fe2ab602d31",
  "channelSegmentId": "a6e9decd-d62a-4bfa-86a4-86edb65362b8",
  "measurementValue": {
    "value": "2019-02-21T06:52:02.975Z",
    "standardDeviation": "PT0.177590922S"
  },
  "featureMeasurementType": "ARRIVAL_TIME"
}
id: 870b173d-739b-49aa-a2e9-2ef8c10d64c2
parentSignalDetectionId: e91c2c6a-7f93-4726-9ea5-843d057dec26
rejected: False
_____ signal_detection_hypotheses _____
```

Arrival Time of First
Signal Detection
Hypothesis



Show Signal Detection Hypotheses (2)

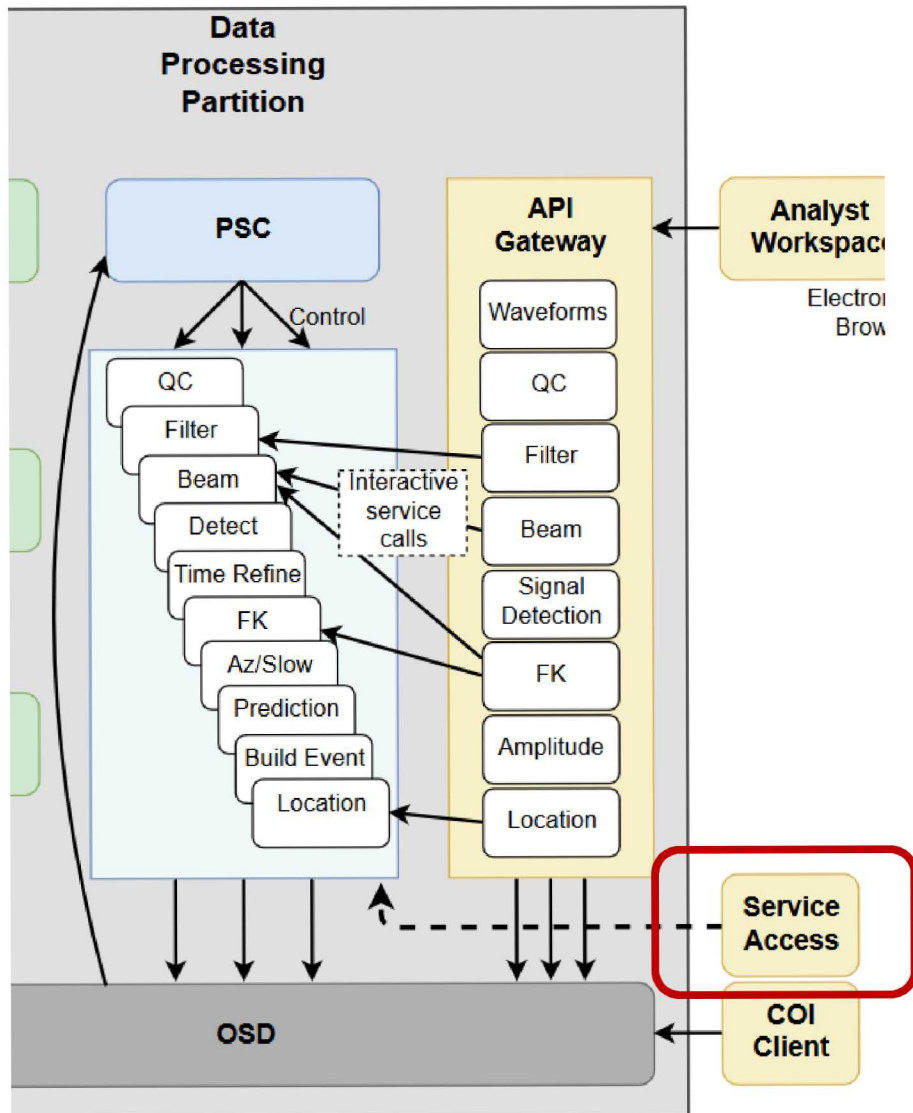
Slowness and Azimuth
Measurements (from FK)

Arrival Time of Second
Signal Detection
Hypothesis (after AIC)

```
rejected: raise
__signal_detection_hypotheses__
creationInfoId: 00000000-0000-0000-0000-000000000000
featureMeasurements:
{
  "id": "28dfb4e6-0c06-4676-acc9-c83d7d07b37e",
  "channelSegmentId": "1e216f05-ca4a-45dd-b06f-41ec4d662d45",
  "measurementValue": {
    "referenceTime": "2019-02-21T06:50:21.130874Z",
    "measurementValue": {
      "value": 4.009190095884129,
      "standardDeviation": 0.39545891407240613,
      "units": "SECONDS_PER_DEGREE"
    }
  },
  "featureMeasurementType": "SLOWNESS"
},
{
  "id": "5ddb724d-c9e4-4020-99e9-e799f50e22d7",
  "channelSegmentId": "1e216f05-ca4a-45dd-b06f-41ec4d662d45",
  "measurementValue": {
    "referenceTime": "2019-02-21T06:50:21.130874Z",
    "measurementValue": {
      "value": 56.309932474020236,
      "standardDeviation": 5.653840741165415,
      "units": "DEGREES"
    }
  },
  "featureMeasurementType": "SOURCE_TO_RECEIVER_AZIMUTH"
},
{
  "id": "e2a8dcc2-5fd0-477e-991d-81ca8f9d88e8",
  "channelSegmentId": "a6e9decd-d62a-4bfa-86a4-86edb65362b8",
  "measurementValue": {
    "value": "UNKNOWN",
    "confidence": 1.0
  },
  "featureMeasurementType": "PHASE"
},
{
  "id": "baa8eca7-5071-4d09-b931-db8b32124ad3",
  "channelSegmentId": "a6e9decd-d62a-4bfa-86a4-86edb65362b8",
  "measurementValue": {
    "value": "2019-02-21T06:52:02.450Z",
    "standardDeviation": "PT0.17509373S"
  },
  "featureMeasurementType": "ARRIVAL_TIME"
}
id: 2d1c4f15-d54e-483f-9fd1-3f9407919e3b
parentSignalDetectionId: e91c2c6a-7f93-4726-9ea5-843d057dec26
rejected: False
```

Access to Processing Services





- “Control” services expose endpoints for claim check (OSD references) or streaming data
