

# **Data Pipeline- 6.5**



Сс	onte	nts	
1.	Intro	oduction	6
2	l.1.	Introducing Data Pipeline	6
1	L.2.	Overview	6
1	L.3.	Target Audience	6
2.	Sup	ported Web Browsers	6
ź	2.1.	Unsupported Browsers	6
3.	Terr	minology	7
4.	Logg	ging In	9
5.	Hon	nepage Menu Panel	11
5	5.1.	List Pipeline	12
	5.1.	1. Sharing a Pipeline	13
5	5.2.	Create a Pipeline	14
5	5.3.	List Components	15
5	5.4.	Delete Pods	17
5	5.5.	Scheduler	17
5	5.6.	Data Channel	
5	5.7.	Settings	
6.	Use	r Interface	
6	5.1.	Creating a Pipeline	
	6.1.	1. Resource Allocation	21
e	5.2.	Adding Components to the Canvas	21
6	5.3.	Toggle Log Panel	25
e	5.4.	Search Component in Pipelines	26
6	5.5.	Global Variable	28
6	5.6.	Deleting a Pipeline	29
6	5.7.	Pipeline Monitoring	29
6	5.8.	Connecting Components	
	6.8.3	1. Event Creation	
	6.8.2	2. Updating an Event	32
	6.8.3	3. Connecting Event to a Component	33
	6.8.4	4. Shared Event	34
	6.8.	5. Preview Data	35
	6.8.	6. Flush Event	
7.	Com	nponent Pallet	



-	7.1.	Sorting and Grouping Components	37
-	7.2.	Searching a Component	38
8.	Com	ponent Properties	39
8	8.1.	Basic Information Tab	40
8	8.2.	Meta Information Tab	40
8	8.3.	Configuration and Logs Tabs	42
9.	Com	ponents	44
9	9.1.	Reader	44
	9.1.1	S3 Reader	44
	9.1.2	P. HDFS Reader	48
	9.1.3	8. Cassandra Reader	52
	9.1.4	RDBMS Reader	53
	9.1.5	5. ES Reader	56
	9.1.6	5. SFTP Reader	58
	9.1.7	7. GCS Reader	60
	9.1.8	3. SFTP Stream Reader	63
	9.1.9	9. MongoDB Reader	65
	9.1.1	.0. Notification Subscriber	67
	9.1.1	1. Azure Reader	74
	9.1.1	2. Azure Cosmos DB Reader	77
	9.1.1	13. Azure Metadata Reader	79
	9.1.1	4. Azure Blob Reader	81
0	9.2.	Writer	85
	9.2.1	S3 Writer	85
	9.2.2	2. RDBMS Writer	87
	9.2.3	B. HDFS Writer	89
	9.2.4	l. Cassandra Writer	91
	9.2.5	5. ES Writer	93
	9.2.6	6. MongoDB Writer	94
	9.2.7	7. Notification Publisher	97
	9.2.8	3. Azure Writer	04
	9.2.9		
	9.2.1		
	9.2.1		
0	9.3.	Transformation1	10



<ul> <li>9.3.2. Date Formatter Component</li></ul>	115 117 119 121 124 127 130
9.3.4. SQL Component 9.3.5. Dataprep Script Runner Component	117 119 121 124 127 130
9.3.5. Dataprep Script Runner Component	119 121 124 127 130
	121 124 127 130
9.3.6. File Splitter	124 127 130
	127 130
9.3.7. Python Script (Custom Python Script)	130
9.3.8. Rule Splitter	
9.3.9. Stored Procedure Runner	400
9.3.10. Stream Component	132
9.3.11. Flatten JSON	134
9.3.12. Email Component	136
9.4. ML	138
9.4.1. R Model Runner	138
9.4.2. Python Model Runner	140
9.5. Ingestion	141
9.5.1. API Server Ingestion	142
9.5.2. Sqoop Executer	145
9.5.3. GCS Monitor	146
9.5.4. AWS SNS Monitor	150
9.5.5. Script Runner	155
9.5.6. MQTT Consumer	158
9.5.7. API Connector	159
9.5.8. SFTP Python Monitor	161
9.5.9. OPC UA	164
9.5.10. Eventhub Subscriber	165
9.5.11. Twitter Scrapper	167
9.5.12. API Ingestion	169
9.6. WebSocket	170
9.6.1. WebSocket Listener	170
9.6.2. WebSocket Producer	172
9.7. Scheduler	173
10. Settings	174
10.1. Kafka Configuration	175
10.2. Dataprep Scripts	
10.3. Data Science Models	



10.3.1.	Using a Deployed Model in Data Pipeline	. 177
10.4.	Logger	. 178
10.5.	Default Configuration	. 178
11. Thir	d-Party Data Ingestion using Data Pipeline	. 179
11.1.	Data Ingestion from an MQTT Consumer	. 180
11.2.	Configuring a Data Pipeline According to the Script	. 180
11.2.1.	Making a POST request along with the data	. 182
12. Sign	ing Out	. 183
12.1.	Forgot Password Option	. 184
12.2.	Force Login	. 185



### **1. Introduction**

#### 1.1. Introducing Data Pipeline

The Data Pipeline is an Enterprise data orchestration and transformation tool, that allows you to seamlessly design and deploy your DataOps or MLOps workflow. It can handle both Streaming and batch data seamlessly. The Data pipeline offers an extensive list of data processing components that help you automate the entire data workflow, Ingestion, transformations, and running AI/ML models.

In the Data Pipeline plugin, we treat data as events, Data Processing components can listen to events, as data hits those events, the process kick starts automatically, these processes then publish the output to another event. This allows data engineers to chain the process and build large data flows.

#### 1.2. Overview

This guide covers:

- Features and menus provided on the Data Pipeline homepage
- Steps to create and manage Pipeline
- Create and manage various pipeline components

#### 1.3. Target Audience

The document is targeted at the following audience:

- Data Engineers
- Data Scientists
- ML-Ops Engineers

### **2.** Supported Web Browsers

The BDB Platform is a web browser-based application. The users can run the BDB Platform and its various plugins on the below-given versions of the browsers:

Google Chrome	Latest Version (recommended web browser)
Mozilla Firefox/ Firefox ESR	Latest Version
Microsoft Edge	Latest Version
Apple Safari	10

The supported browser versions are driven by the capabilities the UI employs and the dependencies it uses. UI features will be developed and tested against the supported browsers.

#### **2.1. Unsupported Browsers**

While the UI may run successfully in unsupported browsers, it is not actively tested against them. Additionally, the UI is designed as a desktop experience and is not currently supported in mobile browsers.



### 3. Terminology

Component Deployment: Component Deployment means the Deployment on Kubernetes.

Components (Deployment Type): There are two types of component categories in context to deployment-type (Spark and Docker).

Spark: Drivers and Executers.

When these components get deployed, we get two pods one is for the component driver and another one is the executor. The number of executors can be increased according to the processing load or the throughput required by the user.

■ Component	List			Search	Component
<b>↑</b> =	Name Deployment 1	ype Component Type	Version	Component IsExposed Group	Actions
+	Split Component docker	system	6.0.0-R3	Transformation •	Ο
	Date Formatter spark	system	6.0.0-R3	Transformation •	Ο
Ţ	Join Component spark	system	6.0.0-R3	Transformation •	Ο
x	R Model docker	system	6.0.0-R3	ML	Ο
3	API Server docker Ingestion	system	6.0.0-R3	Ingestion	Ο
0	Websocket docker Listener docker	system	6.0.0-R3	Websocket •	Ο
\$	Websocket	a istam	6 0 0 D2	Wahrackat	•
¢	daskar	erete m		er page: 10  11 - 20 of 56	

Docker: a single pod gets deployed in this case.

### Base Components

These are the components that are responsible for the backend functionality of the pipeline. Each of these components' Pod should be UP and Running for the smooth functioning of the pipeline module. Base components are as given below:

- event-trigger
- log manager
- pipeline-scheduler
- task invoker
- task manager
- webservice
- ws-producer
- pipeline-scheduler

### Workflow Editor

The canvas where we create/develop the pipeline is the workflow editor.



Pipeline Editor > Sample Pipe	line 🖉 💷 💷 😳 💷 😳 🔁 🗐	•
Search Component	Private Shared	
System Custom	Fvents	+
Reader 🔹		
Writer 🔹		
Transformation 🔹		
ML 🝷		
Ingestion 🔹		
Websocket 🔹		
Scheduler 👻		

The user can open the Pipeline Editor page by clicking the '**Pipeline Workflow Editor**' icon provided on the top right side of the Pipeline List page.

<b>BBB</b>						۹ 🏭 🖓 🖪
≡ Pip	peline List				Search	۹ 🛛 🗖
♠	Name	Resources	Created Date 🕹	Status	Action	• ts_3 Pipeline Workflow Editor Details
+	ts_3	Low	Mar 28, 2021	•	O II OActivate	Description
=:	ts_2	Low	Mar 28, 2021	•	O II OActivate	- API version v1
Ţ	mani_Weather_API	Low	Mar 22, 2021	•	O II OActivate	Components (1)
×	ts_1	Low	Mar 21, 2021	•	O 🕕 O Activate	Created Date Mar 28, 2021
J	ts	Low	Mar 17, 2021	•	O 🔝 O Activate	Updated Date Mar 28, 2021
0	demo_mon	Low	Mar 16, 2021	•	O II OActivate	MetaDataVersion 1.0
\$	itc_dataload	Medium	Mar 16, 2021	•	O II OActivate	
			Items per page: 1	0 👻 1	- 10 of 119  < < >	1

Note: While clicking the 'Pipeline Workflow Editor' icon the user gets the New Pipeline dialog box. The user can get the Pipeline Workflow Editor page only after creating a new pipeline.

Search Component Image: Space and Custom   System Custom   Reader Image: Space and Custom   Writer Image: Space and Custom   ML Image: Space and Custom   Ingestion Image: Space and Custom   Websocket Image: Space and Custom   Scheduler Image: Space and Custom   Scheduler Save	Pipeline Editor			
Reader   Writer   Transformation   ML   Ingestion   Websocket   Chediuller	Search Component		Private Share	ed
Writer   Transformation   ML   Ingestion   Websocket   Chedular     Kesource Allocation   O Low   Medium   High	System Custom		F Events	+
Transformation   ML   Ingestion   Websocket   Checked	Reader			
ML   Max 60 characters 0/60  Pipeline Description  Resource Allocation  C Low  Medium  High	Writer			
ML	Transformation			
Ingestion   Resource Allocation  Low  Medium  High	ML			
Websocket  C Low  Medium  High	Ingestion			
Scheduler Save	Websocket 🔻			
	Scheduler 🔹	Save		



#### Logs

Advanced Logs: when someone refers to advanced logs, that person is asking to monitor whether the components on the workflow editor are deployed or not.

