



Deliverable

Lambda Architecture Operations Guide

Prepared for

McDonald's

© 2016 Informatics IQ, Inc.
Confidential. All rights reserved.

May 24, 2016

Table of Contents

ARCHITECTURAL OVERVIEW	4
PLATFORM OVERVIEW	6
LAMBDAOPS USER	6
DIRECTORY PATHS	6
NETWORK DIAGRAM	7
HOST NAMES	10
HTTP ENDPOINT	11
OPERATION	11
ELASTIC LOAD BALANCER	11
MESSAGE METADATA	17
SPARK	18
MASTER	18
OPERATION	18
DYNAMODB	20
TABLE SUMMARY	20
TABLE CAPACITY	21
ACCESS CONTROL	23
KAFKA	26
OPERATION	26
LAMBDA CONSOLE	27
LANDING PAGE	27
DASHBOARD	27
ON-DEMAND	29
ADMINISTRATION - REDSHIFT	31
ADMINISTRATION - DATASOURCES	32
ADMINISTRATION - NODES	36
ADMINISTRATION - COMMANDS	38
ADMINISTRATION - JOBS	39
ADMINISTRATION - MESSAGES	41
ADMINISTRATION – MESSAGE METERS	41
ADMINISTRATION – ROUTING RULES	43
ADMINISTRATION – PENDING USER REQUESTS	44
ADMINISTRATION – PUBLISHER MANAGEMENT	46

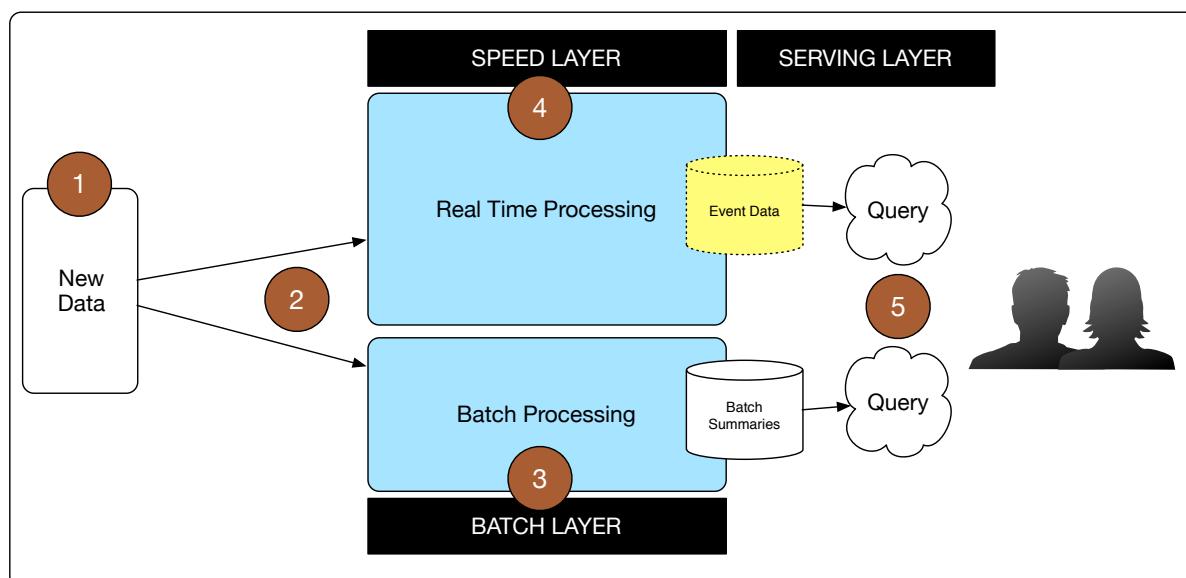
ADMINISTRATION – API ACCOUNT MANAGEMENT	49
PROFILE AND SIGN OUT	50
USER REGISTRATION	52
COMMAND DEFINITIONS	53
JOB DEFINITIONS	54
OPERATION	54
<u>EMR</u>	<u>56</u>
RESIZE	56
<u>REDSHIFT</u>	<u>57</u>
RESIZE	58
TERMINATE	58
CONNECTING TO REDSHIFT	59
<u>APPENDIX</u>	<u>60</u>
DATA LAKE INGEST	61
AWS RESOURCE TAGGING	68
EXPORTING JAVASCRIPT LOGS	70
POSTMAN AUTHORIZATION SETUP	72
SCREEN COMMAND REFERENCE	75
IAM PERMISSION RECOMMENDATION	75

Architectural Overview

The Lambda architecture is a high-throughput mixed workload data-processing architecture designed to handle massive quantities of data by leveraging both batch- and stream-processing. This architectural approach strikes a balance between latency, throughput, and fault-tolerance by using periodic batch processing (“batch layer”) to provide comprehensive views of historical datasets, while simultaneously using real-time stream processing (“speed layer”) to provide up-to-the-minute views of event datasets. These two views are materialized for presentation (“serving layer”) often combined to represent a holistic context for direct end-users reporting and downstream information consumers (human and machine).

The traditional Lambda design is shown below and explained using the following numbering scheme on the diagram:

1. New data is provided to the system at different intervals. Real-time or near real-time data is referred to as events. Data flowing into the system periodically (e.g. daily, hourly, weekly) are referred to as batch.
2. As data enters the system, it is processed by one of two layers: The Batch and Speed Layers.
3. Raw immutable data is deposited into the batch layer. Data is kept in its raw form so that analysis is isolated from any business transformations because it is possible, even likely, that such rules could change.
4. Events enter the system through the speed layer so that they may be processed in real-time; however, events are also placed in the batch layer so that this information may be included in new analytic analysis or in periodic batch processing jobs.
5. Most transformed or manipulated data is persisted in other data stores so that it is easily consumed or analyzed. Traditionally, tools for analyzing event level data or batch data are very different and likely consumed by different roles. Ideally, historical data from the batch layer is used to build predictive analytic models that are used to analyze real-time events in order to detect situations requiring immediate action by human and machine actors.



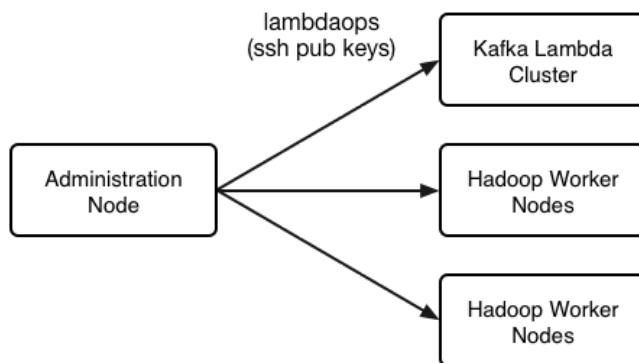
The proposed architecture enables the ability to identify key behaviors based on historical models and just-in-time interpretations of real-time event streams. As new events enter the system automated monitors initiate human- and machine-readable notifications to allow proactive responses to a wide range of conditions, thresholds, models, and event outcomes.

Platform Overview

CAUTION – in some cases Microsoft Word will automatically replace a “-“ (dash character) with a Unicode equivalent. Please use caution if you decide to copy/paste directly from this document – consider pasting to a standard text editor first to allow you to confirm proper syntax. This means you.

lambdaops User

The lambdaops user is the identity that most of the platform's process will run as. The following diagram summarizes the ssh key relationship relative to the Admin node (source) and the lambda nodes (targets, recipients of the public ssh key). The intent of the lambdaops user (and group) is to limit access to those with sudo access (to elevate to lambdaops) without requiring full-blown root/root-equivalent permission. That is, users that operate, manage, or administer the platform should only have sudo-to-lambdaops privileges – root access is not necessary or necessary for only a select number of administrators.



Directory Paths

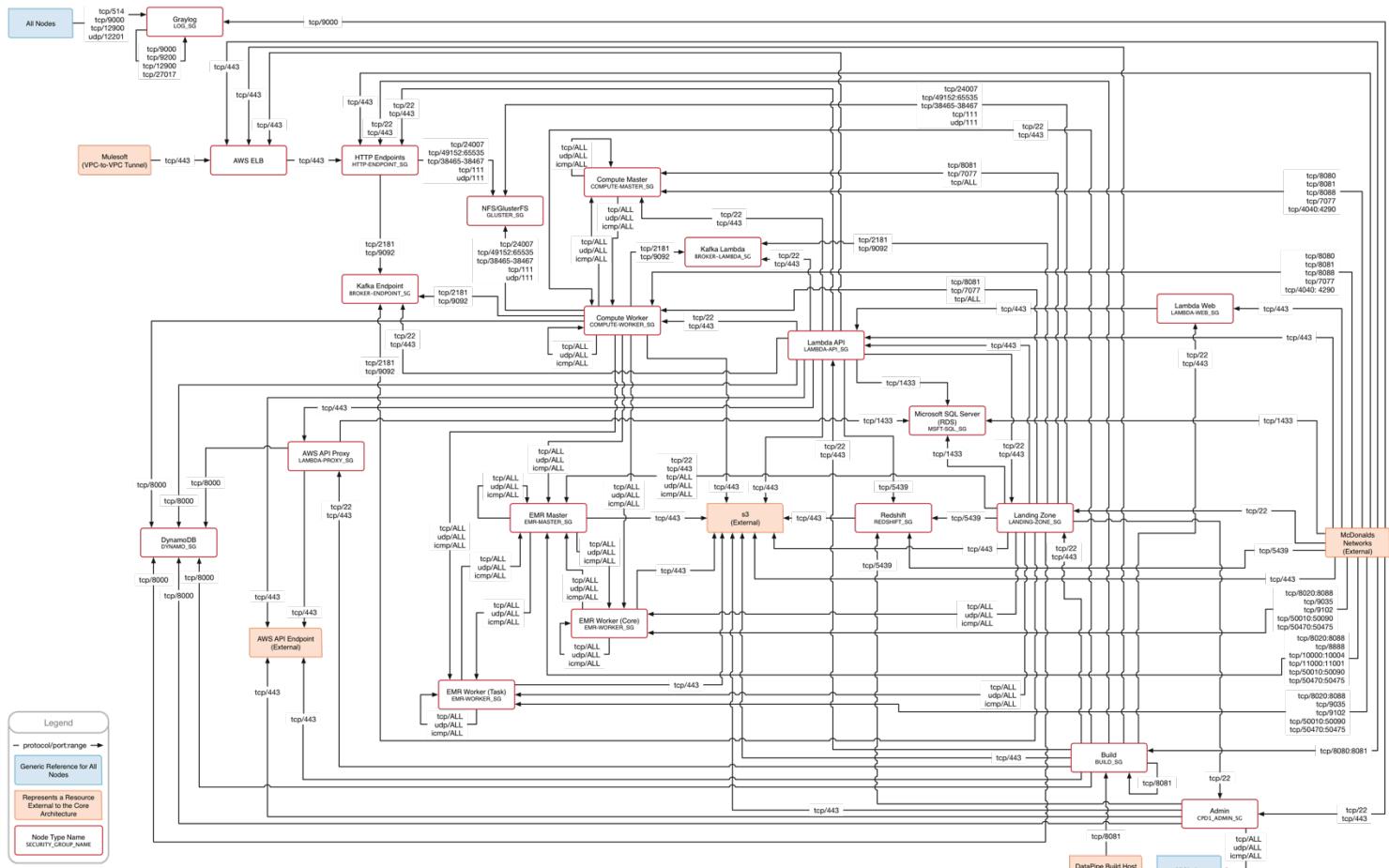
The following table describes the platform-specific directory paths used for this solution. All other Linux-related directories have not been customized (i.e. they remain standard and customary). Subsequent sections will call out these or other specific paths as necessary for clarity.

Directory	Purpose
/usr/local/mcd	This is the top-level directory for all McDonald's-specific applications and configurations
/var/log/lambda	This is the top-level log directory for all platform-related log files
/opt	The [standard] top-level directory for optional software. In this case, this directory contains dependent software components such as Kafka or Spark

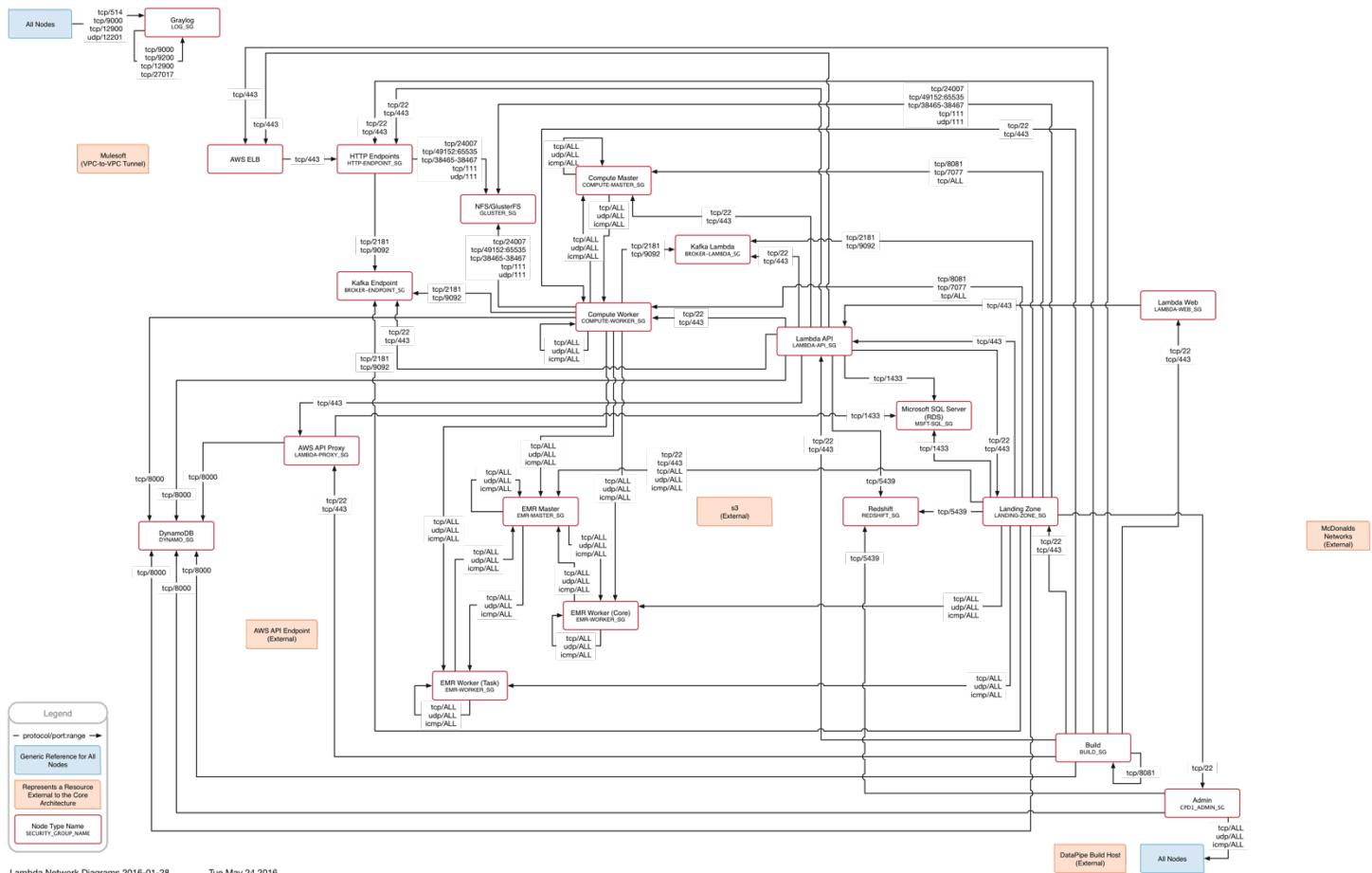
Network Diagram

The following is a diagram of the network paths enforced by security groups and AWS/McDonald's firewall rules. Consider this diagram as a snapshot in time; refer to the AWS security group and/or McDonald's firewall rule that is actually active for more information.

Entire network configuration



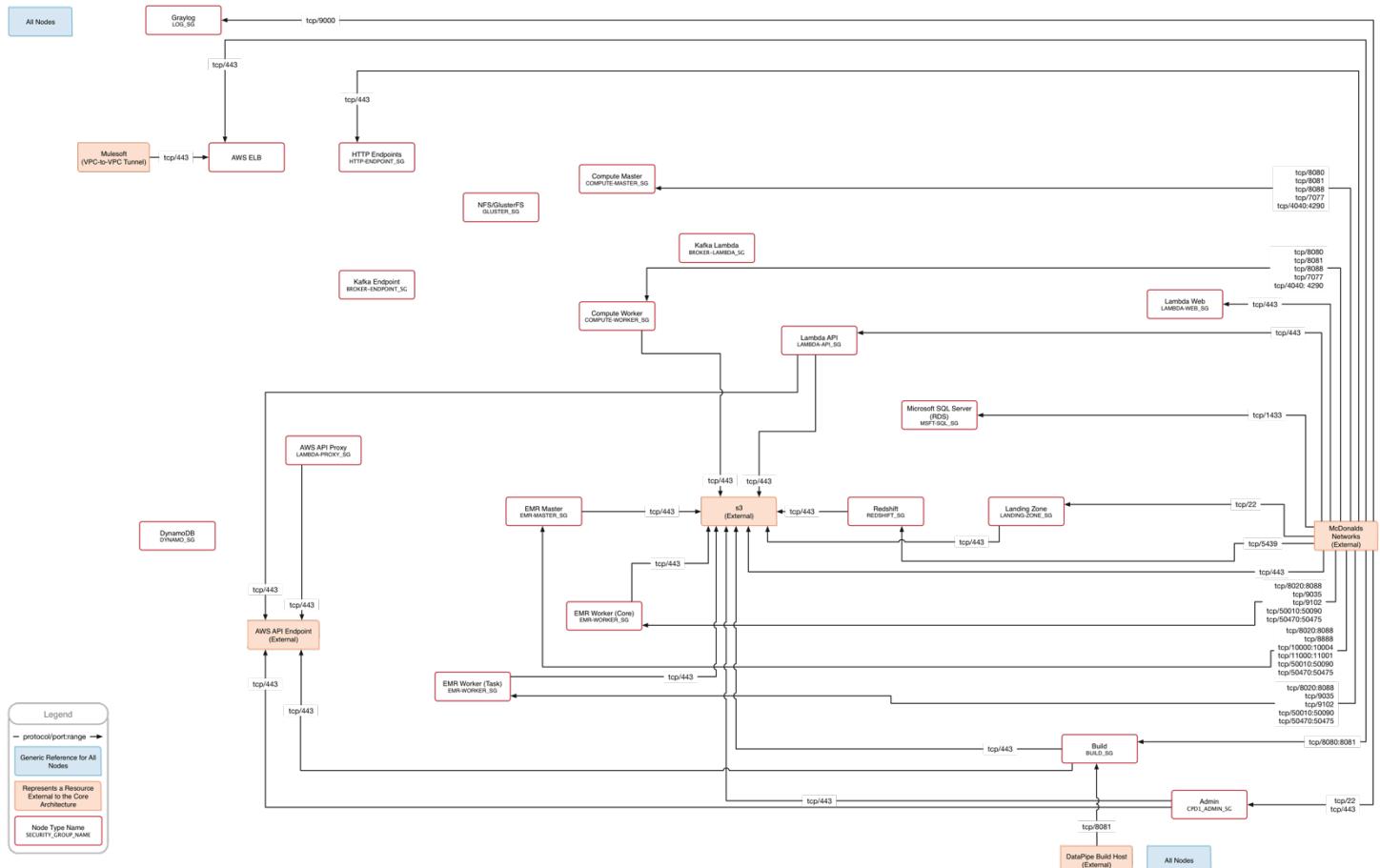
Internal network configuration (AWS)



Lambda Network Diagrams 2016-01-28

Tue May 24 2016

External network configuration



Lambda Network Diagrams 2016-01-28

Tue May 24 2016

Host Names

The following is a list of host name entries.

10.1.154.13	usaecducc1lad1.mcd.com	usaecducc1lad1	# us-east-cprod_us-cds-cpd1_admin1
10.1.154.54	usaecducc1bel.mcd.com	usaecducc1bel	# us-east-cprod_us-cds-cpd1_broker-endpoint1
10.1.154.57	usaecducc1be2.mcd.com	usaecducc1be2	# us-east-cprod_us-cds-cpd1_broker-endpoint2
10.1.154.59	usaecducc1be3.mcd.com	usaecducc1be3	# us-east-cprod_us-cds-cpd1_broker-endpoint3
10.1.154.50	usaecducc1b11.mcd.com	usaecducc1b11	# us-east-cprod_us-cds-cpd1_broker-lambda1
10.1.154.37	usaecducc1b12.mcd.com	usaecducc1b12	# us-east-cprod_us-cds-cpd1_broker-lambda2
10.1.154.7	usaecducc1b13.mcd.com	usaecducc1b13	# us-east-cprod_us-cds-cpd1_broker-lambda3
10.1.154.40	usaecducc1b1d1.mcd.com	usaecducc1b1d1	# us-east-cprod_us-cds-cpd1_build1
10.1.154.46	usaecducc1cm1.mcd.com	usaecducc1cm1	# us-east-cprod_us-cds-cpd1_compute-master1
10.1.154.41	usaecducc1cm2.mcd.com	usaecducc1cm2	# us-east-cprod_us-cds-cpd1_compute-master2
10.1.154.21	usaecducc1cm3.mcd.com	usaecducc1cm3	# us-east-cprod_us-cds-cpd1_compute-master3
10.1.154.18	usaecducc1cw1.mcd.com	usaecducc1cw1	# us-east-cprod_us-cds-cpd1_compute-worker1
10.1.154.31	usaecducc1cw2.mcd.com	usaecducc1cw2	# us-east-cprod_us-cds-cpd1_compute-worker2
10.1.154.61	usaecducc1cw3.mcd.com	usaecducc1cw3	# us-east-cprod_us-cds-cpd1_compute-worker3
10.1.154.4	usaecducc1cw4.mcd.com	usaecducc1cw4	# us-east-cprod_us-cds-cpd1_compute-worker4
10.1.154.56	usaecducc1cw5.mcd.com	usaecducc1cw5	# us-east-cprod_us-cds-cpd1_compute-worker5
10.1.154.11	usaecducc1g11.mcd.com	usaecducc1g11	# us-east-cprod_us-cds-cpd1_gluster1
10.1.154.39	usaecducc1g12.mcd.com	usaecducc1g12	# us-east-cprod_us-cds-cpd1_gluster2
10.1.154.20	usaecducc1g13.mcd.com	usaecducc1g13	# us-east-cprod_us-cds-cpd1_gluster3
10.1.154.22	usaecducc1hep1.mcd.com	usaecducc1hep1	# us-east-cprod_us-cds-cpd1_http-endpoint1
10.1.154.23	usaecducc1hep2.mcd.com	usaecducc1hep2	# us-east-cprod_us-cds-cpd1_http-endpoint2
10.1.154.62	usaecducc1hep3.mcd.com	usaecducc1hep3	# us-east-cprod_us-cds-cpd1_http-endpoint3
10.1.154.47	usaecducc1lp1.mcd.com	usaecducc1lp1	# us-east-cprod_us-cds-cpd1_lambda-proxy1
10.1.154.36	gdapi.mcd.com gdapi	usaecducc1la1	# us-east-cprod_us-cds-cpd1_lambda-api1
10.1.154.32	usaecducc1lw1.mcd.com	usaecducc1lw1	# us-east-cprod_us-cds-cpd1_lambda-web1
10.1.154.43	usaecducc1lz1.mcd.com	usaecducc1lz1	# us-east-cprod_us-cds-cpd1_landing-zone1
10.1.154.38	usaecducc1log1.mcd.com	usaecducc1log1	# us-east-cprod_us-cds-cpd1_log1

Http Endpoint

The http endpoints act as the entry point for the speed and batch layers. They respond to standard http protocol requests and use standard authentication. The Amazon Elastic Load Balancer (ELB) is the actual recipient of the request and forwards to any of the endpoint servers that are available in a round-robin order.