 - waveform-qc-control-service
 - filter-control-service
 - beam-control-service
 - signal-detector-control-service
 - fk-control-service
 - fp-service
 - signal-detection-association-control-service
 - event-location-control-service

Example: Access Feature Prediction Service

Compute FeaturePredictions from source and receiver locations (streaming)	
Path	/feature-measurement/prediction/for-source-and-receiver-locations
Method	POST
Summary	Computes and returns feature predictions based on an event location (latitude, longitude, depth, and time) and multiple receiver locations (latitude, longitude, depth, and elevation).
Description	Computes and returns FeaturePrediction COI objects based on a list of FeatureMeasurementTypes of the predictions to calculate, a source EventLocation, one or more receiver Locations, a PhaseType, a named earth model, and a potentially empty list of prediction corrections. ...
Request Content Type	application/json or application/msgpack
Request Body	1.featureMeasurementTypes - List of FeatureMeasurementType COI enumerations describing the types of feature predictions to calculate. 2.sourceLocation - An EventLocation COI object describing the location of an event producing a signal. 3.receiverLocations - A list of Location COI objects. This operation uses these locations to compute FeaturePredictions of signals produced by an event at the provided sourceLocation. 4.phase - A PhaseType COI enumeration describing the type of signal for which this operation will calculate predictions. 5.model - The String name of an EarthModel used to calculate the predictions. 6.corrections (optional) - List of FeaturePredictionCorrectionDefinition COI objects describing the corrections to apply to the prediction. The list can be omitted if no predictions are requested. 7.processingContext - A ProcessingContext COI object describing where this feature prediction occurs within the processing workflow.
Response Content Type	application/json or application/msgpack for status 200 responses text/plain for non-status 200 responses
Response Body	featurePredictions - List of FeaturePrediction COI objects computed from the parameters provided in the request body.