Pipeline Editor > demo\_mon 0 🔯 🖻 🕴 💿 🗃 😄 🛄 😂 💻 🕸 Search Component 1< E Log xc System Logs ce Logs 3 SFTP Stream Rea Reader . True initialized Ready True Writer ContainersReady True PodScheduled True Transformation SFTP Monito Initialized True ML . Ready True ContainersReady True Ingestion PodScheduled True

Note: This tab appears when the pipeline is active inside the log panel.

#### Event

An event, in general, is referred to as the Kafka topic that works as a communication channel between two components.

In-event: the previous event/ Kafka-topic. This gives input to the component.

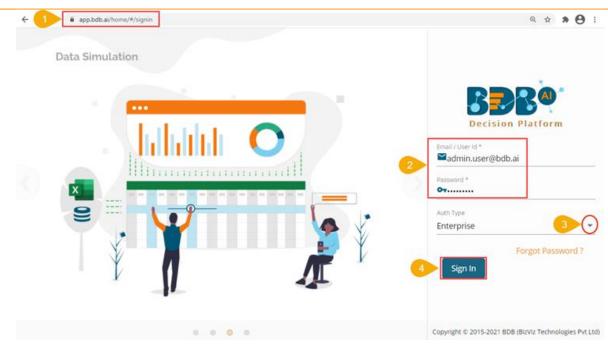
Out-event: the event after the component is out event. This is where the output of the component is present.



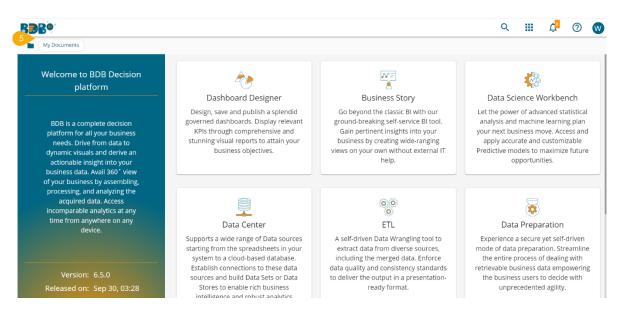
## 4. Logging In

- i) Browse the URL link: <u>https://app.bdb.ai</u> to open the Sign In page.
- ii) Enter user-specific credentials (Email ID with valid Password).
- iii) Select an authentication type using the drop-down.
- iv) Click the 'Sign In' option.





v) The Platform homepage opens.



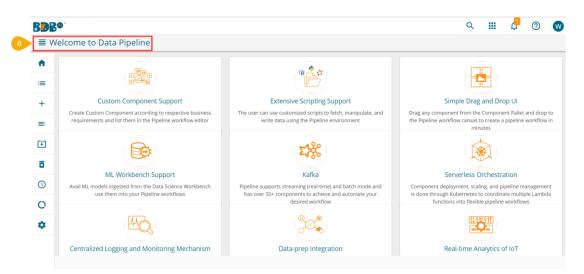
Note: Make sure that the user has permission to access the Data Pipeline plugin.

- vi) Click the Apps menu.
- vii) Select the Data Pipeline Plugin.





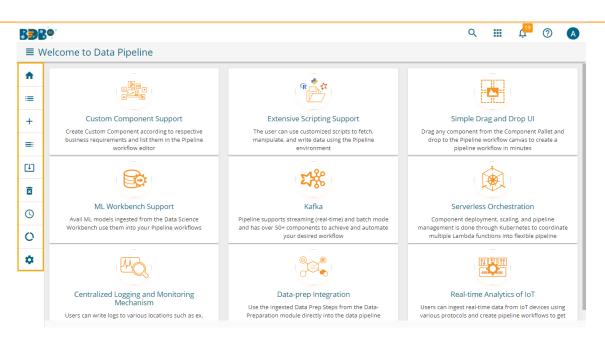
viii) The Data Pipeline Homepage opens.



### 5. Homepage Menu Panel

The Data Pipeline homepage opens with a menu panel on the left side of the screen. The user can expand the menu panel by a click on it as highlighted in the following image:





The user can access various options from the Homepage Menu Panel to work upon the data flow.

#### 5.1. List Pipeline

The 'List Pipelines' option opens the available Pipeline List for the logged-in user. All the saved pipelines by a user get listed on this page. The 'Details' tab on the right side of the page displays the basic details of the selected pipeline.

<b>≣</b> Pip	oeline List									Search	C	l 🔒 🗖
<ul><li>↑</li></ul>	Name	Resources	Created Date ↓	Status				A	lction		• Python script	×
+	Python script	Medium	Dec 10, 2021	•	7	1	0	16	<	<b>Empty</b>	API version	\ \
	test6	Medium	Dec 10, 2021	•	7	1	0	ı.	<	• Deactivate	Components	V
X	RDBMS_test	Medium	Dec 10, 2021	•	7	1	0	њ	<	Activate	Created Date	Dec 10, 202
Э	SFTP reader and monitor	Medium	Dec 10, 2021	•	7	1	0	16	<	<ul> <li>Activate</li> </ul>	Updated Date	Dec 10, 202
С	Training_Pipeline	Low	Dec 10, 2021	•	7	1	0	16	<	<ul> <li>Deactivate</li> </ul>	MetaDataVersion	1
¢	TrainingPL2	Low	Dec 10, 2021	•	7	1	0	16	<	Oeactivate	Description	
	Twitter scrapper data	Medium	Dec 9 2021	•	5	1	•	m	~	Activate		
			ltems pe	r page: 10	•	1 -	10 of 3	356	<	$\langle \rangle \rangle$		

Note:

- a. The user can open the Pipeline Editor for the selected pipeline from the list by clicking the '**View**' o icon or the '**Pipeline Workflow Editor**' icon on the Pipeline List page.
- b. The user can search for a specific pipeline by using the Search bar on the Pipeline List. By typing a common name all the existing pipelines having that word will list. E.g., By typing 'Sam' all the existing pipelines with the word 'Sample' in it get listed in the following image:



2 🗏 Pip	eline List					3	pipelinemi	(	۹
	Name	Resources	Created Date 🕹	Status	A	Action		• flattenJsonTest	× ×
+	pipelineMigration22	Low	Aug 26, 2021	•	↗ ⊙ 🖪	< 0	Activate	API version	v1
=:	pipelineMigration21	Low	Aug 25, 2021	•	↗ ⊙ ₶	< 0	Activate	Components	3
Œ	pipelineMigration20	Low	Aug 25, 2021	•	↗ ⊙ ₪	< 0	Activate	Created Date	Sep 7, 2021
×	pipelineMigration19	Low	Aug 25, 2021	•	↗ ⊙ 🖪	< 0	Activate	Updated Date	Sep 7, 2021
O	pipelineMigration19	Low	Aug 25, 2021	•	↗ ⊙ 🖪	< 0	Activate	MetaDataVersion	1.0
0	pipelineMigration18	Low	Aug 25, 2021	•	↗ ⊙ 🖪	< 0	Activate	Description	
۵	pipelineMigration17	Low	Aug 25, 2021	•	7 0 🕅	< 6	Activate		
			Items	per page: 10	▼ 1 - 10 of 22	<	< > >I		

#### 5.1.1. Sharing a Pipeline

The user can share pipeline with one or multiple users and user Groups using the 'Share Pipeline' option.

- i) Click the 'List Pipelines' icon to open the List Pipelines.
- ii) Select a Pipeline and click on the 'Share Pipeline' < icon.

Resources Low Low	Created Date ↓ Oct 29, 2020 Oct 29, 2020 Oct 29, 2020 Oct 29, 2020	Status •	7	-	, 11. 11.	Action	<ul> <li>Activate</li> <li>Activate</li> </ul>	Python script     Details     API version	×
Low	Oct 29, 2020 Oct 29, 2020	•	7	-	11.	<	-		V
Low	Oct 29, 2020	•	7	-		•	-	API version	v
				0	ılı	<	Activate	API version	V.
Low	Oct 29, 2020	•							
			/	•	ıl.	<	Activate	Components	
Medium	Oct 28, 2020	•	7	0	ılı	<	Activate	Created Date	Dec 10, 2021
Medium	Oct 13, 2020	•	7	0	2	<	<b>©</b> Empty	Updated Date	Dec 10, 2021
Low	Oct 5, 2020	•	7	0	ıl.	<	Activate	Description	
ut Low	Jul 22, 2020	•	7	0	ıl.	<	<ul> <li>Activate</li> </ul>		
	Low	Low Oct 5, 2020 ut Low Jul 22, 2020	Low Oct 5, 2020 • ut Low Jul 22, 2020 •	Low Oct 5, 2020 • <b>7</b>	Low Oct 5, 2020 • 7 •	Low Oct 5, 2020 • 7 © ut Low Jul 22, 2020 • 7 © II	Low Oct 5, 2020 • 7 ③ fi <	Low         Oct 5, 2020         Image: Constraint of the second se	Low Oct 5, 2020 • 7 • Activate Description

- iii) The 'Share Pipeline' window opens.
- Select an option out of the given choices: User and User Group or Exclude User (the 'Exclude User' option can be chosen, if the document is already shared with a user/user group and you wish to exclude them from the privilege).
- v) Select a user or user group using the displayed list of users/ user groups.
- vi) Click the arrow to move the selected User(s)/ User Group(s).

4	user user group exclude user
.0	bulkuserjvraa RestAutomation RestAutomation RestAutomation BestAutomation BestAutomation Bilden bilden bilden bilden Sabarish automationuserste Sabarish Angha Angha K N Dijesh Digesh op



- vii) The selected user(s)/user group(s) get moved to the box given on the right.
- viii) Click the 'Save' option.

Share Data Pipeline	exclude user
RestAutomation RestAutomation RestAutomation RestAutomation RestAutomation bdbTester Sabariesh automationuserstg kalanidhi kalanidhi m Anagha Anagha K N bijeesh bijeesh op shyam shyam Anagha team-pa	7 Shivam Agnihotri

ix) Privilege for the selected pipeline gets updated and the same gets communicated through a message.