Operation

The following commands are used to start and stop the endpoints. The commands are the same for each node. Execute the following command on each node to start or stop the http endpoint.

Start

SSH to the endpoint node of interest as the ec2_user and run

```
screen -dmS endpoint bash -c 'cd /usr/local/mcd/lambda_ep-1.0 && sudo ./bin/lambda_ep -Djdk.tls.ephemeralDHKeySize=2048 -Djdk.tls.rejectClientInitiatedRenegotiation=true -J-Xms512M -J-Xmx3g -J-server'
```

Stop

SSH to the endpoint node of interest as the ec2_user and verify that an existing session exists (refer to the appendix of this document for a short Screen reference)

```
screen -ls
```

Connect to the existing session

```
screen -r endpoint
```

Now shutdown the service by simply pressing the control-c combination, you can now exit your screen session by typing

Exit

Large messages (payloads greater than 1 Mb) are shunted to the shared mount point (each endpoint and the Spark workers have access to read/write to this location):

```
/lambda/shunt/us-east-1
```

Elastic Load Balancer

To identify and administer the ELB, navigate to the AWS console and select the EC2 Dashboard.

The screenshot shows the AWS EC2 Management Console dashboard. On the left sidebar, under the 'LOAD BALANCING' section, the 'Load Balancers' option is selected. The main content area displays various Amazon EC2 resource statistics for the US East (N. Virginia) region:

- 520 Running Instances
- 0 Dedicated Hosts
- 1303 Volumes
- 23 Key Pairs
- 0 Placement Groups
- 129 Elastic IPs
- 3721 Snapshots
- 105 Load Balancers
- 407 Security Groups

A callout box highlights the 'Create Instance' section, which includes a 'Launch Instance' button and a note stating: "Build and run distributed, fault-tolerant applications in the cloud with Amazon Simple Workflow Service."

Scheduled Events section shows 1 instance has scheduled events.

AWS Marketplace section lists products like SAP HANA One 244GB and TIBCO Spotfire® Analytics Platform (Hourly).

On the lower left-hand side select Load Balancers and then type “cds” to filter the resources as follows.

The screenshot shows the AWS EC2 Management Console with the 'Load Balancers' section selected in the sidebar. A search bar at the top of the main content area contains the text 'cds'. The results table displays two load balancers:

Load Balancer Name	DNS name	Port Configuration	Availability Zones	Instance Count	Health Check	Created At	Monitoring
US-CDS-STG1-HTTP-ENDP...	internal-US-CDS-STG1-HTTP...	443 (HTTPS, ACM Certificate: e4216...)	us-east-1b	3 Instances	TCP:443	January 15, 2016 at 4:36:54...	
US-CDS-CPD1-HTTP-ENDP...	internal-US-CDS-CPD1-HTTP...	443 (HTTPS, ACM Certificate: 278fd...)	us-east-1b	3 Instances	TCP:443	January 15, 2016 at 4:40:50...	

The sidebar also shows the 'COMMANDS' section is selected.

This is the Description section of the ELB, note the security group and vpc identifiers.

The Instances section lists the ec2 instances that the ELB will redirect to and the current status of those instances – under normal operating conditions they should all read “InService” and a total instance count of three.

The Health Check section shows the rules used to determine the status of a given instance. In this case the instance will be checked on port 443 every 5 seconds and will be marked as unhealthy after 7 consecutive failures. In the event an instance is marked as unhealthy, that instance will have to successfully respond 3 consecutive times before it will be marked as “InService” (the ELB will automatically redirect traffic any of the other instances).

Load Balancer Name	DNS name	Port Configuration	Availability Zones	Instance Count	Health Check	Created At	Monitoring
US-CDS-STG1-HTTP-ENDP...	internal-US-CDS-STG1-HTTP-ENPD...	443 (HTTPS, ACM Certificate: e4216...	us-east-1b	3 Instances	TCP:443	January 15, 2016 at 4:36:54...	
US-CDS-CPD1-HTTP-ENDP...	internal-US-CDS-CPD1-HTTP-ENPD...	443 (HTTPS, ACM Certificate: 278fd...	us-east-1b	3 Instances	TCP:443	January 15, 2016 at 4:40:50...	

Load balancer: US-CDS-CPD1-HTTP-ENDPOINT-ELB1

Description Instances **Health Check** Monitoring Security Listeners Tags

Ping Target: TCP:443
 Timeout: 3 seconds
 Interval: 5 seconds
 Unhealthy threshold: 7
 Healthy threshold: 3
[Edit Health Check](#)

The Monitoring section shows the current operational metrics and has been left out for brevity. The Security section shows the relationship between the ELB and the associated security group.

Load Balancer Name	DNS name	Port Configuration	Availability Zones	Instance Count	Health Check	Created At	Monitoring
US-CDS-STG1-HTTP-ENDP...	internal-US-CDS-STG1-HTTP-...	443 (HTTPS, ACM Certificate: e4216...)	us-east-1b	3 Instances	TCP:443	January 15, 2016 at 4:36:54...	
US-CDS-CPD1-HTTP-ENDP...	internal-US-CDS-CPD1-HTTP-...	443 (HTTPS, ACM Certificate: 278fd...)	us-east-1b	3 Instances	TCP:443	January 15, 2016 at 4:40:50...	

Load balancer: US-CDS-CPD1-HTTP-ENDPOINT-ELB1

Description Instances Health Check Monitoring **Security** Listeners Tags

Security Group ID	Name	Description
sg-cb7910b2	US-CDS-CPD1_HTTP-ENDPOINT_SG	US CDS CPD1 HTTP-ENDPOINT Security Group

Edit

The Listeners section shows the forwarding rule(s) that that ELB will use. In this case the ELB will only respond to secure traffic on port 443 (https) - ***there is NEVER a time that port 80 or any other unsecure channel should be configured – publishers are known in advance and will not self-discovery our endpoint services.*** The secure channel is maintained from the ELB to the receiving endpoint server.

The specific rule receives traffic [only] on port 443 and redirects to its designated endpoint instance also on port 443 as follows.

Message Metadata

Every message that is sent to the endpoint is merged with the following message metadata. The metadata is used to identify ownership, priority, and lineage of the source message. For payloads greater than 1 Mb, the message is referred to as a reference and the “payloadDataLocation” will contain the fully qualified path of the file. Since this file may be in transit, this metadata may change slightly as it moves from the GlusterFs server into S3. Once the file comes to rest in s3, only the metadata is sent from one process to the next along its pipeline.

```

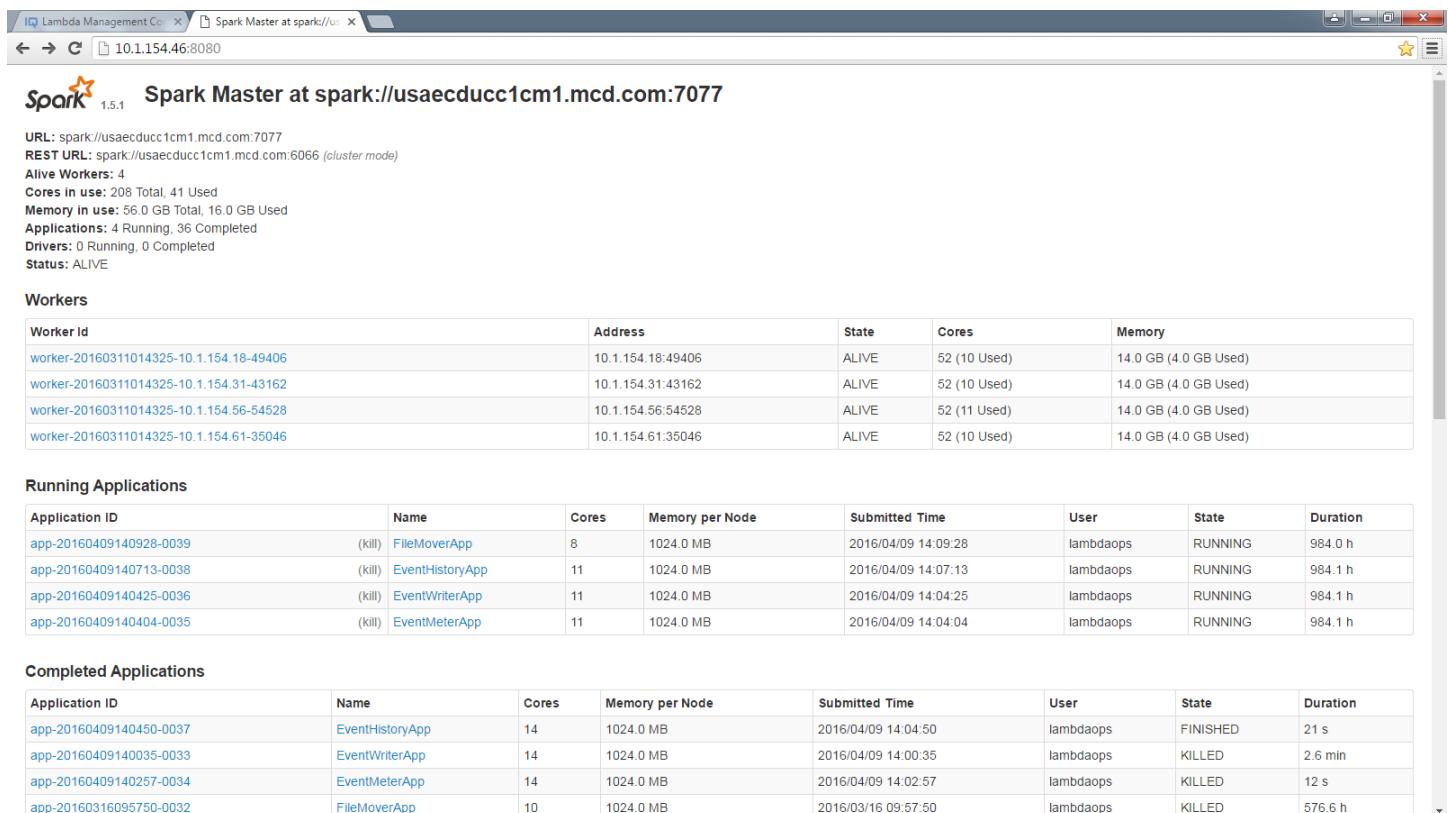
1  {
2      "msgMetaData": { Message metadata
3          "ipAddress": "10.200.8.254", Source ip address
4          "protocol": "HTTP/1.1", Source protocol
5          "senderContentType": "application/json", Source content type
6          "pubKey": "88dddfdd9-0153-1001-8723-d2741829b168", Publisher key
7          "status": "valid", Message status
8          "endpointResponseCode": 200, Response code sent to source
9          "organization": "organization|lambda", Publisher organization
10         "payloadType": "test.pub.mapping.offers.1", Payload (message) type
11         "defaultProcessMode": "X", Process mode - speed or batch
12         "messageSizeBytes": 90, Total message size
13         "payloadDataLocation": "none", External path to large files, none when N/A
14         "payloadContentEncoding": "none", Payload encoding, default none
15         "msgPayloadChecksum": "f2e2de4686...", The payload checksum
16         "msgMessageId": "c91df0cc-0153-1000-bede-87fa3470e21b", Unique message id
17         "lambdaProcessedTime": "2016-02-25T20:00:17.066Z", Last touch UTC timestamp
18         "endpointReceiptTime": "2016-02-25T20:00:15.952Z" Endpoint UTC timestamp
19     },
20     "msgPayloadData": { Payload - valid JSON is nested, base64 encoded otherwise
21         "example": {
22             "keyField": "sample value is here"
23         }
24     }
25 }
```

Spark

Apache Spark is the next generation of high-performance data processing framework for both stream and batch processing. Spark is fundamental to the Lambda Architecture and plays an integral role for data movement and machine learning. In the current implementation, at present Spark is exclusively used for data movement; machine learning and stream subscriptions may be incorporated at a later date.

Master

The Spark Master (<http://usaecdmc1.mcd.com:8080/>) is the resource coordinator and is used to monitor the various run-time aspects of both batch and streaming applications. For more information about the Spark Master refer to the documentation located at <http://spark.apache.org>.



Spark 1.5.1 **Spark Master at spark://usaecdmc1.mcd.com:7077**

URL: spark://usaecdmc1.mcd.com:7077
REST URL: spark://usaecdmc1.mcd.com:6066 (cluster mode)
Alive Workers: 4
Cores in use: 208 Total, 41 Used
Memory in use: 56.0 GB Total, 16.0 GB Used
Applications: 4 Running, 36 Completed
Drivers: 0 Running, 0 Completed
Status: ALIVE

Workers							
Worker ID	Address	State	Cores	Memory			
worker-20160311014325-10.1.154.18-49406	10.1.154.18:49406	ALIVE	52 (10 Used)	14.0 GB (4.0 GB Used)			
worker-20160311014325-10.1.154.31-43162	10.1.154.31:43162	ALIVE	52 (10 Used)	14.0 GB (4.0 GB Used)			
worker-20160311014325-10.1.154.56-54528	10.1.154.56:54528	ALIVE	52 (11 Used)	14.0 GB (4.0 GB Used)			
worker-20160311014325-10.1.154.61-35046	10.1.154.61:35046	ALIVE	52 (10 Used)	14.0 GB (4.0 GB Used)			

Running Applications

Application ID	Name	Cores	Memory per Node	Submitted Time	User	State	Duration
app-20160409140928-0039	(kill) FileMoverApp	8	1024.0 MB	2016/04/09 14:09:28	lambdaops	RUNNING	984.0 h
app-20160409140713-0038	(kill) EventHistoryApp	11	1024.0 MB	2016/04/09 14:07:13	lambdaops	RUNNING	984.1 h
app-20160409140425-0036	(kill) EventWriterApp	11	1024.0 MB	2016/04/09 14:04:25	lambdaops	RUNNING	984.1 h
app-20160409140404-0035	(kill) EventMeterApp	11	1024.0 MB	2016/04/09 14:04:04	lambdaops	RUNNING	984.1 h

Completed Applications

Application ID	Name	Cores	Memory per Node	Submitted Time	User	State	Duration
app-20160409140450-0037	EventHistoryApp	14	1024.0 MB	2016/04/09 14:04:50	lambdaops	FINISHED	21 s
app-20160409140035-0033	EventWriterApp	14	1024.0 MB	2016/04/09 14:00:35	lambdaops	KILLED	2.6 min
app-20160409140257-0034	EventMeterApp	14	1024.0 MB	2016/04/09 14:02:57	lambdaops	KILLED	12 s
app-20160316095750-0032	FileMoverApp	10	1024.0 MB	2016/03/16 09:57:50	lambdaops	KILLED	576.6 h

Operation

Start

SSH to the master 1 node and sudo to lambdaops and run

```
/opt/spark/default/sbin/start-all.sh
```

This script starts all the master and workers of the Spark cluster

Stop

SSH to the master 1 node and sudo to lambdaops and run

```
/opt/spark/default/sbin/start-all.sh
```

This script stops all the master and workers for the Spark cluster

DynamoDB

DynamoDB was chosen by the McDonald's architecture team as a good alternative to both HBase and Cassandra as a key/value store. The Lambda Architecture leverages DynamoDB for metadata, configurations, and operational metrics (referred to as message meters, or just metering).

Table Summary

The following is a table manifest of the tables defined in DynamoDB. The “cprod_us_cds” is a string NOT a schema. This has been added to avoid object collisions with other VPC users and/or other non-production Lambda environments.

Name	Key	Description
cprod_us_cds.Attribute	attributeId (String)	Generic key/value table used to store properties and settings
cprod_us_cds.AttributeDictionary	attributeDictionaryId (String)	Dictionary of attributes
cprod_us_cds.AttributeGroup	attributeGroupId (String)	Generic grouping of attributes
cprod_us_cds.AttributeValue	attributeValueId (String)	The value of the generic attribute as defined in Attribute
cprod_us_cds.Commands	commandId (String)	Command definitions and settings
cprod_us_cds.ComponentRelationships	pkHash (String)	Association table that groups Components together
cprod_us_cds.Components	componentId (String)	Represents the meta-resource definition
cprod_us_cds.Jobs	jobId (String)	Job definitions and settings
cprod_us_cds.JobStatus	jobStatusId (String)	Job status information and logging
cprod_us_cds.Languages	languageId (String)	Internationalization, currently English
cprod_us_cds.MemberGroups	pkHash (String)	Member group definitions
cprod_us_cds.MemberRelationships	pkHash (String)	Member relationships to groups and organizations
cprod_us_cds.MemberRoles	pkHash (String)	Platform role definitions
cprod_us_cds.Members	memberId (String)	Member (users and organizations) definitions
cprod_us_cds.MessageHistory	pkHash (String)	Stores incoming message metadata, used by the EventHistory Spark application to perform uniqueness checking
cprod_us_cds.Messages	messageId (String)	Message definitions
cprod_us_cds.MeterEntityMappings	pkHash (String)	Meter definition association table
cprod_us_cds.Meters	meterId (String)	Counter for a given time interval – new meters are used (aggregated) in real-time
cprod_us_cds.MeterValues	meterValueId (String)	Actual Meter (counter) for a given organization and message type
cprod_us_cds.OfferPrices	pkHash (String)	Price list for service offerings, currently AWS Redshift
cprod_us_cds.Offers	offerId (String)	Service bundles, currently AWS

		Redshift
cprod_us_cds.OrderItems	orderItemId (String)	All the relevant service line items – each Redshift cluster that is created is priced and provisioned based on its Order and OrderItems
cprod_us_cds.Orders	orderId (String)	The order summary entries for all provisioned services, currently Redshift and its associated datasets
cprod_us_cds.ResourceComponents	pkHash (String)	The provisioned services currently Redshift (active/inactive) and the datasource subscriptions (active/inactive)
cprod_us_cds.Resources	resourceId (String)	The grouping of resource components, considered the “header” for the ResourceComponent
cprod_us_cds.Roles	roleId (String)	Member role definitions
cprod_us_cds.RoutingRules	routingRuleId (String)	Rules used to dynamically route inbound messages for a given Spark application to one or more Kafka topics
cprod_us_cds.ScheduledJobs	scheduledJobId (String)	Job run history

Table Capacity

The following graphic summarizes the IOP capacity for the DynamoDB tables used. These values can be adjusted up or down as volume and throughput rate requirements dictate.

https://console.aws.amazon.com/dynamodb/home?region=us-east-1#tables:selected=usrd_metadata

Name	Status	Partition key	Sort key	Indexes	Total read capacity	Total write capacity
cprod_us_cds.Attribute	Active	attributeId (String)	-	0	5	5
cprod_us_cds.AttributeDictionary	Active	attributeDictionaryId (String)	-	0	5	5
cprod_us_cds.AttributeGroup	Active	attributeGroupId (String)	-	0	5	5
cprod_us_cds.AttributeValue	Active	attributeValueId (String)	-	0	5	5
cprod_us_cds.Commands	Active	commandId (String)	-	0	5	5
cprod_us_cds.ComponentRelationships	Active	pkHash (String)	-	0	5	5
cprod_us_cds.Components	Active	componentId (String)	-	0	5	5
cprod_us_cds.Jobs	Active	jobId (String)	-	0	5	5
cprod_us_cds.JobStatus	Active	jobStatusId (String)	-	0	5	5
cprod_us_cds.Languages	Active	languageId (String)	-	0	5	5
cprod_us_cds.MemberGroups	Active	pkHash (String)	-	0	5	5
cprod_us_cds.MemberRelationships	Active	pkHash (String)	-	0	5	5
cprod_us_cds.MemberRoles	Active	pkHash (String)	-	0	5	5
cprod_us_cds.Members	Active	memberId (String)	-	0	5	5
cprod_us_cds.MessageHistory	Active	pkHash (String)	-	0	5	5
cprod_us_cds.Messages	Active	messageId (String)	-	0	5	5
cprod_us_cds.MeterEntityMappings	Active	pkHash (String)	-	0	5	5
cprod_us_cds.Meters	Active	meterId (String)	-	0	5	5
cprod_us_cds.MeterValues	Active	meterValueId (String)	-	0	5	5
cprod_us_cds.OfferPrices	Active	pkHash (String)	-	0	5	5
cprod_us_cds.Offers	Active	offerId (String)	-	0	5	5
cprod_us_cds.OrderItems	Active	orderItemId (String)	-	0	5	5
cprod_us_cds.Orders	Active	orderId (String)	-	0	5	5
cprod_us_cds.ResourceComponents	Active	pkHash (String)	-	0	5	5
cprod_us_cds.Resources	Active	resourceId (String)	-	0	5	5
cprod_us_cds.Roles	Active	roleId (String)	-	0	5	5
cprod_us_cds.RoutingRules	Active	routingRuleId (String)	-	0	5	5
cprod_us_cds.ScheduledJobs	Active	scheduledJobId (String)	-	0	5	5

To view or change the provisioned IOP values, set the necessary capacity as shown below.