Example: Request Magnitude Corrections

Set up the Request

```
data = ""{
    "featureMeasurementTypes" : [
        {
            "featureMeasurementTypeName": "MAGNITUDE_CORRECTION"
        }
    ],
    "sourceLocation" : {
        "latitudeDegrees" : 0.0,
        "longitudeDegrees" : 0.0,
        "depthKm" : 50.0,
        "time" : 0.0
    },
    "receiverLocations" : [%s],
    "phase" : "%s",
    "model" : "ak135",
    "corrections" : [],
    "processingContext" : {
        "analystActionReference" : null,
        "processingStepReference" : {
            "processingStageIntervalId" : "65054172-557f-42ac-894f-941cf55ca5f2",
            "processingSequenceIntervalId" : "67c8ef40-d6b4-41d3-a825-733cc22391be",
            "processingStepId" : "0817a095-4a1f-4d8f-8897-3d55777c70b5"
        }
    },
    "storageVisibility" : "PUBLIC"
}"" % (stadata, phase)
#print(data)
```

POST to Service

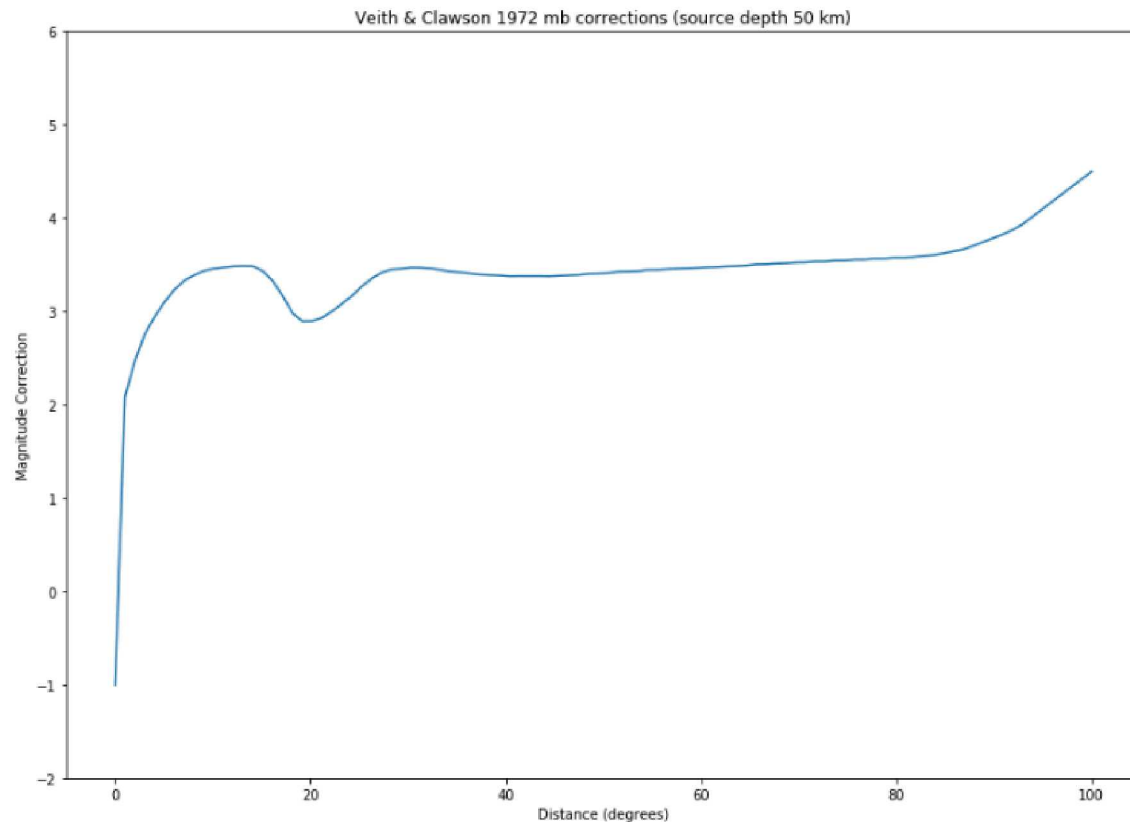
```
response = requests.post("http://testbed:9203/feature-measurement/prediction/for-source-and-receiver-locations", data, verify=True)
```

```
#print(response)
```

Plot the Magnitude Corrections for a Range of Distances

```
In [58]: # P Phase
LONGITUDE_MIN = 0.0
LONGITUDE_MAX = 100.0
NUM_SAMPLES = 100
phase = "p"
longitudes = np.linspace(LONGITUDE_MIN, LONGITUDE_MAX, NUM_SAMPLES)
corrections = findCorrections(phase, longitudes)
plt.figure(figsize=(14,10))
plt.plot(np.array(longitudes), np.array(corrections))
plt.xlabel('Distance (degrees)')
plt.ylabel('Magnitude Correction')
plt.ylim([-2,6])
plt.title('Veith & Clawson 1972 mb corrections (source depth 50 km)')
```

Out[58]: Text(0.5, 1.0, 'Veith & Clawson 1972 mb corrections (source depth 50 km)')



END