≣ Pip	oeline List		ata Pipeline shared p	rivileges has h	een unda	teri			Sam	Q	0 💷
<b>f</b>			uccessfully	inviteges tias b	een upua	leu				<ul> <li>sample_pipeline</li> </ul>	×
=	Name	Resources	Created Date 👽	Status			,	Retion		Details	
+	sample 4	Low	Oct 29, 2020	•	7	0	ıl.	<	<ul> <li>Activate</li> </ul>		
	sample3	Low	Oct 29, 2020	•	7	0	ıl.	<	<ul> <li>Activate</li> </ul>	API version	V1
≡:	sample2	Low	Oct 29, 2020	•	7	0	ıl.	<	<ul> <li>Activate</li> </ul>	Components	
<b>U</b>	sample	Medium	Oct 28, 2020	•	7	0	ıl.	<	<ul> <li>Activate</li> </ul>	Created Date	Oct 13, 2020
0	sample_pipeline	Medium	Oct 13, 2020	•	7	0	ıl.	<	<b>Empty</b>	MetaDataVersion	Dec 9, 2021
\$	sample_SFTP	Low	Oct 5, 2020	•	7	0	11.	<	<ul> <li>Activate</li> </ul>	Description	
	Smart Meter XML Input	Low	Jul 22, 2020	•	7	0	ıl.	<	<ul> <li>Activate</li> </ul>		
			ltems p	er page: 10	▼ 3 <sup>3</sup>	1 – 39 o	f 39	\\	< > >		

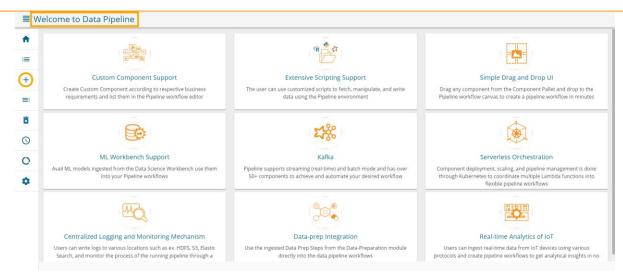
x) By completing the steps mentioned above, the concerned pipeline gets successfully shared with the selected user/user group or the selected users can also be excluded from their privileges for the concerned pipeline.

**Note:** A non-admin user can '**Edit/Modify'** and '**View'** a pipeline but cannot '**Share'** it or '**Delete'** a shared pipeline.

#### 5.2. Create a Pipeline

The '**Create Pipeline**' option redirects the user to create a new pipeline and eventually to the Pipeline Editor (the user interface) to create, update, activate, and save the Pipeline Workflow. Click the '**Create Pipeline**' icon from the menu panel.





A new dialog box opens redirecting the user to the Pipeline Editor.

Pipeline Editor			
Search Component		Private	Shared
System Custom		F Events	+
Reader 🔹			
Writer	式 New Pipeline ×		
Transformation	Enter name *		
ML	Max 60 characters 0/60 Pipeline Description		
Ingestion	Resource Allocation		
Websocket	O Low O Medium O High		
Scheduler	Save		

Note: The stepwise process to create a Pipeline workflow is described later in this document.

#### 5.3. List Components

All the available component names are listed on this page. The user gets to know about the Deployment Type, Component Type, Version, Component Group about a component. The Expose status is indicated through the green color coding if the related component is exposed.

The user gets the Component version information from the List Component page. The version of the components get updated with each new release of the respected components.



80.							Q :	I 💾	?
Compo	onent List					Search C	Component		۹
	Mana	Dealer mean Trace	Company to Taxa	Marrian	Company of Comm	la Francisco d	Antione	-	1
	Name	Deployment Type	Component Type	Version	Component Group	IsExposed	Actions	J	_
	S3 Reader	spark	system	6.5.0-R1	Reader	٠	Θ		
	HDFS Reader	spark	system	6.5.0-R1	Reader	•	Ο		- 1
	Cassandra Reader	spark	system	6.5.0-R1	Reader	•	Ø		
	RDBMS Reader	spark	system	6.5.0-R1	Reader	•	Θ		
	ES Reader	spark	system	6.5.0-R1	Reader	•	Θ		
	S3 Writer	spark	system	6.5.0-R1	Writer	•	Θ		
	DDRMC Writer	chark	custom	6 5 0 D1	Writer	•	•		
					Items per page: 10	<b>)</b> 🔻 1 – 10 d	of 58   <	< >	>

The user can get the Basic Information about the selected component by clicking the 'Actions' • icon. The basic information can be edited from this window if needed. The user gets Ports and Spark Component Information if the selected Deployment Type option is '**Spark**'.

Basic Information	^
Name * S3 Reader	
Deployment Type * spark	Image Name * 371194629034.dkr.ecr.us-west-2.amazonaws.com/com.bdbizviz.dp/s:
Version * 6.5.0-R1	isExposed True
Component Type * System	Component Group * Reader
Ports	~
Spark Component Information	~

The user can search for the desired components using the Search bar. The Component List gets customized based on the searched components.

<mark>≣</mark> Compo	nent List					53		Q (
<b>n</b>	News	Dealer and Trans	Component Type	Manzian	Component Group	IsExposed	4.00	
=	Name	Deployment Type	Component Type	Version	Component Group	ISEXposed	Actions	
+	S3 Reader	spark	system	6.5.0-R1	Reader	•	0	
	S3 Writer	spark	system	6.5.0-R1	Writer	•	Ø	
Î								
0								
0								
\$								
					Items per page:	<u>10 ▼</u> 1-2 o	of 2  < <	> >1



#### 5.4. Delete Pods

The user can delete the orphan pods using this option.

- i) Click the '**Delete**' icon.
- ii) The 'Delete Orphan Pods' page opens.
- iii) Click the '**Delete**' icon.

2 ■ Delete C	Drphan Pods			
÷	Component Name	Created On	Pipeline	
=				
+	Rule Splitter	Sep 8, 2021, 3:27:03 PM	testMigration08	
=				
1 🗊				
Q				
0				
٥				

iv) A notification message appears, and the listed orphan pods get deleted.

■ Delete Orphan I	Pods			C 🔳
<b>n</b>	Component Name	Pods Deleted Successfully! Created On	Pipeline	
=				
+				
=:				
Ð				
×				
0				
0				
•				

#### 5.5. Scheduler

It opens the Scheduler List with the status and Triggered Time of the scheduled pipeline components mentioned in the right-side panel.

Scheduler Name	Scheduler Time	Next Run Time	Pipeline Name	Schedulermig	
Schedulermig	0 0/1 * 1/1 * ? *	03-09-2021 17:04:00	MigrationPipeline	Description	Triggered Time
TESTNAME	0 0/2 * 1/1 * ? *	17-08-2021 18:58:00	TestDemo	Successfully Executed	Aug 9, 2021, 12:46:01 PM
sch1	0 0/1 * 1/1 * ? *	13-08-2021 10:59:00	NewScheduler	Successfully Executed Successfully Executed	Aug 9, 2021, 12:47:00 PM Aug 9, 2021, 12:48:00 PM
sch1	0 0/1 * 1/1 * ? *	13-08-2021 10:43:00	NewScheduler	Successfully Executed	Aug 9, 2021, 12:49:00 PM
test003	0 0/2 * 1/1 * ? *	13-08-2021 16:02:00	b	Successfully Executed Successfully Executed	Aug 9, 2021, 12:50:00 PM Aug 9, 2021, 12:51:00 PM
testdf	0 0/2 * 1/1 * ? *	20-08-2021 11:00:00	mig13	Successfully Executed	Aug 9, 2021, 12:52:00 PM
sch3	0 0/2 * 1/1 * ? *	None	testMigration03	Successfully Executed Successfully Executed	Aug 9, 2021, 12:53:00 PM Aug 9, 2021, 12:54:00 PM
	000+14+0+	None	test figuries 0.4	1 - 10	of 38  < <

Note: The Scheduler Time column displays the time as Cron expressions.



#### 5.6. Data Channel

The Data Channel window displays the status of the Kafka Server.

∎D	ata Channel	
<b>f</b>	Channel	Status
=	kafka-cluster-kafka-0.kafka-cluster-kafka-brokers.kafka.svc:9092	•
+	kafka-cluster-kafka-1.kafka-cluster-kafka-brokers.kafka.svc:9092	•
-	kafka-cluster-kafka-2.kafka-cluster-kafka-brokers.kafka.svc:9092	•
0		
•		
٠		

#### 5.7. Settings

The user gets settings information about various pipeline functionalities. The Settings page provides information about Kafka Configuration, Data Prep Scripts, Data Science Models, and Logger. The Default Configuration and Pod Status get displayed under the Settings option.

<b>≣</b> Pi	peline Settings		
<b>A</b>	Kafka Configuration	Kafka Configuration	С 🖪
:=	Dataprep Scripts		
+	Data Science Models	Basic Information ~	
=:	Logger	Configuration Details*	
×	Default Configuration		
0			
0			
٢			

The details of each Settings option are described under the Settings section of this document.

### **6. User Interface**

The User Interface of the Data Pipeline provides a canvas for the user to build the data flow (Pipeline Workflow).

The Data Flow creation process can be divided into three parts as mentioned below:

- 1. Creating a Pipeline
- 2. Adding Components to the Canvas
- 3. Connecting Components

#### 6.1. Creating a Pipeline

- i) Navigate to the Data Pipeline landing page.
- ii) Click the 'Create Pipeline' option provided on the top right side of the Pipeline landing page.



•		R 📥 🕸	
=		Ē	
+)	Custom Component Support	Extensive Scripting Support	Simple Drag and Drop UI
=	Create Custom Component according to respective business requirements and list them in the Pipeline workflow editor	The user can use customized scripts to fetch, manipulate, and write data using the Pipeline environment	Drag any component from the Component Pallet and drop to the Pipeline workflow canvas to create a pipeline workflow in minutes
	<u>A</u>	20	
J		erso	
0	ML Workbench Support	Kafka	Serverless Orchestration
\$	Avail ML models ingested from the Data Science Workbench use them into your Pipeline workflows	Pipeline supports streaming (real-time) and batch mode and has over 50+ components to achieve and automate your desired workflow	Component deployment, scaling, and pipeline management is dor through Kubernetes to coordinate multiple Lambda functions into flexible pipeline workflows
	MO.		
	Centralized Logging and Monitoring Mechanism	Data-prep Integration	Real-time Analytics of IoT
	Users can write logs to various locations such as ex. HDFS, S3, Elastic Search, and monitor the process of the running pipeline through a	Use the ingested Data Prep Steps from the Data-Preparation module directly into the data pipeline workflows	Users can ingest real-time data from IoT devices using various protocols and create pipeline workflows to get analytical insights in

- iii) The 'New Pipeline' window opens asking for the basic information.
- iv) Enter a name for the new Pipeline.
- v) Describe the Pipeline (Optional).
- vi) Select a resource allocation option using the radio button- the given choices are:
  - i. Low
  - ii. Medium
  - iii. High
- vii) Click the 'Save' option to create the pipeline. By clicking the 'Save' option, the user gets redirected to the pipeline workflow editor.