The screenshot shows the AWS DynamoDB console with the 'Capacity' tab selected for the 'cprod_us_cds.Attribute' table. The table has 5 Read capacity units and 5 Write capacity units provisioned. The estimated cost is \$2.91 / month.

Access Control

The users, groups, roles, and their associated relationships are stored in the Members, MemberGroups, MemberRoles, and MemberRelationships tables. The following screenshots illustrate the configuration for McDonald's within DynamoDB.

Members – This entity contains the core attributes of a member, members can be groups, users, organizations, publishers, etc.

DynamoDB

Tables

Reserved capacity

Create table Actions

Filter by table name

Name

- cprod_us_cds.Attribute
- cprod_us_cds.AttributeDictionary
- cprod_us_cds.AttributeGroup
- cprod_us_cds.AttributeValue
- cprod_us_cds.Commands
- cprod_us_cds.ComponentRelations
- cprod_us_cds.Components
- cprod_us_cds.Jobs
- cprod_us_cds.JobStatus
- cprod_us_cds.Languages
- cprod_us_cds.MemberGroups
- cprod_us_cds.MemberRelationships
- cprod_us_cds.MemberRoles
- cprod_us_cds.Members**
- cprod_us_cds.MessageHistory

cprod_us_cds.Members Close

Overview Items Metrics Alarms Capacity Indexes Triggers Access control

Scan: [Table] cprod_us_cds.Members: memberId Viewing 1 to 40 items

memberId	enabled	identifier	lastUpdate	memberType	description	label	attributes
group/validation	true	validation_cluster_12	2016-02-11T...	group			
user dave.male	true	dave.malecki@us.mcd.com	2016-03-14T...	user	Dave Malecki	Dave Malecki	{ "email": { ... }}
user Rick.Jorda	false	Rick.Jordan@us.mcd.com	2016-03-09T...	user	Rick Jordan	Rick Jordan	{ "email": { ... }}
group/validation	true	validation_cluster_5	2016-02-10T...	group			
group test2	true	test2	2016-03-14T...	group			
user admin@inf	true	admin@informaticsiq.com	2016-03-16T...	user	lambda admi...	lambda admi...	{ "password": { ... }}
group tda+_vali	true	tda+_validation_cluster_1c	2016-03-16T...	group			
group daas_tes	true	daas_test	2016-03-14T...	group			
group/validation	true	validation_cluster_1	2016-02-10T...	group			

MemberRoles – This entity contains the association of Members and their assigned Roles. Currently there are three roles: Administrators, Managers, and Readers

DynamoDB

Tables

Reserved capacity

Create table Actions

Filter by table name

Name

- cprod_us_cds.Attribute
- cprod_us_cds.AttributeDictionary
- cprod_us_cds.AttributeGroup
- cprod_us_cds.AttributeValue
- cprod_us_cds.Commands
- cprod_us_cds.ComponentRelations
- cprod_us_cds.Components
- cprod_us_cds.Jobs
- cprod_us_cds.JobStatus
- cprod_us_cds.Languages
- cprod_us_cds.MemberGroups
- cprod_us_cds.MemberRelationships
- cprod_us_cds.MemberRoles**
- cprod_us_cds.Members
- cprod_us_cds.MessageHistory
- cprod_us_cds.Messages
- cprod_us_cds.MeterEntityMappings
- cprod_us_cds.Meters
- cprod_us_cds.MeterValues
- cprod_us_cds.OfferPrices
- cprod_us_cds.Offers

cprod_us_cds.MemberRoles Close

Overview Items Metrics Alarms Capacity Indexes Triggers Access control

Scan: [Table] cprod_us_cds.MemberRoles: pkHash Viewing 1 to 4 items

pkHash	memberId	organizationId	roleId	attributes
user ron.strong	user ron.strong@us.mcd.com	organization mcd	manager	{ "key1": { "S...
user dave.mai	user dave.malecki@us.mcd.com	organization mcd	administrator	{ "key1": { "S...
user brenden.	user brenden.cyze@us.mcd.com	organization mcd	manager	{ "key1": { "S...
user admin@i	user admin@informaticsiq.com	organization root	administrator	

MemberRelationships – This entity contains the hierarchical relationships in and among Members (users, organizations, groups, publishers, etc.). This structure supports McDonald's market-level organizations (and users belonging to them) as well as Members (in this case, users or groups) belonging to more than one of these markets.

The screenshot shows the AWS DynamoDB console interface. On the left, the navigation pane lists various tables under the 'Tables' section, with 'cprod_us_cds.MemberRelationships' selected. The main area displays the table details for 'cprod_us_cds.MemberRelationships'. The 'Items' tab is selected, showing a list of 12 items. Each item has four fields: pkHash, ancestorId, descendantId, and entitySequence. The data is as follows:

	pkHash	ancestorId	descendantId	entitySequence
1	organization mcd	organization mcd.market.us	user brenden.cyze@us.mcd.com	3
2	organization mcd	organization mcd.market.ca	user john.ford@us.mcd.com	3
3	organization lan	organization lambda	user admin@informaticsiq.com	1
4	organization mcd	organization mcd	organization mcd.market.us	2
5	organization mcd	organization mcd.market.us	user Rick.Jordan@us.mcd.com	3
6	organization mcd	organization mcd.market.us	user ron.strong@us.mcd.com	3
7	organization lan	organization lambda	organization mcd	1
8	organization rcc	organization root	organization lambda	0
9	organization mcd	organization mcd	organization mcd.market.ca	2

Kafka

Apache Kafka is a high-performance publisher/subscriber framework. Kafka is fundamental to the Lambda Architecture and plays an integral role for coordinating event streams for data movement and Event Processors. In the current implementation, at present Kafka is exclusively used for data movement; Event Processors may be incorporated at a later date.

Operation

The following commands are used to start and stop the kafka brokers. The commands can be executed from the admin node.

Start

SSH to the admin node and sudo to lambdaops and run

```
/home/lambdaops/start-kafka-all.sh
```

This script starts all the zookeeper and kafka brokers for both the lambda and endpoint clusters

Stop

SSH to the admin node and sudo to lambdaops and run

```
/home/lambdaops/stop-kafka-all.sh
```

This script stops all the zookeeper and kafka brokers for both the lambda and endpoint clusters

List Topics

SSH to the admin node and run to list all the available topics for a given cluster

```
/opt/kafka/default/bin/kafka-topics.sh --zookeeper  
usaecducc1bel.mcd.com:2181,usaecducc1be2.mcd.com:2181,usaecducc1be3.mcd.com:2181 --list
```

Subscribe to Topic

SSH to the admin node and run to listen to the specific topic (by name)

```
/opt/kafka/default/bin/kafka-console-consumer.sh --zookeeper  
usaecducc1bel.mcd.com:2181,usaecducc1be2.mcd.com:2181,usaecducc1be3.mcd.com:2181 --topic  
iq_lambda_iot_event
```

Creating a Topic

SSH to the admin node and run to create a new topic (by name), change the number of partitions and the replications factor must be between 1 and 3, 3 being complete replication to all brokers in the configured cluster

```
/opt/kafka/default/bin/kafka-topics.sh --zookeeper  
usaecducc1bel.mcd.com:2181,usaecducc1be2.mcd.com:2181,usaecducc1be3.mcd.com:2181 --create  
--topic mapping_redemptions --partitions 10 --replication-factor 3
```

Lambda Console

The Lambda Console serves as the user interface for both end-users and administrators. This section describes all of the user interfaces used by each of these user types.

Landing Page

The landing page contains static content used to inform the end-user with the basic premise and terminology.

Welcome lambda administrator!

Service Health

- All services operating normally
- Updated: May 20 2016 13:38:27 GMT-0500
- Service Health Dashboard

Additional Resources

- Getting Started
- Read our documentation or view our training to learn more about this site.

What is a dataset?

A dataset can be thought of as a data extract with properties or characteristics. For example, a dataset has a name, a size, and an ETA (time to load). You as a user have the ability to subscribe to as many datasets as you wish, just note that the size and window of time have a direct impact on the monthly cost.

Where does my cluster actually live?

That depends on where you choose to create it. At present, we only support creating your cluster in the Amazon's North American East datacenter; however, we have plans to expand our offering to other Regions on an on-going basis.

Can I request a one-time snapshot of data?

Absolutely! In fact the idea is that you can create and destroy as many clusters as you need. That means you can have one cluster that you may decide to keep and have others that come and go. In each case you have the option to load a one-time snapshot or an automatic refresh.

What if the performance isn't what I expected could I change my cluster?

Yes. During the creation process you are walked through a step-by-step wizard. You are presented with pricing for all the configurations that suite the datasets that you have selected. It is at that time that you commit to a give type of cluster. Once the cluster is running, you have the option to increase and decrease the size within limits. For example, you cannot decrease the size of your cluster beyond the minimum required to store the datasets you selected.

Dashboard

Users navigate using the menu items located at the top of the web page, referred to as the “top-nav” (top navigation). Selecting Dashboard -> Meters shows the inbound message event counters.

<https://gdapi.mcd.com/console/welcome>

Lambda Management Console Dashboard On-Demand Administration My Account

Welcome lambda administrator!

Service Health

All services operating normally.

Updated: May 20 2016 13:28:04 GMT-0500

Service Health Dashboard

Additional Resources

Getting Started [Learn more](#)

Read our documentation or view our training to learn more about this site.

What is a dataset?

A dataset can be thought of as a data extract with properties or characteristics. For example, a dataset has a name, a size, and an ETA (time to load). You as a user have the ability to subscribe to as many datasets as you wish, just note that the size and window of time have a direct impact on the monthly cost.

Can I request a one-time snapshot of data?

Absolutely! In fact the idea is that you can create and destroy as many clusters as you need. That means you can have one cluster that you may decide to keep and have others that come and go. In each case you have the option to load a one-time snapshot or an automatic refresh.

What if the performance isn't what I expected could I change my cluster?

Yes. During the creation process you are walked through a step-by-step wizard. You are presented with pricing for all the configurations that suite the datasets that you have selected. It is at that time that you commit to a give type of cluster. Once the cluster is running, you have the option to increase and decrease the size within limits. For example, you cannot decrease the size of your cluster beyond the minimum required to store the datasets you selected.

Where does my cluster actually live?

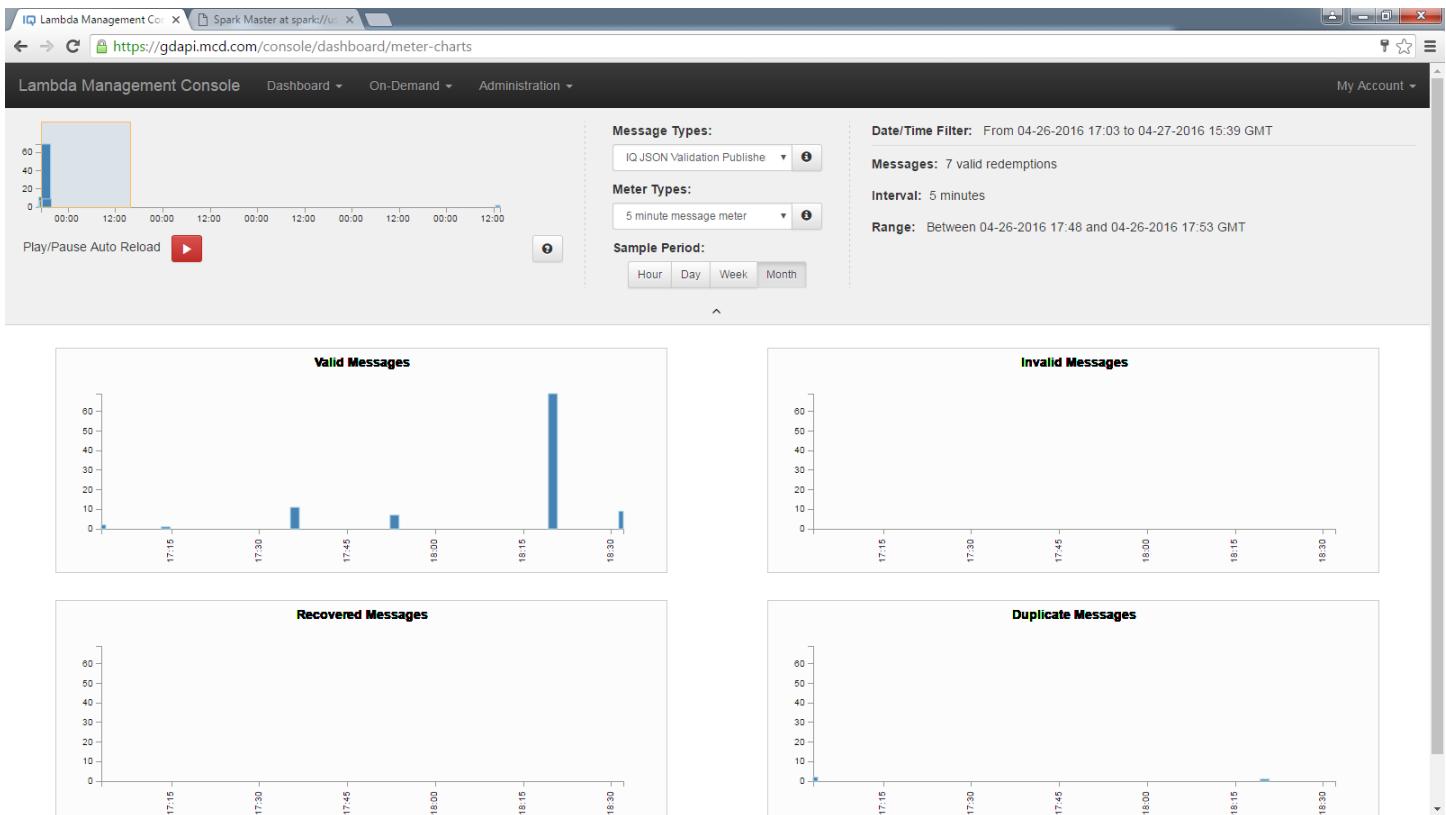
That depends on where you choose to create it. At present, we only support creating your cluster in the Amazon's North American East datacenter; however, we have plans to expand our offering to other Regions on an on-going basis.

What am I paying for?

For the Redshift (analytics database) cluster, you are paying for the cluster itself, intermediate storage used to load your cluster (on an as-needed basis), and your portion of the shared costs to host the environment.

<https://gdapi.mcd.com/console/dashboard/meter-charts>

The top section of the metering dashboard shows a selectable time line that allows user to filter on a given time window. The message type and meter type dropdowns provider additional filtering capabilities. The auto reload button will refresh all the graphs every 5 seconds.



On-Demand

Navigating to the On-Demand -> Manage Existing Cluster.

Welcome administrator!

What is a dataset?
A dataset can be thought of as a data extract with properties or characteristics. For example, a dataset has a name, a size, and an ETA (time to load). You as a user have the ability to subscribe to as many datasets as you wish, just note that the size and window of time have a direct impact on the monthly cost.

Where does my cluster actually live?
That depends on where you choose to create it. At present, we only support creating your cluster in the Amazon's North American East datacenter; however, we have plans to expand our offering to other Regions on an on-going basis.

Can I request a one-time snapshot of data?
Absolutely! In fact the idea is that you can create and destroy as many clusters as you need. That means you can have one cluster that you may decide to keep and have others that come and go. In each case you have the option to load a one-time snapshot or an automatic refresh.

What am I paying for?
For the Redshift (analytics database) cluster, you are paying for the cluster itself, intermediate storage used to load your cluster (on an as-needed basis), and your portion of the shared costs to host the environment.

What if the performance isn't what I expected could I change my cluster?
Yes. During the creation process you are walked through a step-by-step wizard. You are presented with pricing for all the configurations that suite the datasets that you have selected. It is at that time that you commit to a give type of cluster. Once the cluster is running, you have the option to increase and decrease the size within limits. For example, you cannot decrease the size of your cluster beyond the minimum required to store the datasets you selected.

https://gdapi.mcd.com/#

The list of Redshift clusters is show below. Expanding a given cluster reveals the extended properties of that cluster.

Lambda Management Console > Spark Master at spark://us... > https://gdapi.mcd.com/console/resources/list

Lambda Management Console Dashboard On-Demand Administration My Account

Manage Existing Cluster

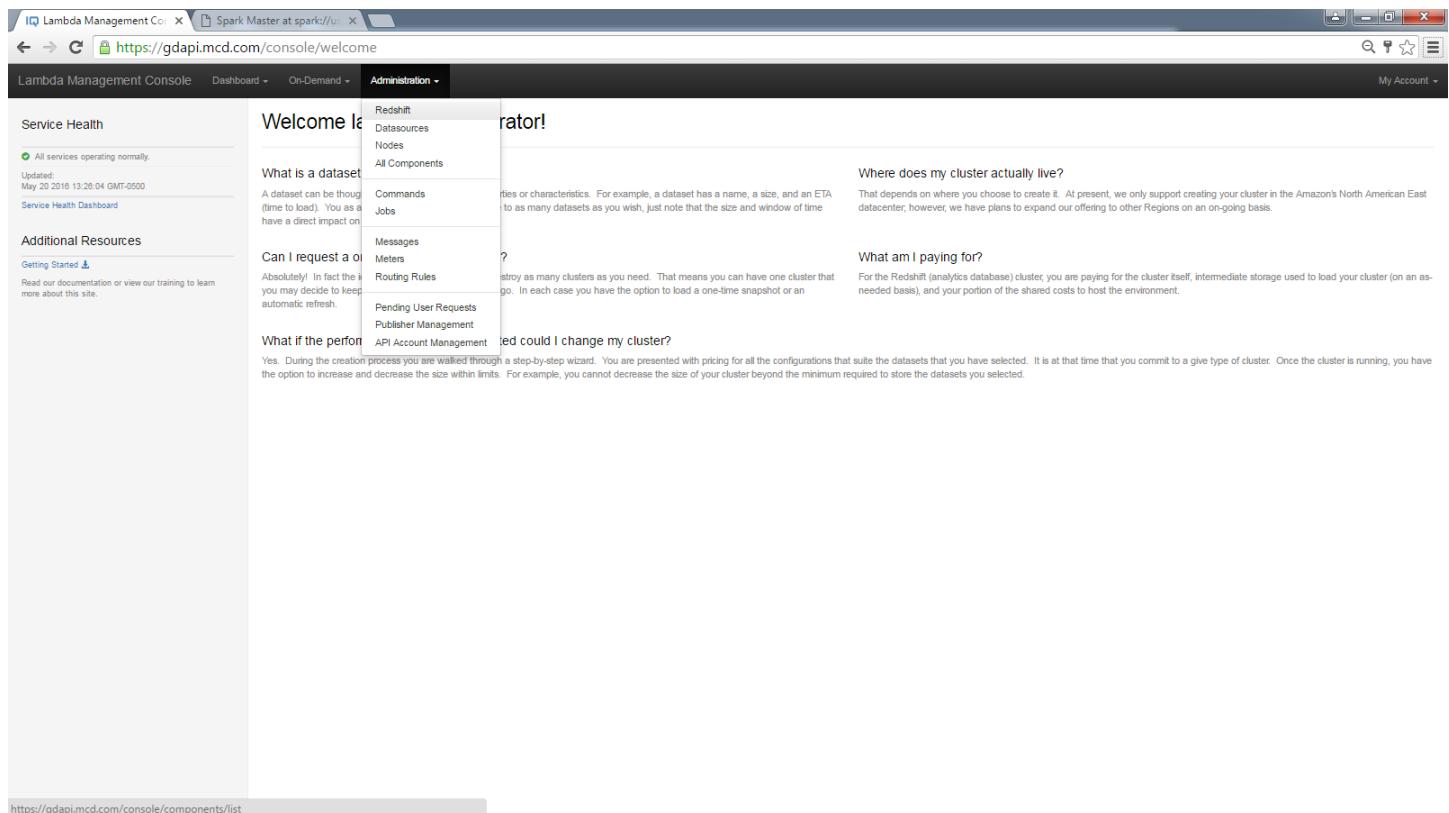
New Environment		Actions			
	Name	Description	Type	Owner	Status
<input checked="" type="radio"/>	DLY SLS TEST3	▼ DLY SLS TEST3	redshift	rstrong@us.mcd.com	Ready
<input checked="" type="radio"/>	Test GBAL_LCAT	▼ Test GBAL_LCAT	redshift	rstrong@us.mcd.com	Ready
<input checked="" type="radio"/>	Sprinklr Insights	▲ Sprinklr Insights	redshift	rstrong@us.mcd.com	Ready
		Date Created:	04/27/2016	Total Storage:	288 GB
		GBL Code:	195500019314	Available Storage:	287 GB
		Estimated Monthly Cost:	\$446.00	Resource Id:	organization mcd.market.us sprinklr_insights_7407d5c6-5a9b-45d4-a3d3-31b6bca68a2d
		Redshift Cluster Identifier:	us-east-cprod-us-cds-cpd1-redshift-195500019314-89820	Node Type:	dc1.large
		Database Endpoint Address:	us-east-cprod-us-cds-cpd1-redshift-195500019314-89820.ciuv2sg591g.us-east-1.redshift.amazonaws.com	Node Count:	2
		Database Endpoint Port:	5439	Redshift Admin Id:	admin
		Redshift Database Name:	ondemand	Redshift Password:	K_f9-vy_5F2#ro9O-p
<input checked="" type="radio"/>	A New Cluster	▼ A New Cluster	redshift	Rick.Jordan@us.mcd.com	Ready
<input checked="" type="radio"/>	Us Daily Sales Demo	▼ Us Daily Sales Demo	redshift	rstrong@us.mcd.com	Terminated

Attribute	Description
Date Created	The date the cluster was actually created
GBL Code	The cost center code specified during the creation process
Estimated Monthly Cost	The approximate fully-loaded (AWS + GTS + Datapipe) costs assuming a constant configuration – forecasts are based off the current configuration, therefore adding/removing nodes will immediately impact this forecast linearly
Redshift Cluster Identifier	The physical identifier that can be found in the AWS console
Database Endpoint Address	The fully qualified domain name of this Resource
Database Endpoint Port	The tcp/ip port number
Redshift Database Name	The database name created within the Redshift cluster during the provisioning process
Total Storage	The approximate storage available base on the current cluster configuration
Available Storage	The approximate remaining storage available base on the current cluster configuration
Resource Id	The internal name used by the Lambda APIs
Node Type	The AWS node configuration type
Node Count	The number of compute nodes (for multi-node)

Redshift Admin Id	clusters the “head” node is provided at no charge, and is not included in this count)
Redshift Password	The user name of the Redshift cluster
	The admin id's password (only visible to owners and admins)

Administration - Redshift

Navigating to Administration (admin users only) -> Redshift



The screenshot shows the Lambda Management Console interface. The top navigation bar includes links for 'Lambda Management Console', 'Dashboard', 'On-Demand', 'Administration', and 'My Account'. The main content area is titled 'Welcome lambda administrator!' and contains several sections:

- Service Health:** Shows 'All services operating normally' with an update timestamp of 'May 20 2018 13:26:04 GMT-0500'.
- Additional Resources:** Includes links for 'Getting Started' and 'Service Health Dashboard'.
- Redshift Administration:** A dropdown menu with options: Redshift, Datasources, Nodes, All Components.
- What is a dataset?**: Describes datasets as having a name, size, and ETA, and notes that many datasets can be created.
- Can I request a one-time snapshot?**: Notes that clusters can be destroyed and replaced.
- What if the performance is poor?**: Notes that clusters can be scaled up or down.
- Redshift could I change my cluster?**: Notes that clusters can be resized.
- Where does my cluster actually live?**: Notes that clusters are currently limited to the Amazon North American East datacenter.
- What am I paying for?**: Notes that costs include cluster itself, storage, and shared costs.
- FAQs:** A list of frequently asked questions about datasets, requests, performance, and cluster changes.