3	🛱 New Pipeline	×
4	Enter name * Sample Pipeline	
5	Pipeline Description	1,
6	Resource Allocation	
		7 Save

- viii) A success message appears to confirm the creation of a new pipeline.
- ix) The Pipeline Editor page opens for the newly created pipeline.



earch Component	Pipeline creation success	Private Shared
System Custom		F Events -
ader	-	
riter	•	
ansformation		
L		
gestion	• • • • • • • • • • • • • • • • • • • •	
ebsocket		
heduler		

- x) The newly created pipeline appears at the top of the Pipeline List.
- xi) The status of the pipeline is indicated through color-coding under the 'Status' column.
  - a. Green-Activated Pipeline

9

b. Red- Deactivated Pipeline

≣ Pi	■ Pipeline List						-	Search				
<b>A</b>	Name	Resources	Created Date 🗸	11 Status				Actic		Sample Pipeline	×	
i≡ 10	Name	Resources	Created Date 🕹	Status				Actio		Details		
+	Sample Pipeline	Medium	Oct 13, 2020	•	7	0	ıh	<	Empty	API version	v1	
=:	sftp_0	Low	Oct 6, 2020	•	7	0	ıh	<	Activate	Components	-	
×	sample_SFTP	Low	Oct 5, 2020	•	7	0	ıh	<	Activate	Created Date	Oct 13, 2020	
0	test09	Low	Sep 22, 2020	•	7	0	ıh	<	Activate	Updated Date	Dec 13, 2021	
0	TEST_18	Low	Sep 18, 2020	•	7	0	ıh	<	Activate	MetaDataVersion	1.0	
٥	Test1609	Low	Sep 16, 2020	•	7	0	11.	<	Activate	Description	-	

The following icons get displayed on the Pipeline Editor page:

Icon	Name	Function
IK .	Open/ Close the Components Pallet	Opens or Closes the Component Pallet
0	Edit the Pipeline Name/resource allocation type	Allows to modify or change the given Pipeline name or change default resource configuration for pipeline components.
E	Toggle Log Panel	Displays the UI Logs.
٩	Search Component in Pipelines	Searches for a specific component across all the existing pipelines.
۶ 🕺	Toggle Event Panel	Opens and closes the Event Panel to create and display created Events
•	Global Variable	Opens the Global Variables window to add new variables.
$\odot$	Activate/Deactivate Pipeline	Activates or deactivates the Pipeline. It appears once the user clicks the Update Pipeline icon
8	Update Pipeline	Saves and updates the pipeline



Î	Delete Pipeline	Deletes the pipeline
53	Full Screen	Displays the present page in full screen
ıl.	Pipeline Monitoring	Displays all the essential details about the selected pipeline to monitor it.
•	Create a New Pipeline	Redirects the users to create a new pipeline
	List Pipeline	Redirects the users to the page listing all the existing pipelines
\$	Settings	Redirects the users to the Pipeline Settings page

#### 6.1.1. Resource Allocation

The following is used to deploy the pipeline with high, medium, or low-end configurations according to the velocity and volume of data that the pipeline must handle.

All the components saved in the pipeline are then allocated resources based on the selected Resource Allocation option depending on the component type (Spark and Docker).

≓ New Pipeline	×
Enter name *	
Max 60 characters 0/60	
Pipeline Description	
	//
Resource Allocation	
🔿 Low 💿 Medium 🔿 High	
	Save

### 6.2. Adding Components to the Canvas

The Component Pallet is situated on the left side of the User Interface. It has a System and Custom Components tab listing the various components.

Search Com	nponent	\$
System	Custom	
Reader	•	
Writer	-	
Transformati	on 🔻	
ML	-	
Ingestion	-	
Websocket	-	
Scheduler	~	



Once the Pipeline gets saved in the pipeline list, the user can add components to the canvas. The user can drag the required components to the canvas and configure it to create a Pipeline workflow or Data flow.

- i) Navigate to the existing data pipeline from the Pipeline List page.
- ii) Click the '**View**' icon for the pipeline.

(The user can perform this step-in continuation to the pipeline creation, but in case if the user has come out of the Pipeline Editor the above steps help him to access it again.)

≣ Pi	ipeline List						Search	۹ 🛛 🕻	
<b>f</b>								Sample Pipeline	×
	Name	Resources	Created Date 🤸	Status		A	ction	Details	
i + 🚺	Sample Pipeline	Medium	Oct 13, 2020	•	70	16	<	API version	
=	sftp_0	Low	Oct 6, 2020	•	↗ ⊙	11. ×	< OActivate	Components	v1
×	sample_SFTP	Low	Oct 5, 2020	•	↗ ⊙	11. ×	< OActivate	Created Date Oct 13, 2	
0	test09	Low	Sep 22, 2020	•	↗ ⊙	п.	< OActivate	Updated Date Dec 13, 2	
0	TEST_18	Low	Sep 18, 2020	•	↗ ⊙	ı. •	< OActivate	MetaDataVersion	1.0
٥	Test1609	Low	Sep 16, 2020	•	↗ ⊙	ıl. «	<  OActivate	Description	-

- iii) The Pipeline Editor opens for the selected pipeline.
- iv) Drag and drop the new required components or make changes in the existing component's meta information or change the component configuration (E.g., the RDBMS Reader is dragged to the workspace in the below-given image).



- v) Click on the component.
- vi) The component-specific fields open asking the basic information about the dragged component:
  - a. Select an Invocation Type using the drop-down menu.
  - b. The Deployment Type appears based on the selected component.
  - c. The Container Image version displays the docker version.



	5 RDBMS Re			
RDBMS Reader Basic Information Meta I	nformation		9 🕯	÷ <u>∔</u> †
Invocation Type* 🗸	Deployment Type spark	Container Image Version 6.0.0-R3		
Failover Event 🔹	Batch Size(min 10)			

- vii) Open the Meta Information tab and configure the required information for the dragged component.
  - i. Host IP Address
  - ii. Port number
  - iii. Username
  - iv. Password
  - v. Database Name
  - vi. Driver- Select from the drop-down menu
  - vii. Table Name
  - viii. Query
  - ix. Limit
  - x. Selected Columns: Fill the following details manually to select the required columns
    - 1. Name
    - 2. Alias Name
    - 3. Column Type
    - Or

Use 'Download Data' and 'Upload File' options to select the desired columns.

- i. Upload File: The user can upload the existing system files (CSV, JSON) using the 'Upload File'  $\triangle$  icon (file size must be less than 2 MB).
- ii. Download Data (Schema): Users can download the schema structure in JSON format by using the 'Download Data' 🗖 icon.

RDBN	//S Reader					0 1		+	<u>+</u>
Bas	ic Information 7 Meta In	formation							
	Host IP Address*	ddress* Port*		name*	Password*				
	Database Name*	Driver*	•	Table Name					
	Query			Limit					
	Selected Columns					- 6	o ×	( –	
	Name	Alias Name		Column Type	• ×				
	Add New Column								

Note: The users can also upload the downloaded JSON file to another component using the 'Upload File'

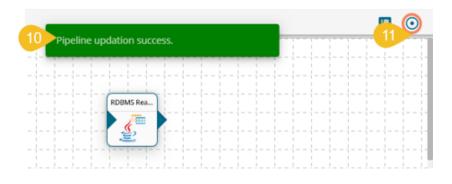


icon.

- viii) Make sure to click the '**Save Component in Storage**' icon <sup>(C)</sup> to update the component details and pipeline to reflect the recent changes in the pipeline. (The user can drag and configure other required components to create the Pipeline Workflow.
- ix) Click the '**Update Pipeline**' 🗖 icon to save the changes.

Pipeline Editor > Sample Pipe	Q 7	● 🕞	Î	0	ı.	: 🕂		\$			
Search Component	RDBMS Reader	0				8	2	Î	:	+	Ţ

- x) A success message appears to assure that the pipeline has been successfully updated.
- xi) Click the 'Activate Pipeline' icon to activate the pipeline (It appears only after the newly created pipeline gets successfully updated).



- xii) A dialog window opens to confirm the action of pipeline activation.
- xiii) Click the '**YES**' to activate the pipeline.



xiv) A success message appears confirming the activation of the pipeline.



Note: The details of the selected pipeline from the Pipeline List appear on the right side of the page.

xv) Another success message appears to confirm that the pipeline has been updated.



15	Pi	pelii	ne u	pdat	tion	succ	ess.							- 1 -	1
	_														
	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

xvi) The Status for the pipeline gets changed on the Pipeline List page.

≣ Pip	oeline List								Search	(	۹ ۵	
A				16						<ul> <li>Sample Pipeli</li> </ul>	ne	×
	Name	Resources	Created Date 🕹	Status				Actio	n	Details		
+	Sample Pipeline	Medium	Oct 13, 2020	$\overline{\bullet}$	7	0	ıl.	<	<b>Empty</b>	ADLucesian		
=:	sftp_0	Low	Oct 6, 2020	•	7	0	ıl.	<	<ul> <li>Activate</li> </ul>	API version		v1
×	sample_SFTP	Low	Oct 5, 2020	•	7	0	11.	<	<ul> <li>Activate</li> </ul>	Components Created Date	Oct 13	-
S	test09	Low	Sep 22, 2020	•	7	0	ılı	<	<ul> <li>Activate</li> </ul>	Updated Date	Dec 13	
0	TEST_18	Low	Sep 18, 2020	•	7	0	ıl.	<	Activate	MetaDataVersion	Dec 15	1.0
٠	Test1609	Low	Sep 16, 2020	•	7	0	ıl.	<	<ul> <li>Activate</li> </ul>	Description		1.0

Note: Click the '**Delete**' icon from the Pipeline Editor page to delete the selected pipeline. The deleted Pipeline gets removed from the Pipeline list.

The status (activated/deactivated) of the pipeline is indicated through color-coding as highlighted in the below image. It turns green if the pipeline is active and remains red for the inactive pipeline.

OPipeline Editor > Sample Pipeline	
------------------------------------	--

Please refer to the Component Pallet section to get detailed information on the Components.