The URL in the browser address bar is <https://gdapi.mcd.com/console/welcome>.

This interface allows administrators to change the general properties of the Redshift cluster definition (not the actual Redshift cluster(s) themselves). In most circumstances the default values will need to be changed.

The screenshot shows the Lambda Management Console interface for editing a component. The URL is <https://gdapi.mcd.com/console/components/edit>. The page title is "Redshift Configuration".

Base Configuration

Owning Organization	Lambda	<input type="button" value="Edit"/>
Identifier	redshift	<input type="button" value="Edit"/>
Name	AWS Redshift	<input type="button" value="Edit"/>
Description	AWS Redshift Cluster Component	<input type="button" value="Edit"/>
Type	redshift	<input type="button" value="Edit"/>
Usage		<input type="button" value="Edit"/>

Attributes

Key	<input type="text"/>	<input type="button" value="Edit"/>
Value	<input type="text"/>	<input type="button" value="Edit"/>

Key	Value	Action
environmentName	us-east-cprod-us	<input type="button" value="Edit"/>
proximityName	cpd1	<input type="button" value="Edit"/>
roleName	redshift	<input type="button" value="Edit"/>
preferredMaintenanceWindow	sat:03:30-sat:04:00	<input type="button" value="Edit"/>
departmentName	cds	<input type="button" value="Edit"/>

Administration - Datasources

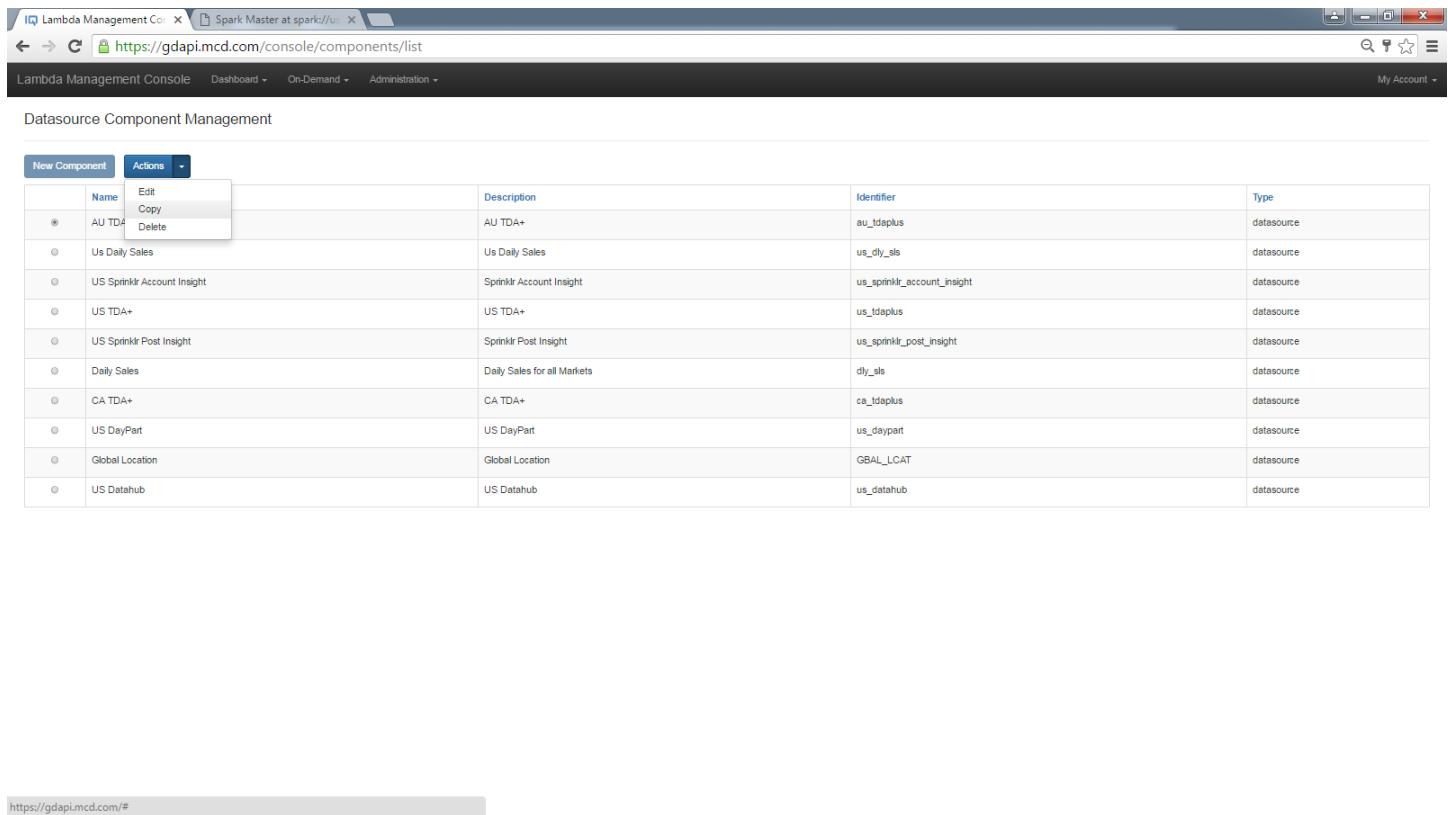
Navigate to Administration (admin users only) -> Datasources

Lambda Management Console Dashboard On-Demand Administration My Account ▾

Datasource Component Management

New Component	Actions ▾			
Name	Description		Identifier	Type
US Datahub	US Datahub - Old		us_datahub	datasource
US TDA - dummy	Used to test two distinct datasources w	should not work	us_tda_dummy	datasource
AU Datahub	AU Datahub		au_datahub	datasource
US Datahub	US Datahub	Pending User Requests	us_datahub	datasource
AU Datahub	AU Datahub	Publisher Management	au_datahub	datasource
Puerto Rico TDA+	Puerto Rico TDA+	API Account Management	pri_tdaplus	datasource
AU TDA+	AU TDA+		au_teradata	datasource
CA Datahub	CA Datahub		ca_datahub	datasource
CA TDA+	CA TDA+		ca_teradata	datasource
US TDA+	US TDA+		us_tdaplus	datasource
UK TDA+	UK TDA+		uk_teradata	datasource
UK Datahub	UK Datahub		uk_datahub	datasource
Rick Datasource	Rick Datasource		ricks_datasource	datasource
US TDA+	US TDA+ - Old		us_tdaplus	datasource

Shows the current datasource definitions. Datasources are associated with Organizations (defining in the Member and MemberRelationship tables). These associations drive security and allow/disallow visibility to a given datasource definition.



The screenshot shows the Lambda Management Console interface. The title bar includes the logo and the URL <https://gdapi.mcd.com/console/components/list>. The main header is "Datasource Component Management". Below it is a table with columns: Name, Description, Identifier, and Type. The table lists ten data sources. A context menu is open over the first row, showing options: New Component, Actions, Edit, Copy, and Delete. The "Copy" option is highlighted.

	Name	Description	Identifier	Type
•	AU TDA+	AU TDA+	au_tdeplus	datasource
•	Us Daily Sales	Us Daily Sales	us_dy_sls	datasource
•	US Sprinklr Account Insight	Sprinklr Account Insight	us_sprinklr_account_insight	datasource
•	US TDA+	US TDA+	us_tdeplus	datasource
•	US Sprinklr Post Insight	Sprinklr Post Insight	us_sprinklr_post_insight	datasource
•	Daily Sales	Daily Sales for all Markets	dy_sls	datasource
•	CA TDA+	CA TDA+	ca_tdeplus	datasource
•	US DayPart	US DayPart	us_deypart	datasource
•	Global Location	Global Location	GBAL_LCAT	datasource
•	US Datahub	US Datahub	us_datahub	datasource

<https://gdapi.mcd.com/>

The recommended approach for defining a new datasource is to “Copy” an existing definition to save time from repetitive data entry.

Lambda Management Console Dashboard On-Demand Administration My A

Datasource Configuration

Base Configuration

Owning Organization	Australia	i
Identifier	au_datahub	i
Name	AU Datahub	i
Description	AU Datahub	i
Type	datasource	i
Usage		i

Attributes

Key		i
Value		i
Copy to Attributes		

Key	Value	Action
redshiftS3Region	us-east-1	Edit
redshiftTablePrimaryKeyColumns	pos_busn_dt, pos_ord_key_id, mcd_gbal_lcat_id_nu, pos_evtnt_typ_cd, pos_itm_line_seq_nu, terr_cd	Edit
quantity	480000000000000	Edit
periods	36	Edit
redshiftTableName	datahub	Edit
unitsPerPeriod	10000000000	Edit
periodUOM	month	Edit
redshiftTableLoadStrategy	incremental	Edit
isoCountryAlpha3	840	Edit
redshiftTableDeleteKeyColumns	pos_busn_dt, terr_cd	Edit
quantityUOM	B	Edit
redshiftCopyOptionalParameters	gzip delimiter '\t' maxerror 100000 dateformat 'YYYY-MM-DD' timeformat 'YYYY-MM-DD HH:MI:SS' emptyasnull blanksasnull	Edit
redshiftS3BucketInitialLoadFolder	s3://s3daas/datahub/840/	Edit
redshiftS3BucketsubFolderDateFormat	yyyy-MM-dd	Edit
redshiftDdl	create table if not exists datahub (pos_busn_dt date, pos_ord_key_id varchar(20), lgcy_lcl_rfr_def_cd varchar(20), mcd_gbal_lcat_id_nu bigint, terr_cd int, pos_evtnt_typ_cd varchar(20), pos_area_typ_shrt_ds varchar(50), pos_trn strt_ts varchar(30), pos_ord_uniq_id varchar(20), pos_trn_typ_cd varchar(20), pos_paid_dvce_id varchar(20), pos_mfy_side_cd varchar(30), pos_prd_dlvr_meth_cd varchar(20), pos_tot_net_trn_am decimal(18,4), pos_tot_grss_trn_am decimal(18,4), pos_tot_npnd_net_trn_am decimal(18,4), pos_tot_npnd_grss_trn_am decimal(18,4), pos_tot_tax_am decimal(18,4), pos_tot_npnd_tax_am decimal(18,4), pos_ord strt_ts varchar(30), pos_ord_end_ts varchar(30), sld_menu_itm_id int, pos_itm_prd_typ_cd varchar(20), pos_itm_actn_cd varchar(20), pos_itm_sds varchar(50), pos_itm_lv_nu int, pos_itm_seq_nu int, pos_itm_line_seq_nu int, pos_itm_tot_qty int, pos_itm_qty_at int, pos_itm_qty_mod int, pos_itm_qty_min int, pos_itm_qty_max int, pos_itm_qty_pct float);	Edit

The critical fields to note are show below. Note that changes to these values will not impact existing configurations.

Name	Description
isoCountryAlpha3	The three character ISO country code
periods	The maximum number of period
periodUOM	Default is B, bytes
quantity	Size of all periods
quantityUOM	Default is B, bytes
redshiftCopyOptionalParameters	Redshift COPY command parameters
redshiftDdl	The DDL for the table definition – this SQL MUST be idempotent
redshiftS3BucketIncrementalLoadFolder	The s3 location for the incremental files for the given period. For example, “s3://s3daas/datahub_delta/840/” the ending “/” is required
redshiftS3BucketInitialLoadFolder	The s3 location for the incremental files for the given period. For example, “s3://s3daas/datahub/840/” the ending “/” is required
redshiftS3BucketSubFolderDateFormat	The Joda date (and time) format mask – this mask will be replaced with a date range for one-time and incremental loads
redshiftS3Region	The AWS region for the s3 bucket – ideally this will be the same region as the Redshift cluster
redshiftTableDeleteKeyColumns	A comma separated (no spaces) list of columns to use for incremental delete statements
redshiftTableLoadStrategy	One-time or incremental load strategies are currently available
redshiftTableName	Target table name
redshiftTablePrimaryKeyColumns	A comma separated (no spaces) list of the physical primary key columns
unitsPerPeriod	Average size of period

Administration - Nodes

Navigate to Administration (admin users only) -> Nodes

<https://gdapi.mcd.com/console/components/list>

These properties define the unique characteristic of the node. At present the node configurations have been created for Redshift, but can be generalized.

Key	Value	Action
memoryUOM	B	Edit
vpcSecurityGroupIds	sg-087910b1	Edit
dbName	ondemand	Edit
masterUserPassword	12345678Aa	Edit
storageUOM	B	Edit
deploymentHoursPerTerabyte	.70	Edit
storageType	hdd	Edit
clusterSubnetGroupName	us-east-cprod-us-cds-cpd1	Edit
ioUOM	GB/s	Edit
storageBias	.90	Edit
nodeType	ds2.xlarge	Edit
clusterType	multi-node	Edit
storage	200000000000	Edit
ECU	14	Edit
io	.4	Edit
publiclyAccessible	false	Edit
masterUsername	admin	Edit
vCPU	4	Edit
maximumNodes	32	Edit
memory	31000000000	Edit

The critical fields to note are show below. Note that changes to these values will not impact existing cluster configurations.

Name	Description
vpcSecurityGroupIds	The security group associated with the VPC
dbName	The name of the database created during the cluster provisioning process
clusterSubnetGroupName	The database subnet group (grouping of subnets) for the cluster
nodeType	Valid AWS Redshift node configuration name – these can be previous generation names
maximumNodes	The AWS limit for the given node configuration
storage	The total storage per node in storageUOM ("B", bytes) At least 0.90, recommended between 0.75 and 0.90; smaller for CPU intensive usage – this parameter skews the compute to storage ratio. The smaller the bias the more compute nodes. Valid values are greater than zero and less than or equal to 1.0
storageBias	

Administration - Commands

Navigate to Administration (admin users only) -> Commands. This list shows the registered command – synchronous actions executed by a client via curl or Postman or internally by the Lambda APIs themselves. The commands are described in subsequent sections.

Name	Description	Action Class	Enabled
CalculateRedshiftUsedStorage	Calculates used storage	com.informaticsiq.lambda.aws.redshift.actions.CalculateRedshiftUsedStorageAction	true
LoadRedshiftTable	Coordinates all instances (threads) of the LoadRedshiftTable Actions necessary to load a database	com.informaticsiq.lambda.aws.redshift.actions.LoadRedshiftClusterAction	true
EstimateRedshiftCost	Command used to estimate the cost of running a redshift instance over a period of time	com.informaticsiq.lambda.aws.redshift.actions.EstimateRedshiftCostAction	true
TerminateRedshiftCluster	Terminate an existing Redshift Cluster (snapshot (backup))	com.informaticsiq.lambda.aws.redshift.actions.TerminateRedshiftClusterAction	true
ConfigureRedshiftNode	Command used to configure Redshift Node	com.informaticsiq.lambda.aws.redshift.actions.ConfigureRedshiftNodeAction	true
CreateRedshiftCluster	CommandAction implementation used to create redshift cluster deployment	com.informaticsiq.lambda.aws.redshift.actions.CreateRedshiftClusterAction	true
ResizeRedshiftCluster	Resize an existing Redshift cluster	com.informaticsiq.lambda.aws.redshift.actions.ResizeRedshiftClusterAction	true
ResizeEMRCluster	Resize an existing EMR cluster (TASK & CORE Nodes)	com.informaticsiq.lambda.aws.emr.actions.ResizeEmrClusterAction	true
LoadRedshiftData	Coordinates all instances (threads) of the LoadRedshiftTable Actions necessary to load a database	com.informaticsiq.lambda.aws.redshift.actions.LoadRedshiftClusterAction	true

This interface allows administrators to configure commands that can be executed via the Lambda REST API interface.

The screenshot shows a web browser window titled "Lambda Management Con..." with the URL "https://gdapi.mcd.com/console/commands/edit". The page is titled "Command" and "General Information". It contains four input fields: "Identifier" (CalculateRedshiftUsedStorage), "Name" (CalculateRedshiftUsedStorage), "Description" (Calculates used storage for a given Redshift cluster), and "Action Class" (com.informaticsiq.lambda.aws.redshift.actions.CalculateRedshiftUsedStorageAction). Below these fields is a checkbox labeled "Enabled" which is checked. At the bottom right are "Cancel" and "Save" buttons. The browser's address bar also shows "Spark Master at spark://...".

Administration - Jobs

Navigate to Administration (admin users only) -> Jobs. This list shows the registered jobs – asynchronous actions executed by a client via curl or Postman or internally by the Lambda APIs themselves. The jobs are described in subsequent sections.

Lambda Management Console Dashboard ▾ On-Demand ▾ Administration ▾ My Account ▾

Jobs

New Job Actions ▾

Name	Description	Enabled
Test Job	Test Job	true
Load Redshift Table	Job to manage the loading of a redshift table	true
Resize EMR Cluster	Job to resize an EMR cluster - managed by Lambda	true
Resize EMR Cluster	Job to resize an EMR cluster	true
Terminate Redshift Cluster	Job to terminate a redshift cluster	true
CreateRedshiftCluster	Job to manage the process of creating redshift clusters	true
Resize Redshift Cluster	Job to resize a redshift cluster	true
Load Redshift Data	Job to manage the loading of the redshift cluster tables with one-time & incremental data corresponding to the subscriptions	true
Test Job 2	Test Job 2	true

Redshift
Datasources
Nodes
All Components
Commands
Jobs
Messages
Meters
Routing Rules
Pending User Requests
Publisher Management
API Account Management

This interface allows administrators to configure jobs that can be executed via the Lambda REST API interface.

Lambda Management Console Dashboard ▾ On-Demand ▾ Administration ▾

Job

General Information

Identifier	LoadRedshiftTable	
Command	Load Redshift Table	
Name	Load Redshift Table	
Description	Job to manage the loading of a redshift table for a given subscription	
Enabled	<input checked="" type="checkbox"/>	
		<input type="button" value="Cancel"/> <input type="button" value="Save"/>

Administration - Messages

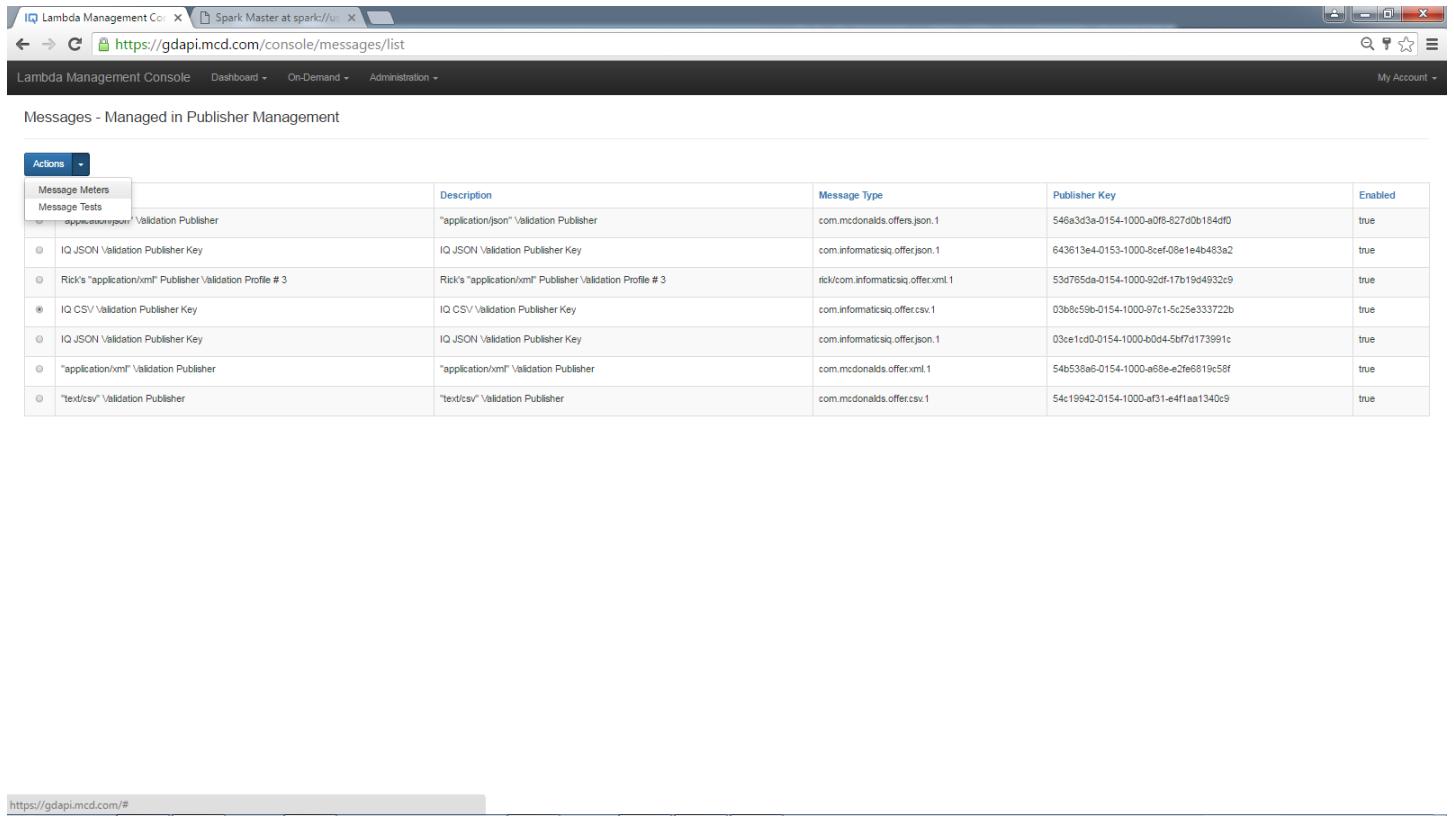
Navigate to Administration (admin users only) -> Messages. This list shows the defined message types. These messages are related to Publishers, described in a subsequent section. Publishers can only send a predefined message type for a given publisher key.