#### 6.3. Toggle Log Panel

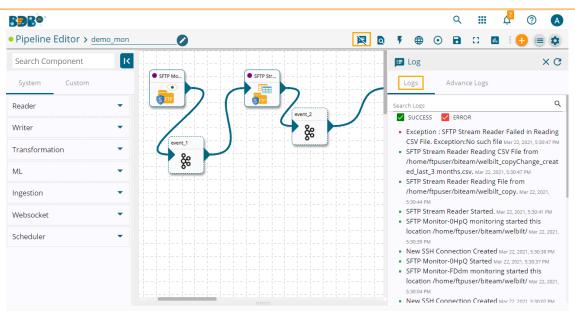
The Toggle Log Panel displays the Logs and Advanced Logs tabs for the Pipeline Workflows.

- i) Navigate to the '**Pipeline Editor**' page.
- ii) Click the 'Toggle Log Panel' icon on the header.

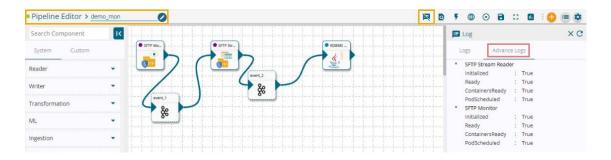
Pipeline Editor > demo_n	non 🕜		E		O I	
Search Component	K					
System Custom	• SFTP Mo	• SFTP Str		• RDBMS	-	
Reader			event_2		J	Q   100%   Q
Writer	-		<pre>kvenCz</pre>			
Transformation	event_1					
ML	-					
Ingestion	•					
Websocket	<ul> <li>- 1 1 1 1</li> <li>- 1 1 1</li> <li>- 1 1 1</li> </ul>					
Scheduler						

iii) A panel **Toggles** displaying the collective component logs of the pipeline under the 'Logs' tab.





iv) Select the 'Advanced Logs' tab to display the status of the component containers.



#### 6.4. Search Component in Pipelines

This feature helps the user to search a specific component across all the existing pipelines. The user can drag the required components to the pipeline editor to create a new pipeline workflow.

i) Click the 'Search Component in pipelines' icon Q from the header panel of the Pipeline Editor.



ii) The Object Browser window opens displaying all the existing pipeline workflows.



🖻 🔍 F	8 👔 🗆 🖬 🗄 🖶 🌉	•
2	Object Browser	
	Search Component	
	Sample Pipeline	~
	test16	~
	itc_dataload	~
	trial_1	~
$\begin{array}{c} -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 $	test15	~
	ltts_welbilt_sftp_py	
		~
	test	~
	ltts_welbilt_dataloa	~
	sample_test	~

- iii) The user can search a component using the 'Search Component' space.
- iv) The user gets prompt suggestions while searching for a component.



v) Once the component name is entered, the pipeline workflows containing the searched pipeline gets listed below.

ĺ	Object Browser	
	Mongodb Writer	
5	test_mongo_1	~
	test_mongo	~

vi) The user can click the collapse/expand button to expand the component panel for the selected pipeline.



Object Browser	
Mongodb Writer	
test_mongo_1	6
test_mongo	~

vii) The user can drag a component to the Pipeline Editor canvass from the object browser window.

	Diject Browser	
	Mongodb Writer	
	test_mongo_1	^
	- 🤎 🔊	
Mongod	🤶 🚬 🛡 🕯	
	O THE OFFICE	
	test_mongo	~

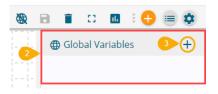
#### 6.5. Global Variable

The user can create a Global Variable by following the below-given steps:

i) Click the Global Variable icon from the User Interface.



- ii) Global Variables panel opens.
- iii) Click the 'Add New Variable' from the Global Variables panel.



- iv) The Add Variable window opens.
- v) Insert Variable name.
- vi) Provide Value for the Variable.
- vii) Click the 'Save' option.



	• R Model	
4	Add Variable	
5	Variable Name * Kafka	
6	Value * admin	
let	Cancel 7 Save	

- viii) A notification message appears.
- ix) The Global Variable gets created and added to the panel.

	٩	F			Î	53	ıl.		Ð		2
Global Variable Successfully Added.				(	🕀 Glo	bal Va	ariab	les		+	
			9		Kafka:	admir	٦			í	

#### 6.6. Deleting a Pipeline

- i) Navigate to the Pipeline Workflow Editor.
- ii) Click the '**Delete Pipeline**' icon on the header panel.



- iii) The '**Confirm**' dialog opens to confirm the deletion.
- iv) Click the '**Yes**' option to confirm.

Confirm		
Are you sure you want to o	delete pipe	eline ?
	NO	YES

v) The selected pipeline gets deleted and gets removed from the List of Pipelines.

### 6.7. Pipeline Monitoring

The user can monitor a pipeline together with all the components associated with the same by using the '**Pipeline Monitoring**' icon. The user gets information about Pipeline components, Status, Types, Last Activated (Date and Time), Last Deactivated (Date and Time), Total Allocated and Consumed CPU %, Total allocated and consumed memory, Number of Records, and Component logs all displayed on the same page.

- i) Navigate to the Pipeline List page.
- ii) Click the 'Monitor' icon.

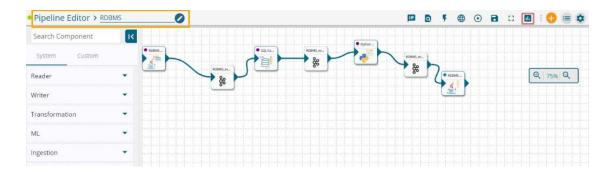


≡ Pip	oeline List				Search	۹ 🛛 🗖
<b>f</b>						RDBMS ×
:=	Name	Resources	Created Date 🦊	Status	Action	Details
+	RDBMS	Low	Mar 21, 2021	•	O Activate	Description
=	ts	Low	Mar 17, 2021	•	O II OActivate	API version v1
	test16	Low	Mar 16, 2021	•	O II O Activate	Components (1)
	itc_dataload	Medium	Mar 16, 2021	•	O il OActivate	Created Date Mar 21, 2021
0	trial_1	Low	Mar 16, 2021	•	O II. OActivate	Updated Date Mar 22, 2021
0	test15	Low	Mar 15, 2021	•	• II OEmpty	MetaDataVersion 1.0
\$	ltts_welbilt_sftp_py	Medium	Mar 12, 2021	•	O II. OActivate	
			Items per page:	10 💌 1	I - 10 of 116   <b>&lt; &lt; &gt; ≻ </b>	

#### Or

Navigate to the Pipeline Workflow Editor page.

Click the '**Pipeline Monitoring**' icon on the Header panel.



iii) The 'Pipeline Monitoring' page opens displaying details of the selected pipeline.

Pipeline		3	RDBMS Last Activated	Mar 1	8, 2021, 4:52:50	PM Total Allocated C	PU	3.5	Total Consumed CPU %	36
tatus			true Last Deactivated			Total Allocated N	lem	4096.0	Total Consumed Mem	595
Component Name	Status Type	Instance	s Last Processed Time	Memory Allocated	CPU Allocated	Consumed Memory	Consumed CPU %	Last Processed Size	Last Processed Count	Total Number Of Records
RDBMS Reader	true realtime	1	22-03-2021 09:24:18	1024m	1	168.75 MB	0.00	0.00 KB	0	0
Python Script	true realtime	1	22-03-2021 09:24:34	1024Mi	0.5	70.5 MB	23.50	264 B	19	100
RDBMS Writer	true batch	1	22-03-2021 09:25:08	1024m	1	201.63 MB	0.00	0.23 KB	4	54
Us	sed Memory			Used CPU %		N	umber of Rec	ords	Component Log	gs C
75 50 25 0 57 0 57 0 57 0 57 0 57 0 57 0	String Star String String	Pro anti-	75 50 25 25 25	New of an and an and an and a	A	150 50 50 50 50 50 50 50 50 50 50 50 50 5	und and and and a	a substant program	<ul> <li>2021, 254/34 PM</li> <li>Python Scrip event 4me 22, 2</li> <li>Python Scrip out event 4me</li> <li>Python Scrip 2021, 254/31 PM</li> <li>Python Scrip event 4me 22, 2</li> <li>Python Scrip</li> <li>Python Scrip</li> </ul>	t-qrms sending data to -22, 2021; 2:54:38 PM t-qrms stopping Mar 22, t-qrms sent data to out

#### 6.8. Connecting Components

Each component in the Data Pipeline works as an independent entity. The user needs to create an Event as a Kafka-based messaging channel to create a data flow. An Event helps the users to read data from various



readers or use the data coming from one pipeline into another pipeline. The Kafka Event helps to deal with the scenarios where one pipeline is dependent on another pipeline's data.

There are two tabs in the Event Panel as described below:

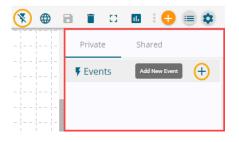
- i) Private: This tab contains a list of all the created events by the user for the current pipeline.
- ii) Shared: This tab contains a list of all the shared events from another pipeline.

#### 6.8.1. Event Creation

- i) Navigate to the Pipeline Editor page.
- ii) Click the 'Toggle Event Panel' <sup>5</sup> icon from the header.

Pipeline Editor > sample_pipelir	ne		Ø					Þ	C	) (	•	€	)	•	•	Î	5	3	ıl.	÷	Đ			2
Search Component	1	1	 	 		 					_					 								
System Custom			 	 		 										 			ſ	Q	100	% C	2	
Reader 🔹			 	 		 										 			L				Ù	
Writer			 	 	 	 										 								
Transformation 🔹		1	 	 	·	 										 ·								
ML 👻			 	 		 										 								
Ingestion 👻			 	 	 	 					1					     ·								
Websocket 🔹			 	 		 										 								
Scheduler 👻			 	 		 	1									 								
Scheduler			 	 		 										 				-				

- iii) The Events Panel appears, and the Toggle Event Panel icon gets changed as Suggesting that the event panel is displayed.
- iv) Click the 'Add New Event' icon from the Event Panel.



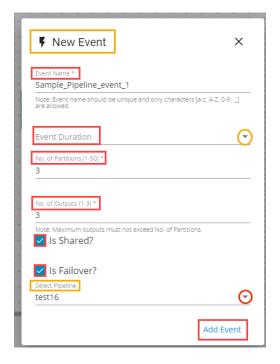
- v) The New Event window opens.
  - a. Provide a name for the new Event.
  - b. Select Event Duration: Select an option from the below-given options.
    - i. Short (4 hours)
    - ii. Medium (8 hours)
    - iii. Long (48 hours)

Note: The event data gets erased after 7 days if no duration option is selected from the available options.

- c. Provide No. of Partitions (1-10).
- d. No. of Outputs: The user can define the number of outputs using this field.
- e. Checkmark the 'Is Shared' option to make the created Event available as a Shared Event.
- f. Checkmark the 'Is Failover' option to enable the Failover Event.
- g. Select a Pipeline using the drop-down menu if you have selected the 'Is Shared?' option.



h. Click the 'Add Event' option.

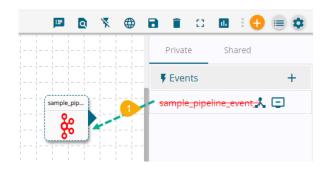


- vi) A notification message appears.
- vii) The newly created Event gets added to the 'Private' tab in the Events panel.

	<b>I</b>	Q 🕅	e 🖬 🖸	🖪 : 🕂 (	
Event added successfully			Private	Shared	
			 🖡 Events		+
			 Sample_Pipe	eline_event 🙏 🕻	3 📋

#### 6.8.2. Updating an Event

i) Drag the created Event component to the Pipeline Editor canvas.



- ii) Click the dragged Event component to open the 'Basic Info' configuration fields.
- iii) The user can edit the following information (except No. of Partition and Event Duration):
  - a. Event Name: Modify the event name.
  - b. No. of Outputs: Set the number of outputs (the maximum allowed number of outputs are 3)
  - c. 'Is Shared?': Enable this option to share the created event.
  - d. 'Is Failover?': Enable this option to create a failover event.
  - e. Select Pipeline: Select one pipeline or multiple pipelines using the drop-down list.



#### iv) Click the 'Save Event' icon to save the changes.

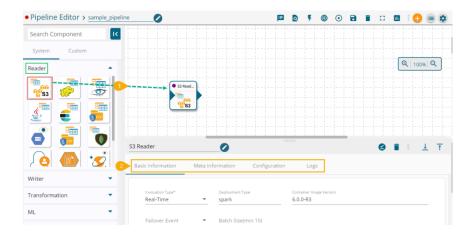
Basic Info Preview		
Event Name *	No. of Partitions (1-50)	
sample_pipeline_event_1	3	
Note: Event name should be unique and only characters [a-z, A-Z, 0-9,] are allowed.		
No. of Outputs (1-3) * 3	Event Duration	
Note: Maximum outputs must not exceed No. of Partitions.		
✓ Is Shared?	□ Is Failover?	
Select Pipeline		
Select Fipeline		

- v) The user gets a notification message stating that the pipeline update is a success.
- vi) The targeted Event component gets updated.

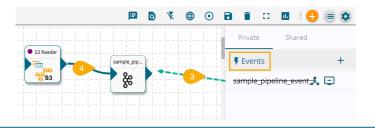


#### 6.8.3. Connecting Event to a Component

- i) Drag a (reader) component to the canvas.
- ii) Configure the parameters of the dragged reader.

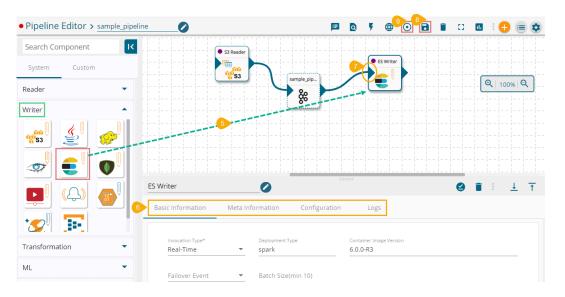


- iii) Drag the Event from the Events Panel.
- iv) Connect the dragged reader component as the input connection to the dragged Event.





- v) Drag a Writer component to the Editor canvass from the Component Pallet.
- vi) Configure the writer component.
- vii) Connect the Writer component as an output connection to the dragged Event to complete the data flow.
- viii) Click the 'Update Pipeline' icon to save the pipeline workflow.
- ix) Click the 'Activate Pipeline' icon to activate the Pipeline.



Note: The user gets notification messages on the successful update and activation of the Data Pipeline.

#### 6.8.4. Shared Event

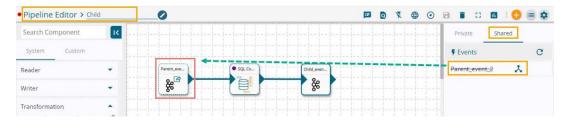
i) Enable the 'Is Shared?' option while creating an Event.

Event Name *	
Parent_event_2	
Note: Event name should be unique and only are allowed.	characters [a-z, A-Z, 0-9,]
Event Duration	
	-
No. of Partitions (1-50) *	
3	
No. of Outputs (1-3) *	
1	
Note: Maximum outputs must not exceed No.	of Partitions.
Shared?	
□ Is Failover?	
Select Pipeline	
Child	_

- ii) Open the pipeline where the same event is shared.
- iii) The shared event gets added to the 'Shared' section of the Event.



 iv) The user can use the Shared Event in the new pipeline with a writer as displayed in the following image:



#### 6.8.5. Preview Data

The user can preview the fetched data from the configuration panel of the Event component in the pipeline workflow.

- i) Navigate to a pipeline workflow (The pipeline should be activated).
- ii) Click on a connected Event component.
- iii) The Basic Information tab of the Event Component opens with the default configuration which is given by the user on Event creation.

Pipeline Editor > device master	er 🖉	🛤 🖻 🦹 🌐 💿 🖬 📋 🗆 🖶 🖶
Search Component	• • • • • • • • • • • • • • • • • • •	
System Custom		-
Reader 👻		• SQL Co
Writer 🝷	Morgod	
Transformation 🔹		
ML 👻		
Ingestion 👻	device_master_event_3	😂 : <u>1</u> -
Websocket 🔹	Basic Info Preview	
Scheduler 👻	Event Name * device_master_event_3	No. of Partitions (1-50) 3
	Note: Event name should be unique and only characters [a-z, A-Z, 0-9, _] are allowed.	
	No. of Outputs (1-3) * 1	Event Duration Short (4 hours) 🗢

- iv) Click the 'Preview' tab from the Event configuration panel.
- v) The Preview Data tab appears displaying the fetched data with the count of the Number of Records.
- vi) Click the '**Download as CSV**' option to download the data in a CSV file.
- vii) Click the '**Refresh Data**' option to refresh the fetched data.

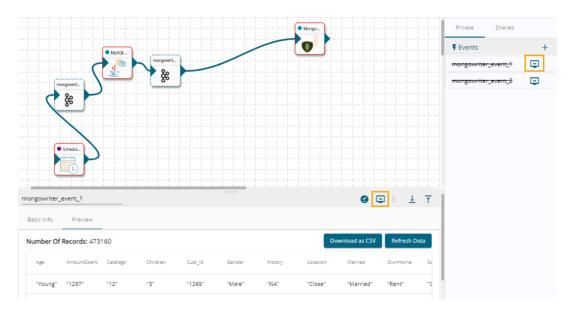
sic Info Preview													
mber Of Records: 37	064										Downlo	ed as CSV R	efresh i
device_jd provider	registration_date state	is_linked_with	_cusconfig_profile_id	iccid	imei	vin	device_active_date	msi	ecall_only	brand	dcm_type	device_status	
"353161077"KFT"	"2020-04- 17T14:08:31Z" "ACTIVE"	true	"98616303- bae6-4793- a008- 2bd0e0c6469a		"353161077	82'JTHB21B14	020 <sup>°°</sup> 2020-06- 03T10:09:34Z*	'null'	"null"	"LEXUS"	"null"	"ACTIVE"	
"353161079"KFT"	"2020-04- 17T14:08:11Z" "ACTIVE"	true	"98616303- bae6-4793- a008- 2bd0e0c6469a	"null"	"3531610790	95'JTJYWRBZ10	050 <sup>°2020-06-</sup> 25T19:17:492°	'null'	"nuil"	"LEXUS"	"null"	"ACTIVE"	
"357703090"DCM19"	"2020-01- 13T11:07:10Z" "ACTIVE"	true	"b7cb2026- 11fd-4589- a72a- 8c557417448f"	"null"	"null"	'JTMDW3FV	300 <sup>°°2020-01-</sup> 15T14:39:182°	'null'	"null"	"TOYOTA"	"null"	"ACTIVE"	
"357703090"DCM19"	"2020-03- 20T09:11:44Z" "ACTIVE"	true	"b7cb2026- 11fd-4589- a72a- 8c557417448f"	"null"	"null"	"JTHAABBH7	102( <sup>2</sup> 2020-03- 30T11:05:05Z"	'null'	"null"	"TOYOTA"	"null"	"ACTIVE"	
"357703090"DCM19"	"2019-11- 20T10:35:572" "ACTIVE"	true	"b7cb2026- 11fd-4589- a72a- 8c557417448f"	"null"	"null"	"VNKKBAC3	40A <sup>*2020-04-</sup> 03T13:49:512*	'null'	"null"	"TOYOTA"	"null"	"ACTIVE"	
"357703090"DCM19"	"2020-01- 27T16:52:49Z" "ACTIVE"	true	"b7cb2026- 11fd-4589- a72a- 8c557417448f"	"null"	"null"	"VNKKAAC3	90A <sup>"2020-02-</sup> 14T15:43:332"	'null'	"null"	"TOYOTA"	"null"	"ACTIVE"	
"357703090"DCM19"	"2020-01- "ACTIVE"	true	"b7cb2026- 11fd-4589-	"null"	"null"	"JTHU958H0	02(2020-01-	'null'	"null"	"TOYOTA"	"null"	"ACTIVE"	



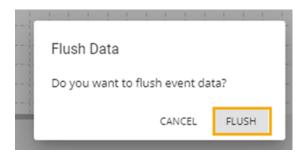
### 6.8.6. Flush Event

This feature helps the user to clear the fetched data from the Kafka Event by using the Flush Event.

- i) Click on an Event component.
- ii) Click on the Preview option from the configuration panel.
- iii) Click the 'Flush Event' icon.



- iv) The Flush Data dialog window opens.
- v) Click the 'FLUSH' option.

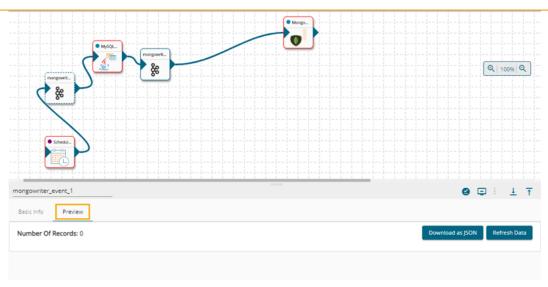


vi) A notification message appears to inform that the data has been flushed.



vii) Open the Preview tab, there will be no fetched data to preview.