Name	Message Type	Publisher Key	Enabled
"application/json" Validation Publisher	com.mcdonalds.offers.json.1	546a3d3a-0154-1000-a0f8-827d0b184df0	true
IQ JSON Validation Publisher Key	com.informaticsiq.offer.json.1	643613e4-0153-1000-8cef-08e1e4b483a2	true
Rick's "application/xml" Publisher Validation Profile # 3	rick.com.informaticsiq.offer.xml.1	53d765da-0154-1000-92df-17b19d4932c9	true
IQ CSV Validation Publisher Key	com.informaticsiq.offer.csv.1	03b8c59b-0154-1000-97c1-5c25e333722b	true
IQ JSON Validation Publisher Key	com.informaticsiq.offer.json.1	03ce1cd0-0154-1000-b0d4-5bf7d173991c	true
"application/xml" Validation Publisher	com.mcdonalds.offer.xml.1	54b538a6-0154-1000-a68e-e2fe6819c58f	true
"text/csv" Validation Publisher	com.mcdonalds.offer.csv.1	54c19942-0154-1000-ef31-e4f1aa1340c9	true

<https://gdapi.mcd.com/console/messages/list>

Administration – Message Meters

Navigate to Administration (admin users only) -> Messages -> Message Meters.



The screenshot shows a web browser window with the URL <https://gdapi.mcd.com/console/messages/list>. The page title is "Messages - Managed in Publisher Management". The left sidebar has tabs for "Actions", "Message Meters", and "Message Tests". The "Message Meters" tab is selected. A table lists eight message meters, each with a description, message type, publisher key, and enabled status.

Message Meters	Description	Message Type	Publisher Key	Enabled
"application/json"\Validation Publisher	"application/json"\Validation Publisher	com.mcdonalds.offers.json.1	546a3d3a-0154-1000-a0f6-527d0b184df0	true
IQ JSON Validation Publisher Key	IQ JSON Validation Publisher Key	com.informaticsiq.offer.json.1	643613e4-0153-1000-8cef-08e1e4b483e2	true
Rick's "application/xml" Publisher Validation Profile # 3	Rick's "application/xml" Publisher\Validation Profile # 3	rick\com.informaticsiq.offer.xml.1	53d765da-0154-1000-92df-17b19d4932c9	true
IQ CSV\Validation Publisher Key	IQ CSV\Validation Publisher Key	com.informaticsiq.offer.csv.1	03b8c59b-0154-1000-97c1-5c25e333722b	true
IQ JSON Validation Publisher Key	IQ JSON Validation Publisher Key	com.informaticsiq.offer.json.1	03ce1cd0-0154-1000-b0d4-5b7fd173991c	true
"application/xml"\Validation Publisher	"application/xml"\Validation Publisher	com.mcdonalds.offer.xml.1	54b538a6-0154-1000-a68e-e2fe6819c58f	true
"text/csv"\Validation Publisher	"text/csv"\Validation Publisher	com.mcdonalds.offer.csv.1	54c19942-0154-1000-af31-e4f1aa1340c9	true

<https://gdapi.mcd.com/#>

This list shows the defined message meters – message metadata counters for a given window of time. In this case, a 5 minute aggregate (individual message metadata is stored in the MessageHistory table). These meters are used immediately and will begin counting all the defined message metadata from the time of definition until it is destroyed.

Name	Description	Period	Period Interval	Action
5 minute message meter	5 minute message meter	5	minute	Remove

Administration – Routing Rules

Navigate to Administration (admin users only) -> Routing Rules. The interface shows a list of routing rules – navigation paths for incoming event streams. These routes can act on individual events and large batch files (speed and batch layers).

Application	Message Type	Payload	Mapping Status	Message Targets
EventHistoryApp	test.pub.mapping.offers.1	true	Pending	cluster: lambda, topic: iq_lambda_text_classifier
EventHistoryApp	test.pub.mapping.offers.1	true	Pending	cluster: lambda, topic: mapping_offers
EventProcessorApp	test.pub.passthrough.v1.json	true	Assessed	cluster: lambda, topic: iq_lambda_transformed_offer
EventMeterApp	test.pub.mapping.offers.1	false	Pending	cluster: lambda, topic: mapping_redemptions

At a high-level, a route is defined for a given Spark application and message type, "...we have this event and we want to send it to these routes for processing." This is where the "pub/sub" relationship is defined. The rule also includes the status of the message – only send valid messages or only invalid message, etc. The status of the message tells the route to only consider messages that have never been processed. Finally, the message can be routed with its original payload or just the metadata (just want to know that the message occurred/arrived) as well as considering duplicate messages – by default the router drops duplicate payloads (via a 64-bit CRC).

The Message Targets is where the "sub" part of "pub/sub" are defined – there is no limit to the number of targets, or subscribers (physically implemented as Kafka topics). As the specific Spark application receives messages, processes them, the framework will automatically broadcast this application's output to all these targets.

Lambda Management Console Dashboard On-Demand Administration My Account ▾

Routing Rule Details

When the following rule conditions are true

Application:	Event History	<input type="button" value="i"/>
Message Type	InformaticsIQ	<input type="button" value="i"/>
Message Status	valid	<input type="button" value="i"/>
Mapping Status	pending	<input type="button" value="i"/>

Route With Payload Route Duplicates

Then route message to the these targets

Cluster	-- Select Cluster --	<input type="button" value="i"/>
Topic	Enter topic name	<input type="button" value="i"/>
Payload Data	<input type="checkbox"/>	
Transient Data:	<input type="checkbox"/>	

Add Target

Message Targets

Cluster	Topic	Payload Data	Transient Data	Action
lambda	iq_lambda_text_classifier	true	true	Remove

Cancel **Save**

Administration – Pending User Requests

Navigate to Administration (admin users only) -> Pending User Requests.

Welcome Lambda Administrator!

Service Health

- All services operating normally.
- Updated: May 20 2016 13:32:01 GMT-0500
- [Service Health Dashboard](#)

Additional Resources

- [Getting Started](#)
- Read our documentation or view our training to learn more about this site.

Pending User Requests

Name	Request Type	Status
John Ford	Market Request	Pending
Mark Muhs	Market Request	Pending

Where does my cluster actually live?

That depends on where you choose to create it. At present, we only support creating your cluster in the Amazon's North American East datacenter; however, we have plans to expand our offering to other Regions on an on-going basis.

What am I paying for?

For the Redshift (analytics database) cluster, you are paying for the cluster itself, intermediate storage used to load your cluster (on an as-needed basis), and your portion of the shared costs to host the environment.

<https://gdapi.mcd.com/console/useradmin/wizard>

This screen shows a list of pending user requests. To approve a user's request, verify that the markets requested are appropriate and check the checkbox next to their name.

User Account Approval

Step 1 of 2

1: User Approval List 2: Confirm Approvals

Select Role:

	Name	Registration Date	Requested Markets	Status
<input checked="" type="checkbox"/>	John Ford	03/09/2016 at 1:05PM	<input checked="" type="checkbox"/> Canada <input checked="" type="checkbox"/> US	Pending
<input type="checkbox"/>	Mark Muhs	03/24/2016 at 9:15AM	<input checked="" type="checkbox"/> US	Pending

Next, select the role from the dropdown list box, “Manager” is shown below and select “Approve” from the Action button.

User Account Approval

Step 1 of 2

1: User Approval List 2: Confirm Approvals

Manager

	Name	Registration Date	Requested Markets	Status
<input checked="" type="checkbox"/>	John Ford	03/09/2016 at 1:05PM	• Canada • US	Pending
<input type="checkbox"/>	Mark Muhs	03/24/2016 at 9:15AM	• US	Pending

Administration – Publisher Management

Navigate to Administration (admin users only) -> Publisher Management.

Publisher Name	Commands	Date	Publisher Key	Account Enabled	Account Status
"application/json" Validation Publisher	Jobs	at 10:18AM	546a3d3a-0154-1000-a0f6-827d0b184df0	true	Active
"application/xml" Validation Publisher	Messages	at 9:15PM	54b538a6-0154-1000-a68e-e2fe6819c58f	true	Active
"text/csv" Validation Publisher	Meters	at 8:40AM	54c19942-0154-1000-af31-e4f1aa1340c9	true	Active
IQ CSV Validation Publisher Key	Pending User Requests	at 6:18PM	03b8c59b-0154-1000-97c1-5c25e333722b	true	Active
IQ JSON Validation Publisher Key	Publisher Management	at 1:27PM	03ce1cd0-0154-1000-b0d4-5bf7d173991c	true	Active
IQ XML Validation Publisher Key	API Account Management	04/26/2016 at 6:27PM	03c5584c-0154-1000-be9e-1ff547db30e9	true	Active

<https://gdapi.mcd.com/console/publisher/list>

The publisher's profile is shown. The Publisher Key is dynamically generated and with the password, must be shared with the publisher. The contact and description fields should contain descriptive information since the typical publisher will be a system, not a human being. The password defined here is not related a Lambda Console user – publishers can be thought of as a special kind of [headless] user. Activate the user by checking the Enabled checkbox and changing the status to “Active.”

The Publisher Message Configuration defines the message format and type. These properties are used to validate the messages sent by the publisher. That is, if a publisher intends to send XML, but we defined their Content Type as text/csv then their messages will fail validation. This is done as a sanity check since any subsequent Processors assume that the messages they are receiving are “valid.” The Payload Type has been left as a free form text field to allow administrators the flexibility to define their own taxonomy. Currently, the message type should be defined as the reverse domain name of the publishers domain name plus the type of message, in this case “offer,” followed by the type (information only) and then the version number.

Lambda Management Cor X Spark Master at spark://us X

https://gdapi.mcd.com/console/publisher/profile

Lambda Management Console Dashboard On-Demand Administration

Publisher Profile

General Account Information

Publisher Key	03b8c59b-0154-1000-97c1-5c25e333722b
Contact First Name	Brenden
Contact Last Name	Cyze
Publisher Name	IQ CSV Validation Publisher Key
Publisher Description	IQ CSV Validation Publisher Key
Contact Email Address	brenden.cyze@us.mcd.com
Email Verification	Contact email address
Password	Profile password
Password Verification	Profile password

Enabled:	<input checked="" type="checkbox"/>
Status	Active

Publisher Message Configuration

Content Encoding	none
Content Type	text/csv
Default Process Mode	Transaction
Payload Type	com.informaticsiq.offer.csv.1

Administration – API Account Management

Navigate to Administration (admin users only) -> API Account Management.

API Account Name	API Key	Account Enabled	Account Status
IQ API Validation Key 1	ec8a487c-0153-1000-8fee-e6c18fa311b6	true	Active
GDW Support	72670d83-0154-1000-a1e8-e28b61240e74	true	Active
API Validation Account	5764d445-0154-1000-ec17-19596bb66c304	true	Active
Divyesh Pandya	c4488b30-0154-1000-80bd-ad0b1fb3d110	true	Active
IQ API Validation Key 2	f8c40edc-0153-1000-8f6d-aed9f731e5e8	true	Active
Christopher Schneider	c44d7f70-0154-1000-b849-149245e3b737	true	Active

The API account profile is shown below. This interface is very similar to the Publisher Key Management interface. The API Account Key is dynamically generated and with the password, must be shared with the API user. The contact and description fields should contain descriptive information since the API may be a system, not a human being. The password defined here is not related a Lambda Console user – API accounts can be thought of as a special kind of [headless] user. Activate the user by checking the Enabled checkbox and changing the status to “Active.”

The screenshot shows a web browser window with the URL <https://gdapi.mcd.com/console/api/profile>. The page title is "Lambda Management Console". The main content is titled "API Account Profile" and contains a form for "API Account Management". The form fields include:

API Account Key	72670d83-0154-1000-a1a8-e26b61240e74
Contact First Name	Suresh
Contact Last Name	Samuel
API Account Name	GDW Support
API Account Description	API Key used by GDW Support to run Lambda API Commands (e.g., ResizeEmrCluster, etc.)
Contact Email Address	Suresh.Samuel@us.mcd.com
Contact Email Verification	Contact email address
Password	Profile password
	Profile password

Below the form are two checkboxes: "Enabled" (checked) and "Status" (set to "Active"). At the bottom are "Cancel" and "Next" buttons.

Profile and Sign Out

Navigate to My Account -> Profile. This form shows the user's key contact information and the markets that they have access to.

lambda administrator's Profile Summary

Profile Information

First Name:

Last Name:

Email Address:

Markets Selected

Chosen Markets:

Profile

Sign Out

Edit

<https://gdapi.mcd.com/console/profile/view>

Navigate to My Account -> Sign Out. This will terminate your session, any asynchronous Jobs (creating a Redshift cluster) will continue to run.

Welcome lambda administrator!

Service Health

All services operating normally.
Updated: May 20 2018 13:38:27 GMT-0500
[Service Health Dashboard](#)

Additional Resources

[Getting Started](#)

Read our documentation or view our training to learn more about this site.

What is a dataset?

A dataset can be thought of as a data extract with properties or characteristics. For example, a dataset has a name, a size, and an ETA (time to load). You as a user have the ability to subscribe to as many datasets as you wish, just note that the size and window of time have a direct impact on the monthly cost.

Where does my cluster actually live?

That depends on where you choose to create it. At present, we only support creating your cluster in the Amazon's North American East datacenter; however, we have plans to expand our offering to other Regions on an on-going basis.

Can I request a one-time snapshot of data?

Absolutely! In fact the idea is that you can create and destroy as many clusters as you need. That means you can have one cluster that you may decide to keep and have others that come and go. In each case you have the option to load a one-time snapshot or an automatic refresh.

What am I paying for?

For the Redshift (analytics database) cluster, you are paying for the cluster itself, intermediate storage used to load your cluster (on an as-needed basis), and your portion of the shared costs to host the environment.

What if the performance isn't what I expected could I change my cluster?

Yes. During the creation process you are walked through a step-by-step wizard. You are presented with pricing for all the configurations that suite the datasets that you have selected. It is at that time that you commit to a give type of cluster. Once the cluster is running, you have the option to increase and decrease the size within limits. For example, you cannot decrease the size of your cluster beyond the minimum required to store the datasets you selected.

Profile

Sign Out

<https://gdapi.mcd.com/>

User Registration

Users gain access to the Lambda Console via opt-in, users have the ability to register for the platform using a self-service registration flow.

Registration

Step 1 of 4

1: Create Profile 2: Choose Markets 3: Confirm Information 4: Pending Approval

First Name	John
Last Name	Doe
Email Address	john.doe@noreply.email
Email Verification	john.doe@noreply.email
Password	*****
Password Verification	*****

Next

Next the user can select all the markets that they need access to by simply typing and adding all that apply to the list on the right-hand side.

Lambda Management Console

Registration

Step 2 of 4

1: Create Profile 2: Choose Markets 3: Confirm Information 4: Pending Approval

Markets

us
Australia
USA

< Remove Market

-- Markets Selected --
-- Select a market to add --

Previous Next

The user is prompted to confirm their selection.

Lambda Management Console

Registration

Step 3 of 4

1: Create Profile 2: Choose Markets 3: Confirm Information 4: Pending Approval

Profile Information

First Name: Jane

Last Name: Doe

Email Address: jdoe@test.com

Markets Selected

Chosen Markets: • USA

Previous **Confirm**

Finally the user is shown the notice. At this point their Member and MemberRelationships have been defined; however, their MemberRole and status have not been set – occurs during the user approval process described in the previous section.

Lambda Management Console

Registration

Step 4 of 4

1: Create Profile 2: Choose Markets 3: Confirm Information 4: Pending Approval

Thank you for registering. Your request will be reviewed within 3 business days. You will be notified when your account is ready.

Command Definitions

The following table summarizes the commands that have been defined.

Name	Description
CalculateRedshiftUsedStorage	Calculates used storage for a given Redshift cluster
ConfigureRedshiftNode	Command used to configure a Redshift node
CreateRedshiftCluster	CommandAction implementation used to create redshift cluster deployment

EstimateRedshiftCost	Command used to estimate the total cost of ownership of a redshift instance over a period of time
LoadRedshiftData	Coordinates all instances (threads) of the LoadRedshiftTable Actions necessary to load a database
LoadRedshiftTable	Coordinates all instances (threads) of the LoadRedshiftTable Actions necessary to load a database
ResizeEMRCluster	Resize an existing EMR cluster (TASK & CORE Nodes)
ResizeRedshiftCluster	Resize an existing Redshift cluster
TerminateRedshiftCluster	Terminate an exiting Redshift cluster without making a snapshot (backup)

Job Definitions

The following table summarizes the jobs that have been defined.

Name	Description
CreateRedshiftCluster	Job to manage the process of creating a Redshift cluster
LoadRedshiftData	Job to manage the process of loading data from s3 into a given Redshift cluster
LoadRedshiftTable	Job necessary to load a database
ResizeEmrCluster	Job to manage the process of resizing an existing EMR cluster
ResizeRedshiftCluster	Job to manage the process of resizing a given Redshift cluster
TerminateRedshiftCluster	Job to manage the process of terminating a given Redshift cluster

Operation

The following commands are used to start and stop the Lambda API server.

Start

SSH to the api node of interest as the ec2_user and run

```
screen -dMS api bash -c 'cd /usr/local/mcd/lambda-api-1.0 && sudo ./bin/lambda_api -Djdk.tls.ephemeralDHKeySize=2048 -Djdk.tls.rejectClientInitiatedRenegotiation=true -J-Xms512M -J-Xmx3g -J-server'
```

Stop

SSH to the api node of interest as the ec2_user and verify that an existing session exists (refer to the appendix of this document for a short Screen reference)

```
screen -ls
```

Connect to the existing session

```
screen -r api
```

Now shutdown the service by simply pressing the control-c combination, you can now exit your screen session by typing

```
exit
```

EMR

Cluster: EMR-CPROD-STATIC Waiting Cluster ready after last step completed.