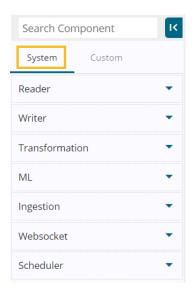
# 7. Component Pallet

The user interface of the Data Pipeline has a component pallet on the left side of the page. The Component Pallet lists System and Custom components of the Data Pipeline.

# 7.1. Sorting and Grouping Components

The Component Pallet displays a list of all the pipeline components. The Components are majorly divided into two groups.

i) System-The pre-designed pipeline components are listed under the 'System' tab.



ii) Custom- The Custom tab lists all the customized pipeline components created by the user.



Search Co	omponent	IK
System	Custom	]
Transform	ation	<b>^</b>
<u> </u>	<i>:</i>	<i>:</i>
selle		

The Component gets grouped based on the Component type. E.g., All the reader components are provided under the 'Reader' menu tab of the Component Pallet.



# 7.2. Searching a Component

A search bar is provided to search across the 50+ components. It helps to find a specific component by inserting the name in the Search Bar.

i) Navigate to the Search bar provided in the Component Pallet.

• Pipeline Editor > <u>api</u>						
Search Cor	nponent	I<				
System	Custom					
Reader		-				
Writer		-				

- ii) Type in the given search bar.
- iii) The user gets prompt suggestions for the searched components. E.g., While searching 'hd', it lists HDFS Reader and HDFS Writer components.

hd	IK
HDFS Reader	
HDFS Writer	-
Writer	•
Transformation	-
	_



- iv) Select an option from the prompted choices.
- v) The searched component appears under the 'System' tab. (E.g., HDFS Reader component as displayed in the below image).

HDFS Read	ler	IK
System	Custom	
Reader		•

# 8. Component Properties

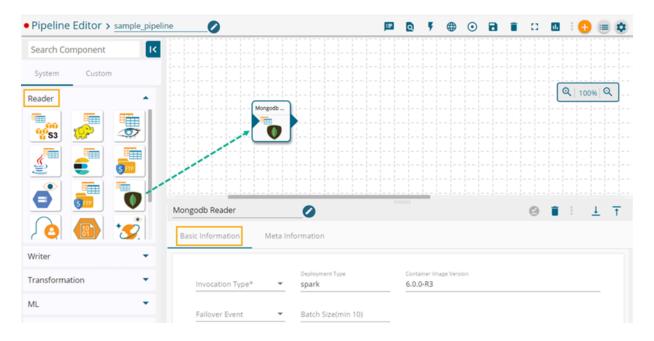
Each component has properties to configure. The user can use a specific component in the Pipeline Workflow only after configuring the required parameters for that component.

This section aims at describing all the component parameters as per the given groups under the Component Pallet.

The following four main tabs appear for each Pipeline component:

- 1. Basic Information
- 2. Meta Information
- 3. Configuration
- 4. Logs

Click on a dragged pipeline component to open the component properties tabs below.





# 8.1. Basic Information Tab

It is the default properties tab to open for the dragged Pipeline Component while configuring the component (E.g., Mongo DB Reader as displayed in the below image).

- Select an Invocation type from the drop-down menu to confirm the running mode of the reader component. Select '**Real-Time**' or '**Batch**' from the drop-down menu.
- Deployment Type: It displays the deployment type for the component. This field comes pre-selected.
- Container Image Version: It displays the image version for the docker container. This field comes predefined.
- Failover Event: Select a failover Event from the drop-down menu.
- Batch Size: Provide a maximum number of records to be processed in one execution cycle (Min limit for this field is 10).

Mongodb Reader	0		٢	Î	 +++++++++++++++++++++++++++++++++++++++	<u>+</u>
Basic Information	Meta Information					
1 Invocation Type* Real-Time	2 Deployment Type spark 5 Batch Size(min 10) 2	Container Image Version 6.0.0-R3				

# 8.2. Meta Information Tab

Open the 'Meta Information' tab and fill in all the connection-specific details of Mogo DB Reader. The

Meta

Information details may vary for each component.

The user can configure the Mongo DB Reader via two connection types as explained below:

odb Reader			0 🖬 :	Ŧ .
Standard				
SRV Database Name*	Host*	User Name*	Password*	

- Meta Information tab Configuration with 'Standard' as the 'Connection Type'
  - Connection Type: Select the '**Standard**' option as the connection type.
  - Port (\*): Port number.
  - Host IP Address (\*): IP address of the host.
  - Username (\*): Provide username.



- Password (\*): Provide a valid password to access the Mongo DB.
- Database Name (\*): Provide the name of the database from where you wish to read data.
- Collection Name (\*): Provide the name of the collection.
- Query: Insert an SQL query (it takes a query containing a Join statement as well).
- Limit: Set a limit for the number of records.
- Additional Parameters: Provide the details of the additional parameters.

ic Information Meta Inform	nation		
Connection Type* Standard	Port* @ENV.	Host* @ENV.MONGO_DATABASE_	User Name* @ENV. MONGO_DATABASE
Password*	Database Name* database	Collection Name*	Query

- Meta Information tab Configuration with 'SRV' as the 'Connection Type'
  - Host IP Address (\*): Hadoop IP address of the host.
  - Username (\*): Provide username.
  - Password (\*): Provide a valid password to access the Mongo DB.
  - Database Name (\*): Provide the name of the database from where you wish to read data.
  - Collection Name (\*): Provide the name of the collection.
  - Query: Insert an SQL query (it takes a query containing a Join statement as well).
  - Limit: Set a limit for the number of records.
  - Additional Parameters: Provide the details of the additional parameters.

lser Name*	Password*	
iser Name*	Password*	
ENV. MONGO_DATABASE		_
Query	Limit	

 Selected Columns: The users can select some specific columns from the table to read data instead of selecting a complete table; this can be achieved via the 'Selected Columns' section. Select the columns which you want to read and if you want to change the name of the column, then put that name in the alias name section otherwise keep the alias name the same as of column name and then select a Column Type from the drop-down menu.

The supported column types are String, Long, Integer, Float, Double, Date, Datetime.

or

Use 'Download Data' and 'Upload File' options to select the desired columns.

Upload File: The user can upload the existing system files (CSV, JSON) using the 'Upload File' icon (file size must be less than 2 MB).



2. Download Data (Schema): Users can download the schema structure in JSON format by using the 'Download Data' icon.

Selected Columns				
Name	Alias Name	Column Type	• ×	
Add New Column				

• After doing all the Meta Information configurations click the 'Save Component in Storage' icon to save the component.



• A notification message appears to inform about the component configuration success.

	Com	ponent	propei	rties sav	ved.						
0	Click t	Component properties saved. Click the ' <b>Update Pipeline</b> ' icon.									
		Q	ş		•	8	Î	::	ıl.	: 🕂	

• A notification message appears to inform that the pipeline is successfully updated.



Note: The user needs to click on the 'Save Component in Storage'  $\leq$  icon for each component of a Pipeline Workflow and the 'Update Pipeline' conto save the configured component and create pipeline workflow.

## 8.3. Configuration and Logs Tabs

The Configuration tab appears after saving the component with Basic and Meta information and updating the Pipeline it is part of.

- Open a component from the Pipeline Workflow which has been updated.
- The concerned component gets the 'Configuration' and 'Logs' properties tabs.

Mongodb Reader		Ø				۷	Î	+	<u>+</u>
Basic Information	M	eta Information	Co	nfiguration	Logs				
Invocation Type* Real-Time	•	Deployment Type spark		Container Image 6.0.0-R3	Version				
Failover Event	*	Batch Size(min 10) 2							



- Click on the 'Configuration' tab to open it.
- Since the selected component (MongoDB Reader) is a Spark component the Configuration tab displays Driver and Executor information.

Mongodb Reader			2	0 0 0	<b>↓</b> <b>↑</b>	<u> </u>
Basic Information	Meta Information	Configuration	Logs			
Core(min 0.1)* 0.1	Memory(min 250)* 1024					
Executor Instances(min 1)* 1	Core(min 1)* 1	Memory(min 250)* 1024				

• If the concerned component has Deployment Type as Docker, then the Configuration tab will display the details of the Limit and Request. E.g., The below image shows the '**Configuration'** tab for the API Server Ingestion component.

Server Ingestion	iformation Configura	ition Logs	2	Î	ē	Ť
sic information Meta Ir	Configura	ltion Logs				
Limit						
CPU(min 100)*	Memory(min 250)*					
1000	5048					
Request						
CPU(min 100, max 1000)*	Instances(min 1)*	Memory(min 250, max 5048)*				
1000	1	4096				

• Click on the 'Logs' tab to open the Component Logs for a specific pipeline component (E.g., Component Logs for the MongoDB Reader are displayed in the following image).

ongodb Reader	0			2 : ÷ .
Basic Information	Meta Information	Configuration	Logs	
Component Logs				Refresh Logs
<ul> <li>Mongodb Real</li> </ul>	der successfully sent data		91:02 AM	Refresh Logs
<ul> <li>Mongodb Read</li> <li>Mongodb Read</li> <li>Mongodb Read</li> </ul>	der successfully sent data der Sending Data to Even der completed reading da	nt Feb 18, 2021, 12:30:56 AM ata from dm+ collect	<b>IT</b> Feb 18, 2021, 12:30:56 AM	Refresh Logs
<ul> <li>Mongodb Read</li> <li>Mongodb Read</li> <li>Mongodb Read</li> </ul>	der successfully sent data der Sending Data to Even	nt Feb 18, 2021, 12:30:56 AM ata from dm+ collect	<b>IT</b> Feb 18, 2021, 12:30:56 AM	Refresh Logs



Note:

- a. The **Basic Information** and **Meta Information** tabs need to be configured by the user before saving the concerned component while the **Configuration** and **Logs** tabs appear after saving the component.
- b. The user can modify the values for the Spark or Docker components (if needed) using the **'Configuration**' tab.
- c. The user needs to click the 'Save Component in Storage' icon after modifying the 'Configuration' tab.

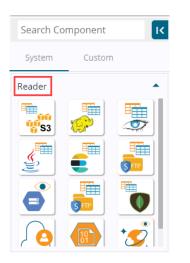
# 9. Components

The Pipeline Editor contains a Component Pallet containing System and Custom components to create Pipeline Workflow.

# 9.1. Reader

The data Reader component helps to read data from different sources and ingest that data into the Pipeline for further processing.