Summary

- ID: j-MFWE4ZYFGDAY
- Creation Date: 2016-01-21 14:01 (UTC-5)
- Elapsed time: 119 days
- Auto-terminate: No
- Termination On Change protection:

Configuration Details

- Release label: emr-4.2.0
- Hadoop distribution: Amazon 2.6.0
- Applications: Hive 1.0.0, Pig 0.14.0, Oozie-Sandbox 4.2.0
- Log URI: s3://aws-logs-730817142111-us-east-1/elasticmapreduce/
- EMRFS consistent view: Enabled

Network and Hardware

- Availability zone: us-east-1b
- Subnet ID: subnet-01cf02a
- Master: Running 1 r3.4xlarge
- Core: Running 10 r3.4xlarge
- Task: Running 10 r3.4xlarge

Security and Access

- Key name: CDS-CPD
- EC2 instance profile: EMR-Prod-Master
- EMR role: EMR_DefaultRole
- Visible to all users: All Change
- Security groups for sg-31b2db48 (ElasticMapReduce-Master: master) More
- Security groups for sg-33b2db44 (ElasticMapReduce-Core & Task: slave) More

Instance Groups

ID	Name	Status	Type	Instance Type	Instance Count	EBS Volumes per Instance	Bid Price
ig-3NAX5TPPO8Y2R	US-EAST-CPROD_US-CDS-CPD1_EMR-WORKER2	Running	TASK	r3.4xlarge	10	Resize	0
ig-6846BKEETG8	US-EAST-CPROD_US-CDS-CPD1_EMR-MASTER1	Running	MASTER	r3.4xlarge	1	0	
ig-0Q4J7HP3K2	US-EAST-CPROD_US-CDS-CPD1_EMR-CORE1	Running	CORE	r3.2xlarge	1	Resize	0

Steps

Configurations

Resize

To resize an EMR cluster, the REST service requires the cluster id, the instance group id, the instance group type (as a safety mechanism), and the number of nodes for that combination – refer to the graphic above. The number of nodes represents the total number of nodes for that instance group. Therefore, a value of 10 means that once the nodes have been provisioned and marked as “Running” there will be 10 total. A subsequent request for 10 nodes would require the called to send in a total number of 20. To shrink the cluster use the same approach, for example a node count of 1 for the persistent EMR cluster will terminate 19 of the 20 provisioned previously. The following is an example of the REST call.

```
curl -X POST -H "Content-Type: application/json" -H "Authorization: Basic ZWM4Q4N2abc123y0xMDAwLThmZWUZjMThmYTMxeWI2OjEyMzx1Njc554" -d '{
  "clusterId": "j-MFWE4ZYFGDAY",
  "instanceGroupId": "ig-3NAX5TPPO8Y2R",
  "instanceGroupType": "task",
  "instanceGroupCount": "30"
}
' "https://gdapi.mcd.com/api/v1/jobs/ResizeEmrCluster/schedule"
```

Redshift

Lambda Management Co. | Spark Master at spark:// | Dashboard [Jenkins] | Redshift - AWS Console

<https://console.aws.amazon.com/redshift/home?region=us-east-1#cluster-details:cluster=us-east-cprod-us-cds-cpd1-redshift-195500019314-89820>

AWS Services Edit AWS-GG-Portal-Read/Brenden... N. Virginia Support

Redshift Dashboard

- Clusters
- Snapshots
- Security
- Parameter Groups
- Reserved Nodes
- Events
- Connect Client

Cluster: us-east-cprod-us-cds-cpd1-redshift-195500019314-89820 Configuration Status Performance Queries Loads Table restore

Cluster: us-east-cprod-us-cds-cpd1-redshift-195500019314-89820

Endpoint us-east-cprod-us-cds-cpd1-redshift-195500019314-89820.ciuv2sg591tg.us-east-1.redshift.amazonaws.com:5439 (authorized)

Cluster Properties

Cluster Name	us-east-cprod-us-cds-cpd1-redshift-195500019314-89820
Cluster Type	Multi Node
Node Type	dc1.large
Nodes	2
Zone	us-east-1b
Created Time	April 27, 2016 at 4:30:48 PM UTC-5
Cluster Version	1.0.1047
VPC ID	vpc-10e97079 (View VPCs)
Cluster Subnet Group	us-east-cprod-us-cds-cpd1
VPC Security Groups	US-CDS-CPD1_REDSHIFT_SG (sg-c87910b1) (active)
Cluster Parameter Group	default.redshift-1.0 (in-sync)

Cluster Status

Cluster Status	available
Database Health	healthy
In Maintenance Mode	no
Parameter Group Apply Status	in-sync
Pending Modified Values	None

Cluster Database Properties

Port	5439
Publicly Accessible	No
Database Name	ondemand
Master Username	admin
Encrypted	No
JDBC URL	jdbc:redshift://us-east-cprod-us-cds-cpd1-redshift-195500019314-89820.ciuv2sg591tg.us-east-1.redshift.amazonaws.com:5439/ondemand
ODBC URL	Driver{Amazon Redshift (x64)};Serverus-east-cprod-us-cds-cpd1-redshift-195500019314-89820.ciuv2sg591tg.us-east-1.redshift.amazonaws.com;Database=ondemand;UID=admin;PWD=insert_your_master_user_password_h

Backup, Audit Logging, and Maintenance

Automated Snapshot Retention Period	1
Cross-Region Snapshots Enabled	No
Audit Logging Enabled	No
Maintenance Window	sat:03:30-sat:04:00

Feedback English | © 2008 - 2016, Amazon Web Services, Inc. or its affiliates. All rights reserved. | Privacy Policy | Terms of Use

The screenshot shows the Lambda Management Console interface. At the top, there are tabs for 'New Environment' and 'Actions'. Below is a table of existing clusters:

	Name	Description	Type	Owner	Status
<input checked="" type="radio"/>	DLY SLS TEST3	▼ DLY SLS TEST3	redshift	rstrong@us.mcd.com	Ready
<input checked="" type="radio"/>	Test GBAL_LCAT	▼ Test GBAL_LCAT	redshift	rstrong@us.mcd.com	Ready
<input checked="" type="radio"/>	Sprinklr Insights	▲ Sprinklr Insights	redshift	rstrong@us.mcd.com	Ready
			Date Created:	04/27/2016	Total Storage: 288 GB
			GBL Code:	195500019314	Available Storage: 287 GB
			Estimated Monthly Cost:	\$446.00	Resource Id: organization mcd.market.us sprinklr_insights_7407d5c6-5a9b-45d4-a3d3-31b6bca68a2d
			Redshift Cluster Identifier:	us-east-cprod-us-cds-cpd1-redshift-195500019314-89820	Node Type: dc1.large
			Database Endpoint Address:	us-east-cprod-us-cds-cpd1-redshift-195500019314-89820.ciuv2sg591g.us-east-1.redshift.amazonaws.com	Node Count: 2
			Database Endpoint Port:	5439	Redshift Admin Id: admin
			Redshift Database Name:	ondemand	Redshift Password: K_f9-vy_5F2#=ro9O-p
<input checked="" type="radio"/>	A New Cluster	▼ A New Cluster	redshift	Rick.Jordan@us.mcd.com	Ready
<input checked="" type="radio"/>	Us Daily Sales Demo	▼ Us Daily Sales Demo	redshift	rstrong@us.mcd.com	Terminated

Resize

To resize a Redshift cluster, especially shrinking a cluster, that the cluster has enough space to store its data once the node has been removed. It is also advisable to take a snapshot of the cluster – note that snapshots increase the AWS costs. The number of nodes represents the total number of nodes for that cluster (worker nodes, this count excluded the “head” node). Therefore, a value of 10 means that once the nodes have been provisioned and marked as “Running” there will be 10 total. A subsequent request for 10 nodes would require the called to send in a total number of 20. To shrink the cluster use the same approach, but with a node count for the smaller cluster. The following is an example of the REST call.

```
curl -X POST -H "Authorization: Basic ZWM4Q4N2abc123y0xMDAwLThmZWUZjMThmYTMxeWI2OjEyMzx1Njc554" -H "Content-Type: application/json" -d ' {
  "resourceId": "organization|mcd.market.us|lili_sandbox",
  "resizeByNodeCount": "3"
}' "https://gdapi.mcd.com/api/v1/jobs/ResizeRedshiftCluster/schedule"
```

Terminate

It is recommended that you manually create a snapshot in the event of an erroneous termination request. The termination process is asynchronous; however, the status of the cluster is updated in real-time and viewable through the Lambda Management Console.

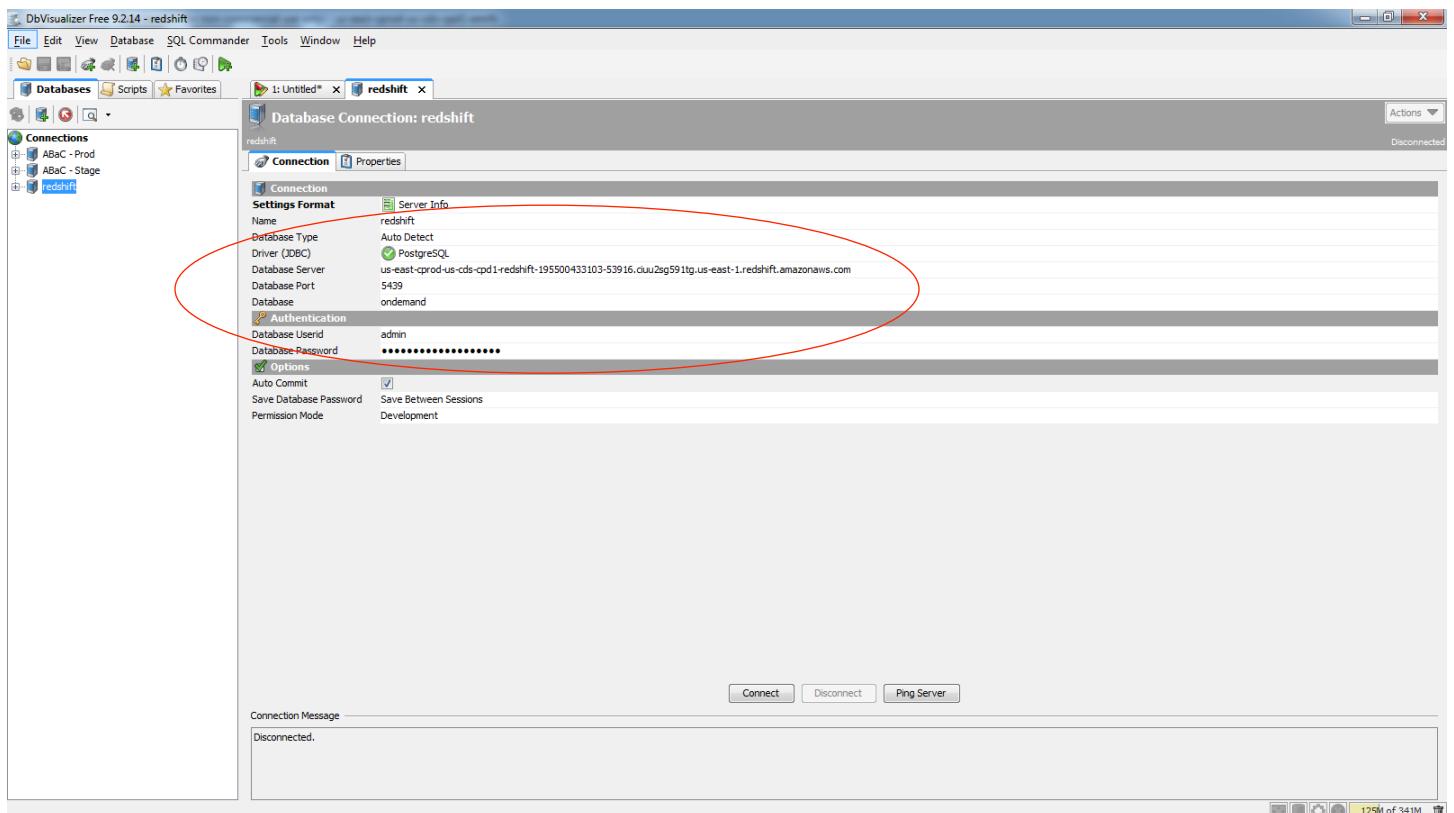
```

curl -X POST -H "Content-Type: application/json" -H "Authorization: Basic
ZWM4Q4N2abc123y0xMDAwLThmZWUZjMThmYTMxeWI2OjEyMzx1Njc554" -d '{
    "resourceId":"organization|lambda|datahub_validation"
}
' "https://gdapi.mcd.com/api/v1/jobs/TerminateRedshiftCluster/schedule"

```

Connecting to Redshift

Connecting to a Redshift cluster requires an ODBC driver; refer to the reference URL below for more details. The database server is the fully qualified “Database Endpoint Address” as shown above. The standard port for Redshift is 5439 – this value will not change. The database or sometimes referred to as database name is “ondemand” – this value will not change. The database user id is “admin” and the password is shown in the Lambda Management Console as “Redshift Password” as shown above. The following graphic shows an example of a typical configuration.



Reference: <http://docs.aws.amazon.com/redshift/latest/mgmt/configure-odbc-connection.html>

Appendix

The following sections contain supplementary material.

Data Lake Ingest

The primary entry point of the Lambda Architecture is via the https endpoints. Members (typically systems) that send data to the Lambda Architecture are referred to as “Publishers” and are assigned an API key that is designated for that individual organization and format.

Publisher Administration – Administrators add, change, and delete Publishers via the following user interface.

The screenshot shows a web-based configuration interface for a publisher profile. At the top, there are tabs for 'Connected Backup' (selected), 'spark://us...', 'Lambda Management Con...', and 'DynamoDB - AWS Cons...'. The URL in the address bar is <https://gdapi.mcd.com/console/publisher/profile>. The main header includes 'Lambda Management Console', 'Dashboard', 'On-Demand', 'Administration', and 'My Account'. The main content area is titled 'Publisher Profile' and contains the following fields:

- Publisher Description:** Validation API Publisher key
- Contact Email Address:** brenden.cyze@us.mcd.com
- Email Verification:** Contact email address
- Password:** Profile password
- Password Verification:** Profile password

Below these fields are two dropdown menus:

- Enabled:** A checked checkbox.
- Status:** A dropdown menu set to 'Active'.

Under the heading 'Publisher Message Configuration' are four more dropdown menus:

- Content Encoding:** none
- Content Type:** application/json
- Default Process Mode:** Transaction
- Payload Type:** com.informaticsiq.offer.json.1

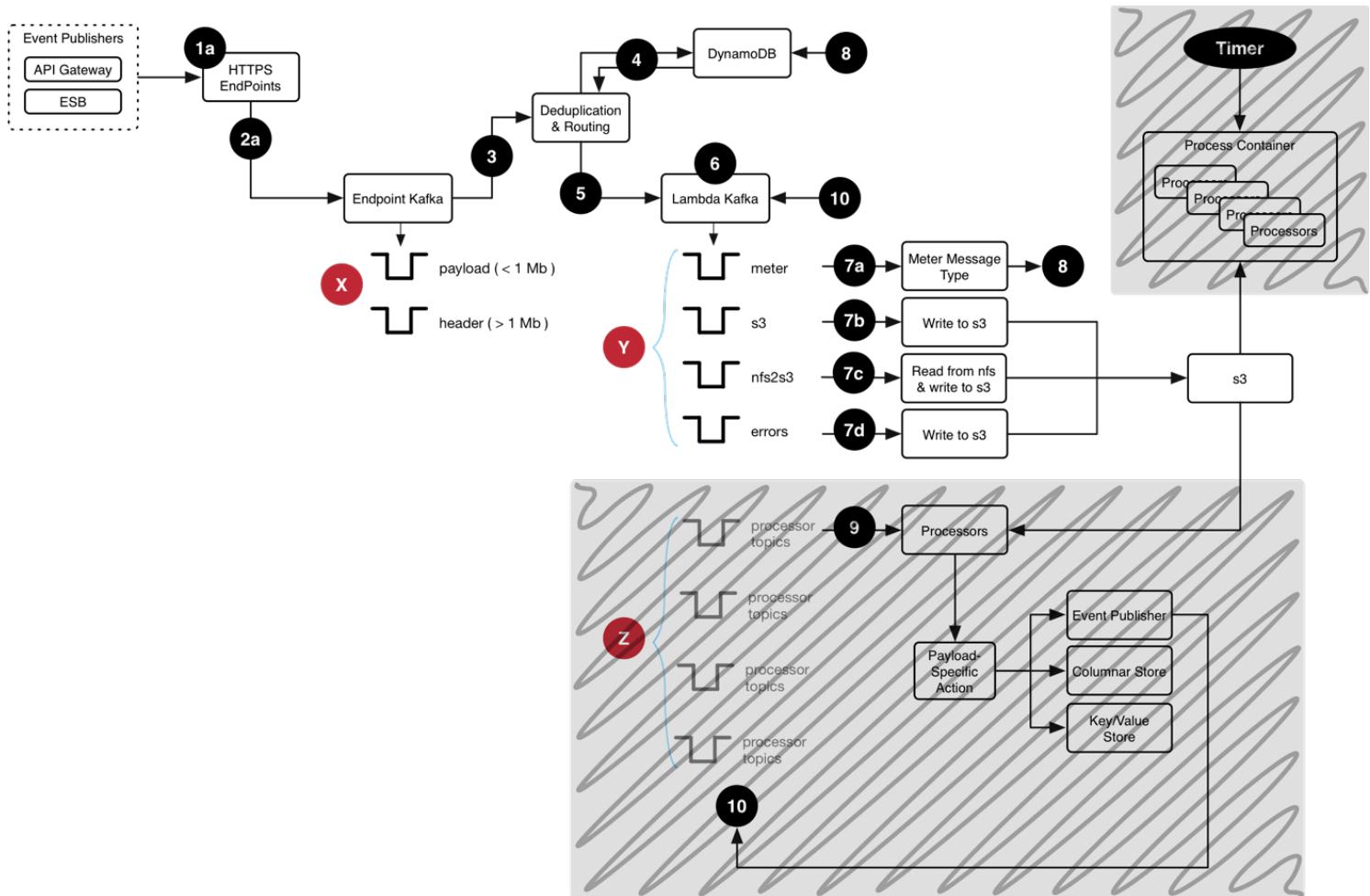
At the bottom right are 'Cancel' and 'Next' buttons.

The Publisher receives an API key and password. From there, that Publisher sends (publishes via http post) messages in the form of a single event, batches of transactions, or large files (in every case, these types must match the expected format as defined by the Publisher key) directly to the endpoint without any transformation, mapping, or formatting.

This diagram illustrates the various paths for small and large events. Publishers and consumers of events from publishers do not have to consume said events in real-time; however, the platform is event-based and the feedback from these events are relayed to subscribers in real-time. This means that a batch file send to the Endpoints will be processed in batch; however, the fact that that file was received, the disposition of the file, and the associated metadata (the fact that we received it – how many, what size, is it unique, etc.) are all made visible to said subscribers to allow them to take the appropriate action (batch or real-time, they are empowered to select the response and run-time performance that suites their individual use case) as the events are happening.

NOTE: The shaded areas are not applicable for the intent of this document

- 1a – Authenticated Publishers sending valid Events received
- 2a – Events marshaled into the Lambda Architecture
- 3 – Events sent to uniqueness checking and routing
- 4 – Event metadata (counters and volumetric) are sent to DynamoDB
- 5 – Routed Events are published to one or more topics for waiting subscribers
- 6 – Lambda Kafka brokers maintain write-once, read-many high-throughput Event consumption
- 7a – Spark Streaming application that acts as a generic “counter” of things – in this case counting and tabulating metadata about messages (every event is metered)
- 7b through 7d are mutually exclusive – its either a small event, large event, or an exception
- 7b – Spark Streaming application that write the Event (single or small payloads) to s3 as the immutable data store
- 7c - Spark Streaming application that write the Event (large payloads, files) to s3 as the immutable data store
- 7d - Spark Streaming application that write exceptions (malformed, invalid, etc) to s3 as the immutable data store



The given Spark Streaming applications – these applications run continuously and are the key part of the event propagation framework.

Spark Master at spark://usaecdmc1cm1.mcd.com:7077

URL: spark://usaecdmc1cm1.mcd.com:7077
 REST URL: spark://usaecdmc1cm1.mcd.com:6066 (cluster mode)

Alive Workers: 5

Cores in use: 260 Total, 52 Used
Memory in use: 70.0 GB Total, 20.0 GB Used
Applications: 4 Running, 29 Completed
Drivers: 0 Running, 0 Completed
Status: ALIVE

Workers

Worker Id	Address	State	Cores	Memory
worker-20160311014325-10.1.154.18-49406	10.1.154.18:49406	ALIVE	52 (10 Used)	14.0 GB (4.0 GB Used)
worker-20160311014325-10.1.154.31-43162	10.1.154.31:43162	ALIVE	52 (10 Used)	14.0 GB (4.0 GB Used)
worker-20160311014325-10.1.154.4-58184	10.1.154.4:58184	ALIVE	52 (11 Used)	14.0 GB (4.0 GB Used)
worker-20160311014325-10.1.154.56-54528	10.1.154.56:54528	ALIVE	52 (11 Used)	14.0 GB (4.0 GB Used)
worker-20160311014325-10.1.154.61-35046	10.1.154.61:35046	ALIVE	52 (10 Used)	14.0 GB (4.0 GB Used)

Running Applications

Application ID	Name	Cores	Memory per Node	Submitted Time	User	State	Duration
app-20160316095750-0032	(kill) FileMoverApp	10	1024.0 MB	2016/03/16 09:57:50	lambdaops	RUNNING	6.2 h
app-20160316093750-0031	(kill) EventWriterApp	14	1024.0 MB	2016/03/16 09:37:50	lambdaops	RUNNING	6.5 h
app-20160314175734-0014	(kill) EventMeterApp	14	1024.0 MB	2016/03/14 17:57:34	lambdaops	RUNNING	46.2 h
app-20160314175346-0011	(kill) EventHistoryApp	14	1024.0 MB	2016/03/14 17:53:46	lambdaops	RUNNING	46.2 h

The following is an example of three different types of events sent directly to the Endpoints using PostMan.

Small JSON message simulating a single event

The screenshot shows the Postman application interface. On the left, there's a sidebar with tabs for 'History' and 'Collections'. Under 'Collections', there are two items: 'mcd.prod - API Commands & Jobs' (5 requests) and 'mcd.prod - Endpoint Tests' (3 requests). Below these are three POST requests: 'HttpEndpoint : JSON Request < 1 KB', 'HttpEndpoint : JSON Request > 1 KB and < 1 MB', and 'HttpEndpoint : JSON Request > 1 MB'. The main area is titled 'HttpEndpoint : JSON Request < 1 KB'. It shows a POST request to 'https://gdmsg.mcdonalds.com/iot/v1/in'. The 'Body' tab is selected, showing raw JSON data:

```
1 [{}  
2 "request": {"param1": "xx", "param2": "yy", "param3": "zz"},  
3 "response": {"code": 200}  
4 ]
```

The Spark Applications log the receipt of the Event (large or small – that is, Speed or Batch)

8081/logPage/?appId=app-20160316095750-0032&executorId=3&logType=stdout

stdout log page for app-20160316095750-0032/3

```
2614 of 12614
09:58:44.033 Executor task launch worker-1 WARN TransferFileAction: Starting local to remote filesystem copy from /lambda/shunt/us-east-1/7fb898f1-0153-1000-a767-a52e55dde902.dat to lambda/mcd/643613e4-0153-1000-8cef-08e1e4b483a2/tmp/coalesce/com.informaticsiq.offer.json.1/passed_lint/dt/2016-03-16/7fb898f1-0153-1000-a767-a52e55dde902_event.json
09:58:44.034 Executor task launch worker-1 WARN TransferFileAction: s3 bucket: 'us-east-cprod-us-cds-cpd1-transient'
09:58:44.034 Executor task launch worker-1 WARN TransferFileAction: targetHdfsPath: 'lambda/mcd/643613e4-0153-1000-8cef-08e1e4b483a2/tmp/coalesce/com.informaticsiq.offer.json.1/passed_lint/dt/2016-03-16/7fb898f1-0153-1000-a767-a52e55dde902_event.json'
09:58:44.034 Executor task launch worker-1 WARN TransferFileAction: sourceNfsPath: '/lambda/shunt/us-east-1/7fb898f1-0153-1000-a767-a52e55dde902.dat'
09:58:44.036 Executor task launch worker-1 WARN TransferFileAction: Source file length: '2082661'
09:58:44.213 Executor task launch worker-1 WARN TransferFileAction: ETag 72cb865db20672ede8fee96cace038c1 for lambda/mcd/643613e4-0153-1000-a767-a52e55dde902_event.json
09:58:44.214 Executor task launch worker-1 WARN TransferFileAction: File written with expiration 'null': lambda/mcd/643613e4-0153-1000-8cef-08e1e4b483a2/tmp/coalesce/com.informaticsiq.offer.json.1/passed_lint/dt/2016-03-16/7fb898f1-0153-1000-a767-a52e55dde902_event.json
09:58:44.251 Executor task launch worker-1 WARN TransferFileAction: File exists on s3 at 'lambda/mcd/643613e4-0153-1000-8cef-08e1e4b483a2/tmp/coalesce/com.informaticsiq.offer.json.1/passed_lint/dt/2016-03-16/7fb898f1-0153-1000-a767-a52e55dde902_event.json' content length : 2082661
09:58:44.251 Executor task launch worker-1 WARN TransferFileAction: ultimateParentPathRoot: /lambda/shunt/us-east-1
09:58:44.251 Executor task launch worker-1 WARN TransferFileAction: archivePathRoot: /lambda/shunt/us-east-1/archive/2016-03-16-13-58/com.informaticsiq.offer.json.1
09:58:44.251 Executor task launch worker-1 WARN TransferFileAction: archivePath: /lambda/shunt/us-east-1/archive/2016-03-16-13-58/com.informaticsiq.offer.json.1/7fb898f1-0153-1000-a767-a52e55dde902.dat_1458136724251
09:58:44.253 Executor task launch worker-1 WARN TransferFileAction: Moving file /lambda/shunt/us-east-1/7fb898f1-0153-1000-a767-a52e55dde902.dat_1458136724251 to /lambda/shunt/us-east-1/archive/2016-03-16-13-58/com.informaticsiq.offer.json.1/7fb898f1-0153-1000-a767-a52e55dde902.dat
09:58:44.261 Executor task launch worker-1 WARN TransferFileAction: Writing message from Kafka to remote filesystem lambda/mcd/643613e4-0153-1000-8cef-08e1e4b483a2/tmp/coalesce/com.informaticsiq.offer.json.1/passed_lint/event/2016-03-16/7fb898f1-0153-1000-a767-a52e55dde902_event.json
09:58:44.261 Executor task launch worker-1 WARN TransferFileAction: s3 bucket: 'us-east-cprod-us-cds-cpd1-transient'
09:58:44.261 Executor task launch worker-1 WARN TransferFileAction: targetHdfsPath: 'lambda/mcd/643613e4-0153-1000-8cef-08e1e4b483a2/tmp/coalesce/com.informaticsiq.offer.json.1/passed_lint/dt/2016-03-16/7fb898f1-0153-1000-a767-a52e55dde902_event.json'
```

The particular information about where the Event and associated file ended-up on s3 – every file is cataloged in a standard multi-tenant enabled directory structure

stdout log page for app-20160316095750-0032/3

Back to Master

Previous 0 B Bytes 0 - 12614 of 12614

```
16/03/16 09:58:44.033 Executor task launch worker-1 WARN TransferFileAction: Starting local to remote filesystem copy from /lambda/shunt/us-east-1/7fb898f1-0153-1000-a767-a52e55dde902.dat to lambda/mcd/643613e4-0153-1000-8cef-08e1e4b483a2/tmp/coalesce/com.informaticsiq.offer.json.1/passed_lint/dt/2016-03-16/7fb898f1-0153-1000-a767-a52e55dde902_event.json
16/03/16 09:58:44.034 Executor task launch worker-1 WARN TransferFileAction: s3 bucket: 'us-east-cprod-us-cds-cpd1-transient'
16/03/16 09:58:44.034 Executor task launch worker-1 WARN TransferFileAction: targetHdfsPath: 'lambda/mcd/643613e4-0153-1000-8cef-08e1e4b483a2/tmp/coalesce/com.informaticsiq.offer.json.1/passed_lint/dt/2016-03-16/7fb898f1-0153-1000-a767-a52e55dde902_event.json'
16/03/16 09:58:44.034 Executor task launch worker-1 WARN TransferFileAction: sourceNfsPath: '/lambda/shunt/us-east-1/7fb898f1-0153-1000-a767-a52e55dde902.dat'
16/03/16 09:58:44.036 Executor task launch worker-1 WARN TransferFileAction: Source file length: '2082661'
16/03/16 09:58:44.213 Executor task launch worker-1 WARN TransferFileAction: ETag 72cb865db20672ede8fee96cace038c1 for lambda/mcd/643613e4-0153-1000-8cef-08e1e4b483a2/tmp/coalesce/com.informaticsiq.offer.json.1/passed_lint/dt/2016-03-16/7fb898f1-0153-1000-a767-a52e55dde902_event.json
16/03/16 09:58:44.214 Executor task launch worker-1 WARN TransferFileAction: File written with expiration 'null': lambda/mcd/643613e4-0153-1000-8cef-08e1e4b483a2/tmp/coalesce/com.informaticsiq.offer.json.1/passed_lint/dt/2016-03-16/7fb898f1-0153-1000-a767-a52e55dde902_event.json
16/03/16 09:58:44.251 Executor task launch worker-1 WARN TransferFileAction: File exists on s3 at 'lambda/mcd/643613e4-0153-1000-8cef-08e1e4b483a2/tmp/coalesce/com.informaticsiq.offer.json.1/passed_lint/dt/2016-03-16/7fb898f1-0153-1000-a767-a52e55dde902_event.json' content length : 2082661
16/03/16 09:58:44.251 Executor task launch worker-1 WARN TransferFileAction: ultimateParentPathRoot: /lambda/shunt/us-east-1
16/03/16 09:58:44.251 Executor task launch worker-1 WARN TransferFileAction: archivePathRoot: /lambda/shunt/us-east-1/archive/2016-03-16-13-58/com.informaticsiq.offer.json.1
16/03/16 09:58:44.251 Executor task launch worker-1 WARN TransferFileAction: archivePath: /lambda/shunt/us-east-1/archive/2016-03-16-13-58/com.informaticsiq.offer.json.1/7fb898f1-0153-1000-a767-a52e55dde902.dat_1458136724251
```

Using the AWS Console, the given Event was persisted in s3 as a file, JSON in particular.

The screenshot shows the AWS S3 Management Console interface. The URL is https://console.aws.amazon.com/s3/home?region=us-east-1#bucket=us-east-cprod-us-cds-cpd1-transient&prefix=lambda/mcd/643613e4-0153-1000-8cef-08e1e4b483a2/tmp/coalesce/com.informaticsiq.offer.json.1/passed_int/event/2016-03-16. The page displays a table of files with columns: Name, Storage Class, Size, and Last Modified. The files listed are various event.json files from March 16, 2016.

Name	Storage Class	Size	Last Modified
7dateabe-0153-1000-b51a-e8ef0d519e56_event.json	Standard	785 bytes	Tue Mar 15 23:23:24 GMT-500 2016
7db09c03-0153-1000-854c-851a084013_event.json	Standard	793 bytes	Tue Mar 15 23:30:42 GMT-500 2016
7db8223-0153-1000-9491-030348639460_event.json	Standard	777 bytes	Wed Mar 16 00:00:05 GMT-500 2016
7dcbe77-0153-1000-a093-4063518a8082_event.json	Standard	1.2 KB	Wed Mar 16 00:03:11 GMT-500 2016
7dcc0aee-0153-1000-90e7-6463a1c2f545_event.json	Standard	1.2 KB	Wed Mar 16 00:04:40 GMT-500 2016
7436997-0153-1000-9994-0725fb98bf_event.json	Standard	1.2 KB	Wed Mar 16 00:50:40 GMT-500 2016
7910e48-0153-1000-83d440704a326ce_event.json	Standard	778 bytes	Wed Mar 16 08:15:20 GMT-500 2016
7948d33-0153-1000-9623-4d9981201f_event.json	Standard	780 bytes	Wed Mar 16 08:18:18 GMT-500 2016
79d553d-0153-1000-b30c-9875a5a9a999_event.json	Standard	782 bytes	Wed Mar 16 08:29:22 GMT-500 2016
7fa50eab-0153-1000-8e1b-58cd69164b19_event.json	Standard	791 bytes	Wed Mar 16 08:38:05 GMT-500 2016
7fa5d5f-0153-1000-9291-3cd0268b563_event.json	Standard	792 bytes	Wed Mar 16 08:38:13 GMT-500 2016
7fa818ed-0153-1000-01f5-0ffba505712_event.json	Standard	1.2 KB	Wed Mar 16 08:38:57 GMT-500 2016
7fb6092-0153-1000-ab97-a98199da9431_event.json	Standard	1.2 KB	Wed Mar 16 08:58:05 GMT-500 2016
7fb6044-0153-1000-8925-646c27221b1e_event.json	Standard	794 bytes	Wed Mar 16 08:58:26 GMT-500 2016
7fb898f1-0153-1000-a767-a52655d9e902_event.json	Standard	849 bytes	Wed Mar 16 08:58:45 GMT-500 2016
7f558d3-0153-1000-aa85-58294d30c1e_event.json	Standard	849 bytes	Wed Mar 16 10:05:05 GMT-500 2016

For large messages (anything greater than 1Mb (configurable, but should not exceed 2mb)) the payload is stored in its raw format in a separate directory (“dat”). Subsequent Processors can consume these files as they arrive or in batch and send them to other Processors or act as Event Publishers and send the data to a data store such as Redshift, DynamoDB, etc.

The screenshot shows the AWS S3 Management Console interface. The URL is https://console.aws.amazon.com/s3/home?region=us-east-1#bucket=us-east-cprod-us-cds-cpd1-transient&prefix=lambda/mcd/643613e4-0153-1000-8cef-08e1e4b483a2/tmp/coalesce/com.informaticsiq.offer.json.1/passed_int/dat/2016-03-16. The page displays a table of files with columns: Name, Storage Class, Size, and Last Modified. The files listed are two dat files from March 16, 2016.

Name	Storage Class	Size	Last Modified
7b689f1-0153-1000-a767-a52655d9e902_event.json	Standard	1.9 MB	Wed Mar 16 08:58:45 GMT-500 2016
7f558d3-0153-1000-aa85-58294d30c1e_event.json	Standard	1.9 MB	Wed Mar 16 10:05:05 GMT-500 2016

The following screenshot illustrates a sample of the wide range of metadata collected about a given Publisher’s data, again all stored within DynamoDB.

The screenshot shows the AWS DynamoDB AWS Console interface. The URL is https://console.aws.amazon.com/dynamodb/home?region=us-east-1#tables:selected=cprod_us_cds.MeterValues. The page displays a table named "cprod_us_cds.MeterValues". A modal window titled "Edit item" is open, showing a JSON representation of a single item. The JSON object contains numerous fields related to message processing metrics, such as enabled, end date, entity ID, entity type, last update date, meter ID, meter value ID, and various byte counts and message counts.

```

1 <!
2   "enabled": true,
3   "endDate": 1458103200000,
4   "entityId": "organization|lambda|com.informaticsiq.offer.json.1",
5   "entityType": "cprod_us_cds.Messages",
6   "lastUpdate": 1458102641502,
7   "meterId": "10-minute-message-meter|organization|lambda",
8   "meterValueId": "10-minute-message-meter|organization|lambda|organization|lambda|com.informaticsiq.offer.json.1|cprod_us_cds.Messages|1458102600000|1458103200000",
9   "meterValues": {
10     "badBytes": 0,
11     "badMessages": 0,
12     "duplicateBytes": 0,
13     "duplicateMessages": 0,
14     "expiredBytes": 0,
15     "expiredMessages": 0,
16     "incompleteBytes": 0,
17     "incompleteMessages": 0,
18     "recoveredBytes": 0,
19     "recoveredMessages": 1,
20     "totalBytes": 103,
21     "totalMessages": 1,
22     "unauthorizedBytes": 0,
23     "unauthorizedMessages": 0,
24     "validBytes": 103,
25     "validMessages": 0
26   },
27   "startDate": 1458102600000
28 }

```

Medium JSON message simulating a many small events (aggregated) or one large event

The entire path of the event has been omitted for brevity, see also #1 of 3 above for more details.

The screenshot shows the Postman application interface. The left sidebar displays a history of requests, including "mcd.prod - API Commands & Jobs" and "mcd.prod - Endpoint Tests". The main area shows a request titled "HttpEndpoint : JSON Request > 1 KB and < 1 MB" with a POST method. The "Body" tab is selected, showing a JSON payload with line numbers. The payload is as follows:

```

1 {"MobileRedemption": [
2   "redemptionEventDateUTC": 1430182718000,
3   "LocationName": "Oak Brook 22nd St. McCopCo #0000",
4   "CampaignID": "00000000000000937627",
5   "DeviceID": "9fa32bb0-47ad-84a5-0ca9-99a372cba85bb",
6   "PromotedItems": [
7     {
8       "LineItems": [
9         {
10           "ItemDescription": "Egg McMuffin",
11           "CategoryCode": "BREAKFEST",
12           "Price": 1.99,
13           "Quantity": 1,
14           "LineItemNumber": 1
15         },
16         {
17           "ItemDescription": "Lg Coffee",
18           "CategoryCode": "BREAKFEST",
19           "Price": 0.89,
20           "Quantity": 1,
21           "LineItemNumber": 2
22       }
23     ]
24   }
25 ]
26 }

```

Below the body, the "Cookies", "Headers (5)", and "Tests" tabs are visible. At the bottom, the status is shown as "Status: 200 OK Time: 247 ms".

Large JSON message simulating thousands of small events (aggregated) or one very large event (such as a single file)

The entire path of the event has been omitted for brevity, see also #1 of 3 above for more details.

The screenshot shows the Postman application interface. The left sidebar displays collections and environments, with 'mcd.prod - API Commands & Jobs' selected. The main area shows a POST request for 'HttpEndpoint : JSON Request > 1 MB' to the URL <https://gdmsg.mcdonalds.com/iot/v1/in>. The 'Body' tab is active, showing two examples of JSON payloads. The first payload (labeled 1) is a large object containing various parameters like validationAction, activityMap, user_input_type_id, page_url, and session_context_id. The second payload (labeled 2) is similar but includes a different timestamp and some additional fields. The 'Send' button is visible at the top right of the request panel.

AWS Resource Tagging

All tagable AWS Resources are tagged with GBL and contact information along with the McDonald's standard metadata tags. The following screen shot is an example of the fields used for tagging the Redshift resource itself.

1: Create Cluster 2: Configure Data Sources 3: Confirm Configuration 4: Deploy Cluster

Cluster:

Node Configuration

Description	Monthly Cost	
AWS ds2.xlarge moderate dense storage instance	\$1,514.00	1

Data Subscriptions

Name	Daily Refresh	Date Range
US TDA+ (1 months)	enabled	02/16/2016 - 03/16/2016
US Datahub (1 months)	enabled	02/16/2016 - 03/16/2016
Total Required Storage:		

Billing Details

GBL Code	1212121212	
Contact Email	cluster_owner_email@here.com	
<input checked="" type="checkbox"/> I understand that all applicable hosting, storage, and other charges will begin immediately		

The following is an example of the tags applied to an existing Redshift cluster (AWS Console).

The screenshot shows the AWS Redshift console with the 'Clusters' page open. On the left, there's a sidebar with options like Redshift Dashboard, Clusters, Snapshots, Security, Parameter Groups, Reserved Nodes, Events, and Connect Client. The 'Clusters' tab is selected. In the main area, there's a table with columns: Cluster, Cluster Status, DB Host, and Actions. Several clusters are listed, including 'us-east-cprod-us-cds-cpd1-redshift-1510-us-east-1a', 'us-east-cprod-us-cds-cpd1-redshift-195500433103-0372', 'us-east-cprod-us-cds-cpd1-redshift-195500433103-4478', 'us-east-cprod-us-cds-cpd1-redshift-195500433103-6416', 'us-gsw-prd1', and 'us-gsw-slg1'. One cluster, 'us-east-cprod-us-cds-cpd1-redshift-195500433103-6416', is highlighted with a blue selection bar.

To the right, a modal dialog titled 'Manage Tags' is open. It contains a section for 'Applied Tags' and a section for 'Add Tags'.

Applied Tags:

Key	Value	Delete
Environment	1	
Name	us-east-cprod-us-cds-cpd1...	
Owner	incident/brenden.cyze@us...	
DataClassification	Sensitive	
Market	us	
GBL	195500433103	
Version	2016.03.01.001	

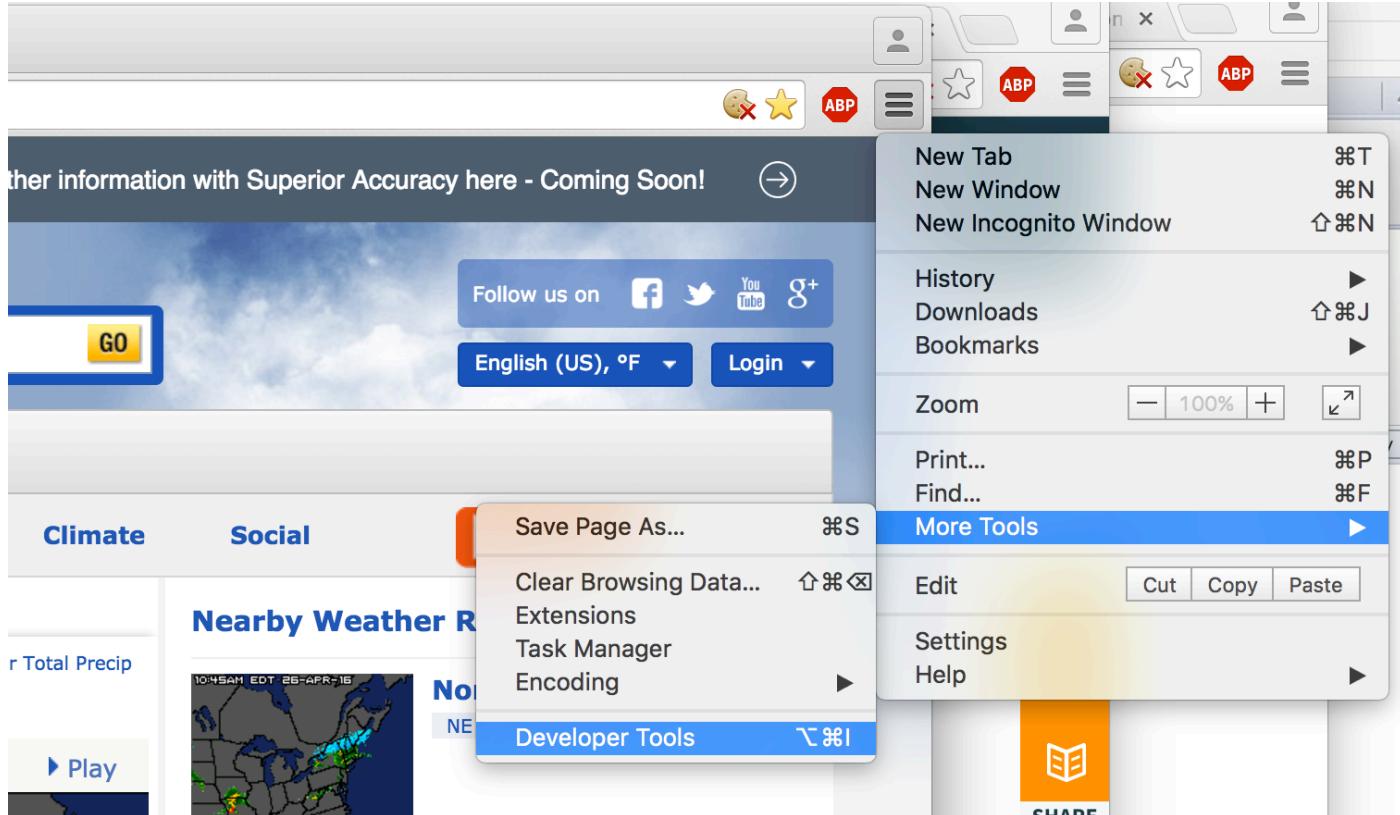
Add Tags:

Key	Value
[Add key]	Empty value

At the bottom right of the dialog are 'Cancel' and 'Apply Changes' buttons.

Exporting JavaScript Logs

To cut down on the size of the log file, please restart Chrome and head directly to the Lambda Console. Once there, click on the right hand side quick menu selector -> “More Tools” -> “Developer Tools”



The following window should appear as follows – note this window may appear on the left or right side of your browser.

United States Doppler Weather
www.accuweather.com/en/us/national/weather-radar

Preview our new website look, enhanced readability, and expanded features for weather information with Superior Accuracy here - Coming Soon!

AccuWeather.com for New York, NY

United States WEATHER | New York, NY LOCAL WEATHER | ALLERGIES | FEATURED

Home | Radar & Maps | News & Video | Severe Weather | Climate | Social | WATCH VIDEOS

Radar | Satellite | Severe | Forecast Maps | Weather Maps

Weather Radar | Interactive | Radar & Satellite | Past 24 Hour Total Precip

United States Weather Radar

Nearby Weather Radars | Northeast Regional

Elements Console Sources Network Timeline Profiles Resources Security Audits Adblock Plus

top ▾ Preserve log

```

✖ Failed to load resource: net::ERR_BLOCKED_BY_CLIENT http://www.googleadservices.com/pagead/conversion.js
✖ Failed to load resource: net::ERR_BLOCKED_BY_CLIENT http://partner.googleadservices.com/gpt/pubads_impl_85.js
▶ AdManager VM131:1
✖ Failed to load resource: net::ERR_BLOCKED_BY_CLIENT http://vortex.accuweather.com/adc2010/ads/js/showads-3.js
✖ Failed to load https://www.google.com/ads/ga-audiences?v=1&aip=1&t=sr_r=4&tid=UA-20804842-1&cid=161847720.1459915810&jid=898775084&_v=j41&z=520706458
resource: net::ERR_BLOCKED_BY_CLIENT
  
```

If the “Console” tab is not selected by default, click it to show the java script console as shown below.

Radar | Satellite | Severe | Forecast | Weather Radar | United States Weather Radar

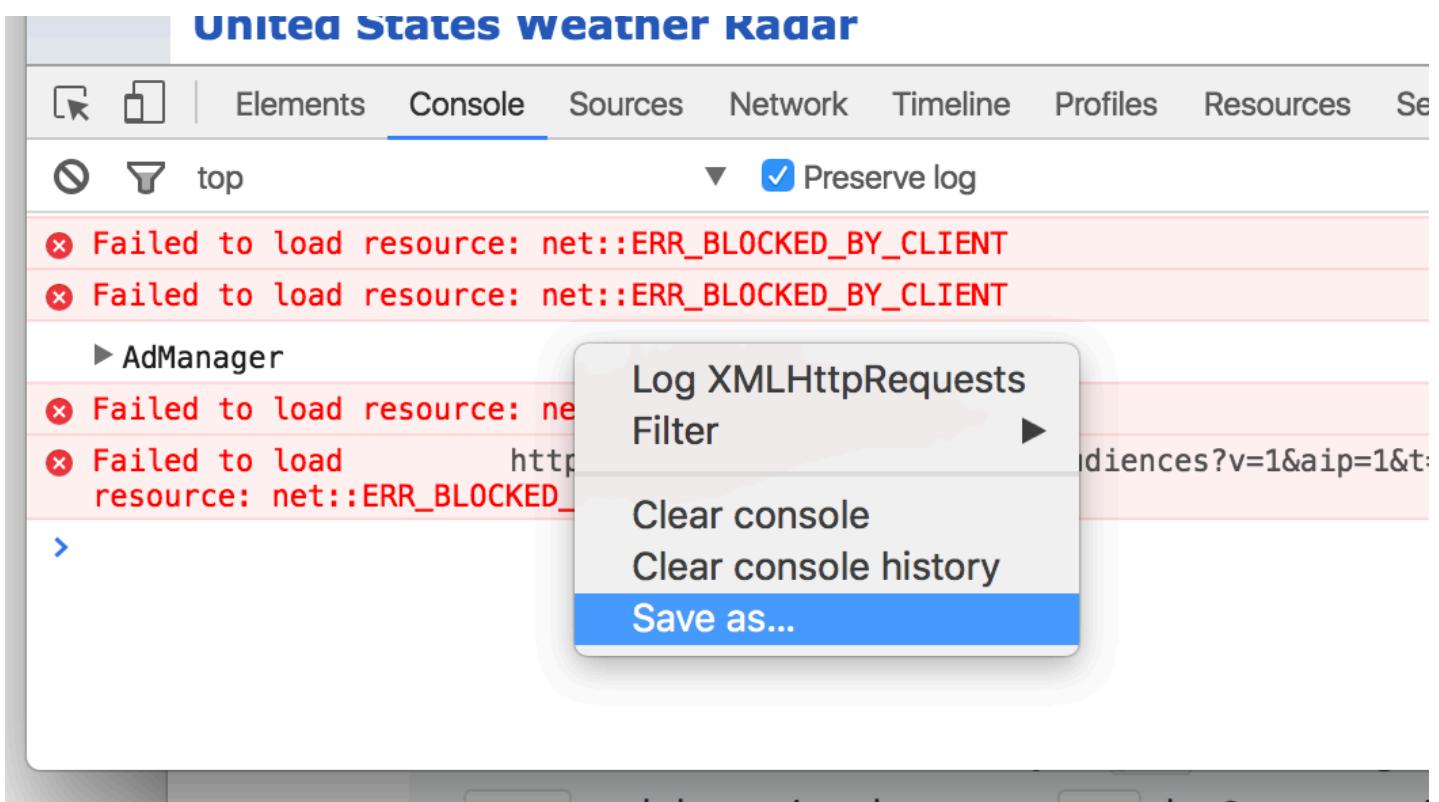
Elements Console Sources Network Timeline

top ▾ Preserve log

```

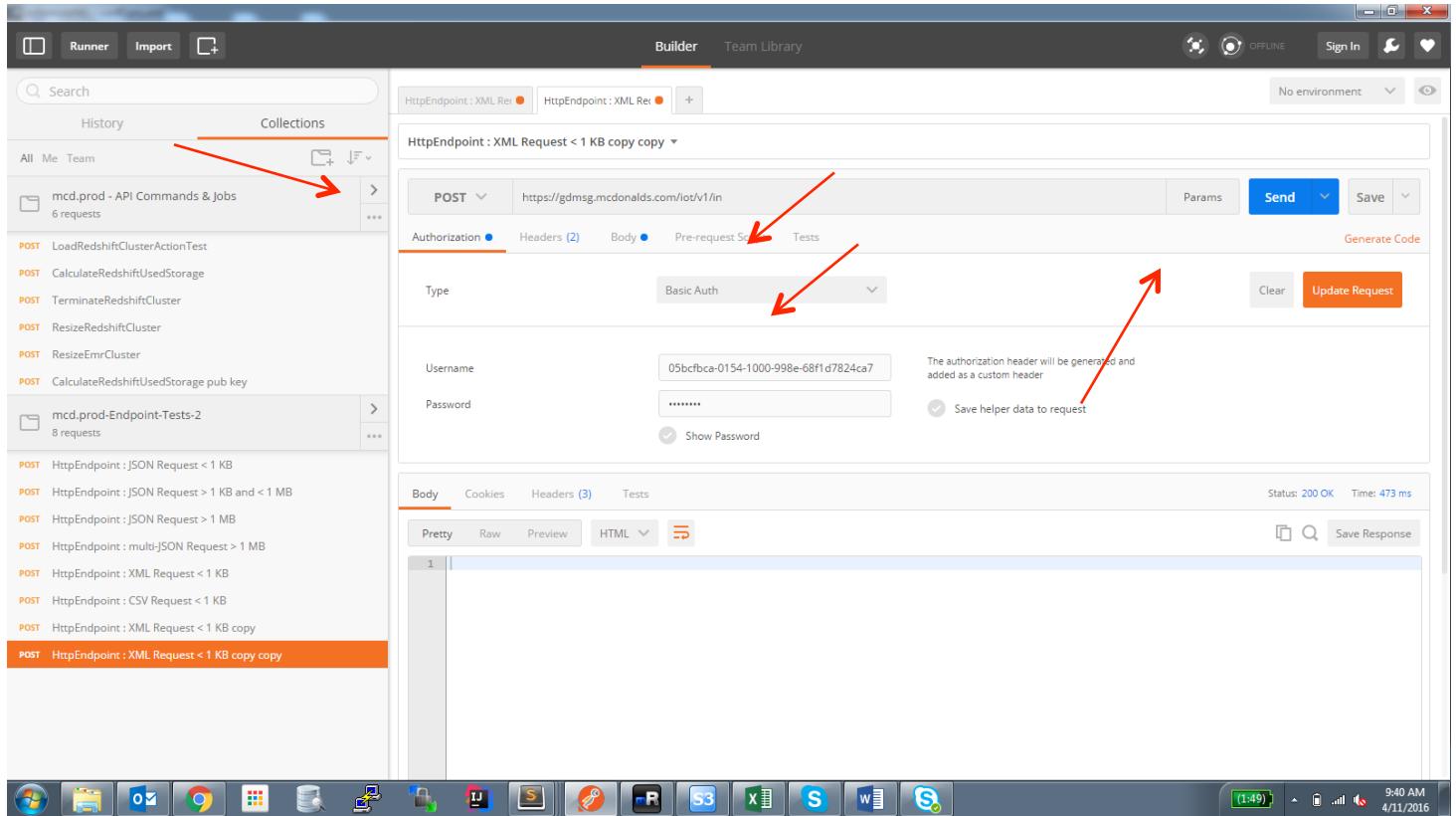
✖ Failed to load resource: net::ERR_BLOCKED_BY_CLIENT
✖ Failed to load resource: net::ERR_BLOCKED_BY_CLIENT
▶ AdManager
✖ Failed to load resource: net::ERR_BLOCKED_BY_CLIENT
✖ Failed to load https://www.google.com/ads/ga-audiences?v=1&aip=1&t=sr_r=4&tid=UA-20804842-1&cid=161847720.1459915810&jid=898775084&_v=j41&z=520706458
resource: net::ERR_BLOCKED_BY_CLIENT
  
```

Right-mouse within the console and select “Save as...” as follows – this will export the raw console output to a file. Please send this file so that we can investigate its contents.

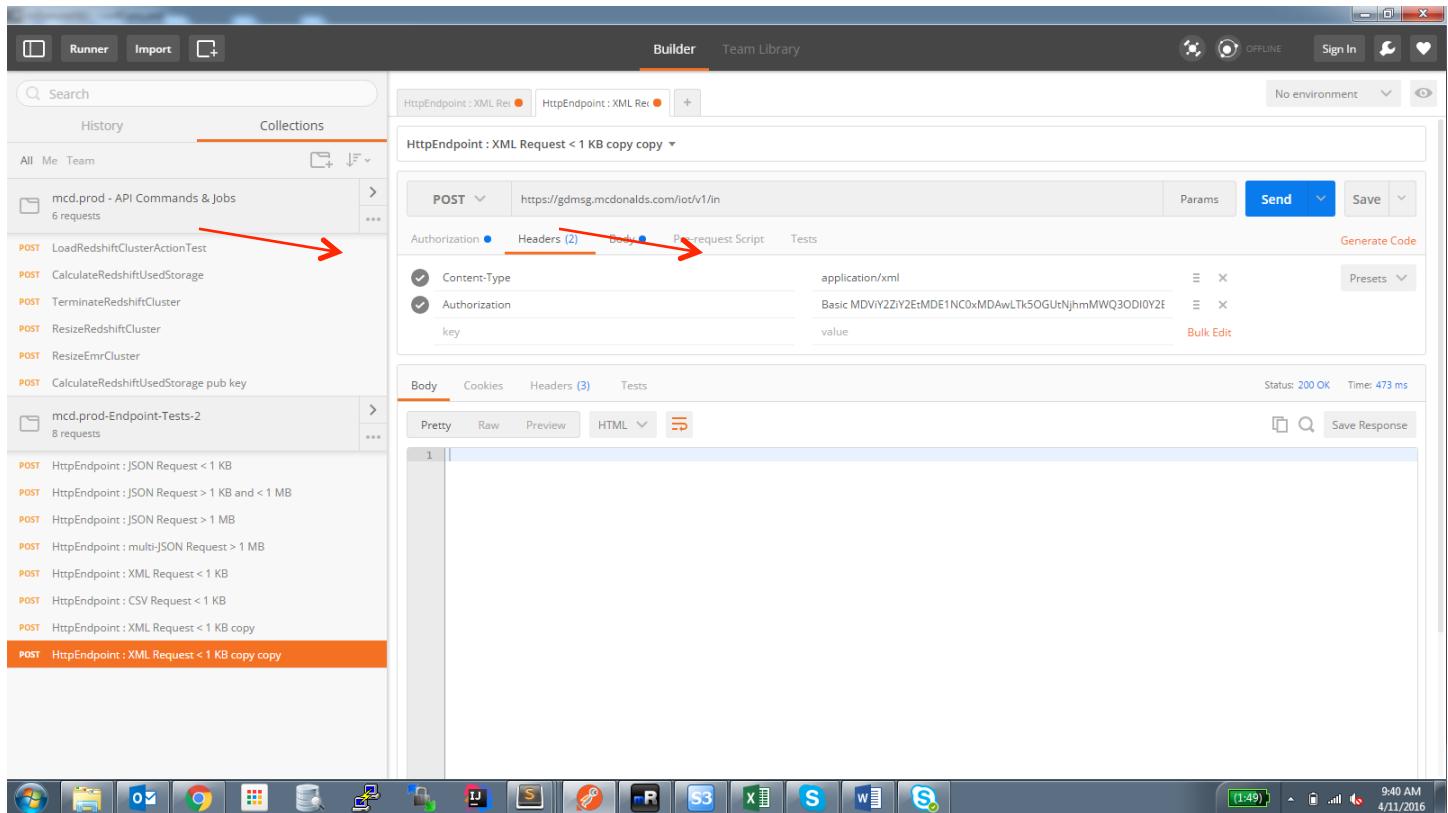


Postman Authorization Setup

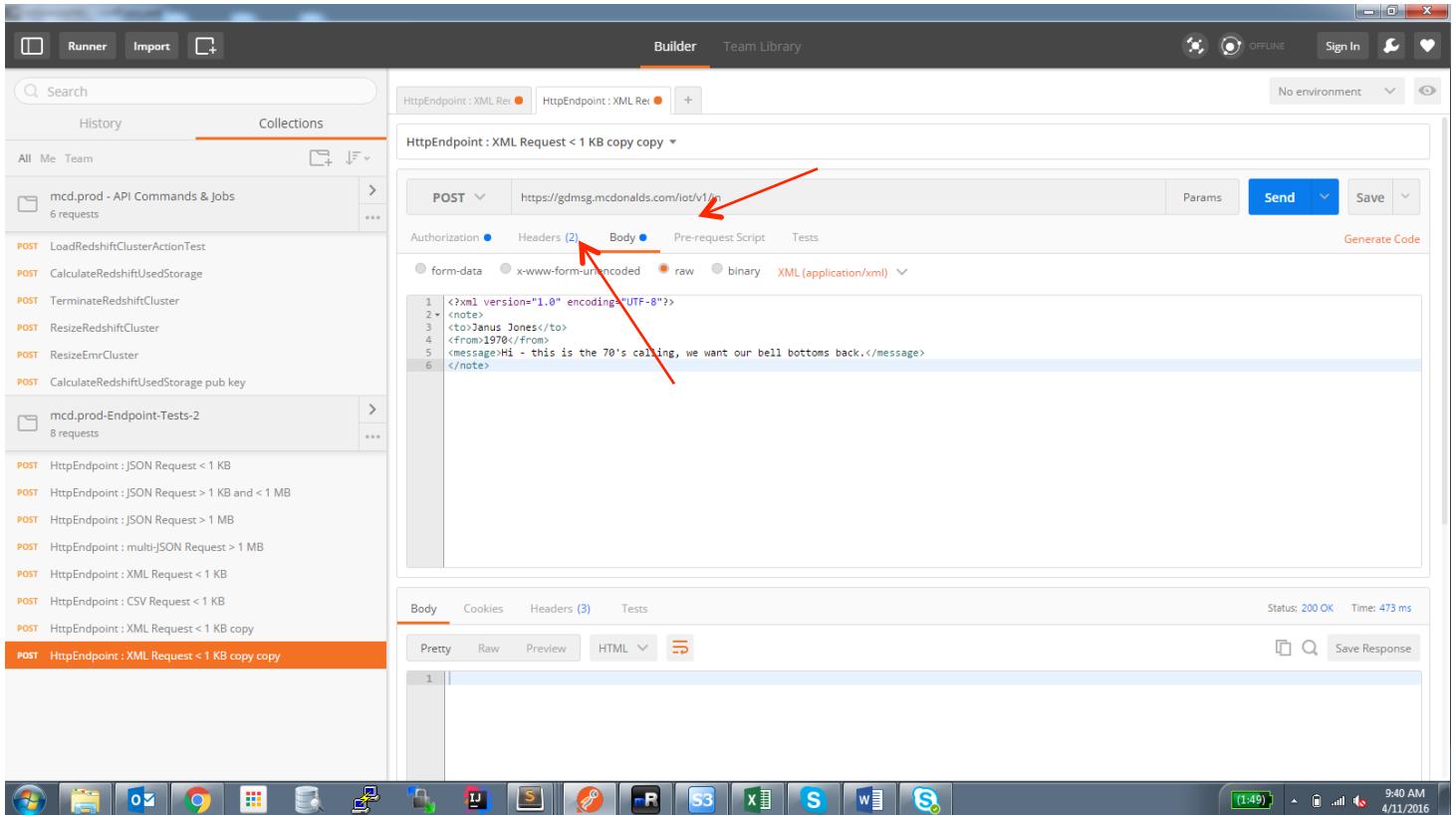
To verify your PostMan authorization, begin by selecting the configuration in question and clicking on the "Authorization" tab as indicated below. From the dropdown list called "Type" choose "Basic Auth" - you will be presented with a "Username" and a "Password" field. For the Username, enter the Pub Key associated with this message type and for the Password enter the password specified while creating the Pub Key. Finally, click the "Update Request" button (this will create an authorization token that will appear in the "Headers" tab next).



Once you have updated the request header, you should notice a header element called “Authorization” with a value that begins with “Basic” blank space and then your encoded username (pub key) and password as noted below. The body of the request – in this case XML – is specified by selecting the “Content-Type” of “application/xml” from the type-a-head.



The body of the request – in this case XML – is specified clicking the “raw” radio button and picking “XML (application/xml)” from the dropdown and shown below.



You are in a position to create an XML message and click the “Send” button to post the message to the EndPoint.

Screen Command Reference

Screen Home Page: <http://www.gnu.org/software/screen/>

Action	Screen Key Sequence
List active screen sessions	screen -ls
Reattach to an existing session by name	screen -r <session name>
Create a new session	screen -S <session name>
Detach (leave active) from a session	control a d
To close a session that is no longer needed	exit

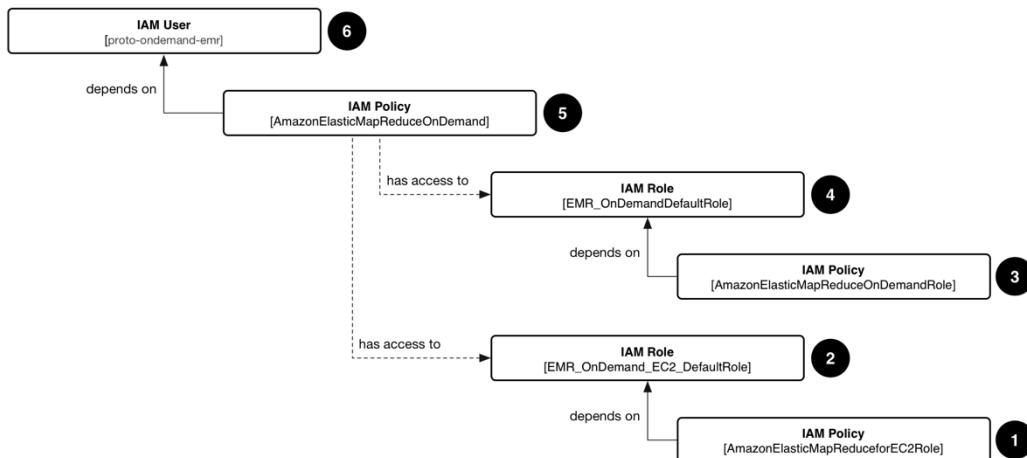
IAM Permission Recommendation

This second outlines the general approach to IAM policies to be used for the “on-demand” components of the Lambda Architecture. This section does not intend to describe in exhaustive detail all the permissions, but rather demonstrate the efficacy of specific changes and how that impact ones ability to [and not to] instantiate and manage EMR (or Redshift) clusters.

Preconditions

This approach assumes that the subnet, security groups, roles (service and job flow), and key-pairs (aka pem key) already exist. This allows us to remove privileges related to these AWS Resources and Actions outright and focus on constraining any remaining Actions necessary to create and maintain a given EMR Cluster.

The following diagram illustrates the relationships between the various IAM entities involved. The table that follows describes each in more detail.



As mentioned previously, these policies specifically address the need to constrain where a cluster can be created – specific subnet - and what roles are bestowed to and by the IAM User involved with creating the EMR Cluster. This means that Actions such as ec2:TerminateInstance (and any other for that matter) can and will be constrained as necessary during a subsequent exercise.

#	Artifact	Name	Description	Attachment
1	IAM Policy	AmazonElasticMapReduceforEC2Role	This is the default policy with no changes - acts as a stub or placeholder for subsequent revisions such as locking down specific s3 Actions or the like – used once the EMR cluster is operational (for activities such as changing the size of the cluster)	AmazonElasticMapReduceforEC2Role.txt
2	IAM Role	EMR_OnDemand_EC2_DefaultRole	This role is attached to a single policy, #1 above	
3	IAM Policy	AmazonElasticMapReduceOnDemandRole	This policy constrains ec2 actions of any kind to a given subnet and removes the "ec2:AuthorizeSecurityGroupIngress", "ec2:DeleteTags", "ec2>CreateSecurityGroup", "ec2:ModifyImageAttribute" Actions. Likewise, this policy limits to a given set of security groups, pem key, and limits the iam:PassRole to #2 and #4 explicitly - used once the EMR cluster is operational	AmazonElasticMapReduceOnDemandRole.txt
4	IAM Role	EMR_OnDemandDefaultRole	This role is attached to a single policy, #3	

5	IAM Policy	AmazonElasticMapReduceOnDemand	Similar in nature to IAM Policy #3, this is a modified version of the default policy (AmazonElasticMapReduce) that removes the "ec2:AuthorizeSecurityGroupIngress", "ec2:DeleteTags", "ec2>CreateSecurityGroup", "ec2:ModifyImageAttribute" Actions and constrained the "iam:PassRole" Action to the two roles mentioned above – used by the user in #6 to create the cluster	AmazonElasticMapReduceOnDemand.txt
6	IAM User	proto-on-demand-emr	The identity used to create the EMR cluster - this user is attached to only one policy, "AmazonElasticMapReduceOnDemand"	

Run-time

The following is an abbreviated portion of the source code found in the EMRProvisioner.scala file. With the exception of the hard-coded values (e.g. subnets, roles, etc.) this code is completely representative of the application. At run-time all of these values are fixed – in fact, the only things that will change are the tag values, s3 Bucket Name, the number/type of instances, and the name of the cluster itself.

```

val request : RunJobFlowRequest = new RunJobFlowRequest()
    .withName(buildClusterInstanceName)

    // Standard root naming will allow for simple policy amendments to constrain access even further
    .withLogUri(s"s3://$buildClusterInstanceName")

    // Must exist, will use this same resource for all on-demand emr clusters
    .withServiceRole("EMR_OnDemandDefaultRole") // Role from #4 above

    // Must exist, will use this same resource for all on-demand emr clusters
    .withJobFlowRole("EMR_OnDemand_EC2_DefaultRole") // Role from #2 above

    // . . . (Removed for clarity)

    .withInstances(new JobFlowInstancesConfig()
        // Must exist, will use this same resource for all on-demand emr clusters
        .withEc2KeyName("proto-cds-automation") // Purpose built key pair exclusively for on-demand emr

        .withHadoopVersion("2.6.0")
        .withInstanceCount(2)

        // Must exist, will use this same resource for all on-demand emr clusters
        .withEc2SubnetId("subnet-0da9999a") // Subnet from IAM Policies above

        // Must exist, will use this same resource for all on-demand emr clusters
        .withEmrManagedMasterSecurityGroup("sg-9999af20") // Standard EMR Master security group

        // Must exist, will use this same resource for all on-demand emr clusters
        .withEmrManagedSlaveSecurityGroup("sg-9999af22") // Standard EMR Slave security group

        .withKeepJobFlowAliveWhenNoSteps(true)
        .withTerminationProtected(false)
        .withMasterInstanceType("r3.2xlarge")
        .withSlaveInstanceType("r3.2xlarge"))

```