The Data Pipeline contains the following Data Readers under the Component Pallet:



# 9.1.1. S3 Reader

- i) Navigate to the Data Pipeline Editor.
- ii) Expand the Reader section provided under the Component Pallet.



Pipeline Editor > demo	0
Search Component	
System Custom	
Reader 2	
Writer 🔹	
Transformation 🔹	
ML 💌	
Ingestion 🔹	
Websocket 🔹	
Scheduler 🔹	

#### iii) Drag and drop the S3 Reader component to the Workflow Editor.



#### iv) Click on the dragged S3 Reader to get the component properties tabs.

				1								11-	
											Q	100%	Q
	Reader											{	
	\$3							 					
				1									
53 Reader		2			1				0	-	:	1	-
S3 Reader	eta Inform								0	1		<u>+</u>	T
			nt Type			ner Image V	Version		0		:	<u>+</u>	1

- v) Basic Information: It is the default tab to open for the component while configuring it.
  - a. Invocation Type: Select an invocation mode out of 'Real-Time' or 'Batch' using the drop-down menu.
  - b. Deployment Type: It displays the deployment type for the reader component. This field comes pre-selected.
  - c. Container Image Version: It displays the image version for the docker container. This field comes pre-selected.



- d. Failover Event: Select a failover Event from the drop-down menu.
- e. Batch Size (min 10): Provide the maximum number of records to be processed in one execution cycle (Min limit for this field is 10).

	S3 Reader		0		٢	Î	0 0 0	<u>+</u>	Ť
5	Basic Information	Meta Inform	nation						
	Invocation Type*	•	Deployment Type spark	Container Image Version 6.5.0-R1					
	Failover Event	*	Batch Size(min 10)						

- vi) Open the 'Meta Information' tab and fill in all the connection-specific details for the S3 Reader.
  - Configuration fields when SNS Monitor is disabled:
    - i) Bucket Name (\*): Folder Name
    - ii) Zone (\*): S3 Zone location
    - iii) Access Key (\*): Access key shared by AWS to login
    - iv) Secret Key (\*): Secret key shared by AWS to login
    - v) Table (\*): Mention the Table or object name which is to be read
    - vi) File Type (\*): Select a file type from the drop-down menu (CSV, JSON, PARQUET, AVRO are the supported file types)
    - vii) Limit: Set a limit for the number of records.
    - viii) Query: Insert an SQL query (it supports query containing a join statement as well)

Basic Information 6 Me	ta Information		
SNS Monitor	Bucket Name*	Zone	Access Key*
Secret Key*	Table*	File Type *	Limit
Query			

- Configuration fields when SNS Monitor is enabled:
  - 1. Access Key (\*): Access key shared by AWS to login
  - 2. Secret Key (\*): Secret key shared by AWS to login
  - 3. Table (\*): Mention the Table or object name which has to be read
  - 4. File Type (\*): Select a file type from the drop-down menu (CSV, JSON, PARQUET, AVRO are the supported file types)
  - 5. Limit: Set limit for the number of records
  - 6. Query: Insert an SQL query (it supports query containing a join statement as well)



S3 Reader					Î	0 0 0	+	Ŧ
Basic Information	Meta Inforr	nation						
SNS Monitor		Access Key*	Secret Key*	Table*				
File Type *	•	Limit	Query					

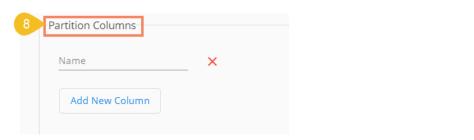
vii) Selected Columns: There is also a section for the selected columns in the Meta Information tab if the user can select some specific columns from the table to read data instead of selecting a complete table so this can be achieved by using the 'Selected Columns' section. Select the columns which you want to read and if you want to change the name of the column, then put that name in the alias name section otherwise keep the alias name the same as of column name and then select a Column Type from the drop-down menu.

Use 'Download Data' and 'Upload File' options to select the desired columns.

- 1. Upload File: The user can upload the existing system files (CSV, JSON) using the '**Upload File**' icon (file size must be less than 2 MB).
- 2. Download Data (Schema): Users can download the schema structure in JSON format by using the '**Download Data**' icon.

Selected Columns					7 🗖 🛆 × –
Name	Alias Name	Column Type	•	×	
Add New Column					

viii) Partition Columns: Provide a unique Key column name to partition data in Spark.



ix) Click the 'Save Component in Storage' icon after doing all the configurations to save the reader component.



x) A notification message appears to inform about the component configuration success.





- a. (\*) the symbol indicates that the field is mandatory.
- b. Either table or query must be specified for the data readers except for SFTP Reader.
- c. Selected Columns- There should not be a data type mismatch in the Column Type for all the Reader components.
- d. The Meta Information fields may vary based on the selected File Type. All the possibilities are mentioned below:
  - i. CSV: 'Header' and 'Infer Schema' fields get displayed with CSV as the selected File Type.

S3 Reader	0		
Basic Information	Meta Information		
SNS Monitor	Access Key*	Secret Key*	Table*
File Type * CSV	Header	Infer Schema	Limit
Query			

ii. JSON: 'Multiline' and 'Charset' fields get displayed with JSON as the selected File Type.

S3 Reader				Ţ
Basic Information	Meta Information			
SNS Monitor	Access Key*	Secret Key*	Table*	
Query				

- iii. PARQUET: No extra field gets displayed with PARQUET as the selected File Type.
- iv. AVRO: This File Type provides two drop-down menus.
  - 1. Compression: Select an option out of the 'Deflate' and 'Snappy' options.
  - 2. Compression Level: This field appears for the Deflate compression option. It provides 0 to 9 levels via a drop-down menu.

S3 Reader	0			<u>+</u>
Basic Information	Meta Information			
SNS Monitor File Type * AVRO	Access Key*	Secret Key*	Table*	
Query		_		

## 9.1.2. HDFS Reader

- i) Drag & Drop the HDFS Reader component on the Workflow Editor.
- ii) Click on the dragged reader component to open the component properties tabs below:

Search Co	mponent IK				
System	Custom				
Reader	•	HDFS Read.			Q   100%   Q
<b>1</b> 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	🥭 👼				
<b>S</b>	<b>.</b>				
•		HDFS Reader	2		◎ ∎ : ⊥ ⊤
$\bigcap_{i \in \mathcal{O}}$	(B)	Basic Information Meta Inform	ation		
Writer	•	Invocation Type*	Deployment Type spark	Container Image Version 6.5.0-R1	
Transforma	ation 🝷		<u> </u>		
ML	-	Failover Event 💌	Batch Size(min 10)		

- iii) Basic Information: It is the default tab to open for the component while configuring it.
  - a. Select an Invocation type from the drop-down menu to confirm the running mode of the reader component. Select '**Real-Time**' or '**Batch**' from the drop-down menu.
  - b. Deployment Type: It displays the deployment type for the component. This field comes preselected.
  - c. Container Image Version: It displays the image version for the docker container. This field comes pre-selected.
  - d. Failover Event: Select a failover event from the drop-down menu.
  - e. Batch Size (min 10): Provide the maximum number of records to be processed in one execution cycle (Min limit for this field is 10).

3	HDFS Reader		0		e	Î	:	<u>+</u>	Ť
	Basic Information	Meta Inform	ation						
	Invocation Type*	•	Deployment Type	Container Image Version 6.5.0-R1					
	Failover Event	*	Batch Size(min 10)						

- iv) Open the '**Meta Information**' tab and fill in all the connection-specific details for the HDFS Reader.
  - 1. Host IP Address (\*): Hadoop IP address of the host
  - 2. Port (\*): Port number
  - 3. Zone (\*): HDFS Zone location
  - 4. Table (\*): Table or object name which must be read
  - 5. Query: Insert an SQL query (it takes query containing a Join statement as well)
  - 6. Limit: Set a limit for the number of records.
  - 7. File Type: Select a file type from the drop-down menu (CSV, JSON, PARQUET, AVRO are the supported file types).



HDFS Reader			9 🖬	: <u>+</u> <u>+</u>
Basic Information 4 Meta	Information			
Host IP Address*	Port* 8020	Zone*	Table*	_
Query		Limit	File Type *	

i. CSV

Configure the following fields when the selected file type is CSV:

- a. Header- Enable or Disable the Header option to confirm whether the stored data has a header or not
- b. Infer Schema- Enable or disable the Infer Schema option to confirm whether the schema is provided or not
- c. Path- Provide a specific path for the file
- d. Limit- Set limit for the number of rows

File Type CSV	•	Header	🗌 Infer Schema
Path			Limit

ii. JSON

Configure the following fields when the selected file type is JSON:

- a. Multiline-Enable this option, if the stored JSON has line breaks in-between
- b. Charset-Character set encoding, E.g., UTF-8
- c. Path-provide a specific path for the file
- d. Limit-set limit for the number of rows

File Type JSON	•	Multiline	Charset
Path			Limit

iii. PARQUET

Configure the following fields when the selected file type is PARQUET:

- a. Path-provide a specific path for the file
- b. Limit- set limit for the number of rows

File Type			
PARQUET	•	Path	Limit

iv. AVRO



Configure the following fields when the selected file type is AVRO:

- a. Compression: Select a compression option from the drop-down menu out of the '**Deflate**' and '**Snappy**' options.
- b. Compression Level- If the selected Compression option is '**Deflate**' then select the compression level from 0-9 (where 0 and 9 are included).
- c. Path-provide the specific path for the file.
- d. Limit- set limit for the number of rows.

File Type	Compression	Compression Level
AVRO -	Deflate	5 •
Path	Snappy	Limit

v) Selected Columns: The users can select some specific columns from the table to read data instead of selecting a complete table; this can be achieved via the 'Selected Columns' section. Select the columns which you want to read and if you want to change the name of the column, then put that name in the alias name section otherwise keep the alias name the same as of column name and then select a Column Type from the drop-down menu. or

Use 'Download Data' and 'Upload File' options to select the desired columns.

- 1. Upload File: The user can upload the existing system files (CSV, JSON) using the '**Upload File**' icon (file size must be less than 2 MB).
- 2. Download Data (Schema): Users can download the schema structure in JSON format by using the '**Download Data**' icon.

5 Selected Colum	nns				5 🗖 🛆 ×
Name	Alias Name	Column Type	e 🔻	×	
Add New C	Iolumn				

vi) Partition Column- Provide a unique Key column name to partition data in Spark.

6	Partition Columns		
	Name	×	
	Add New Column		

vii) After doing all the configurations click the 'Save Component in Storage' icon to save the component.



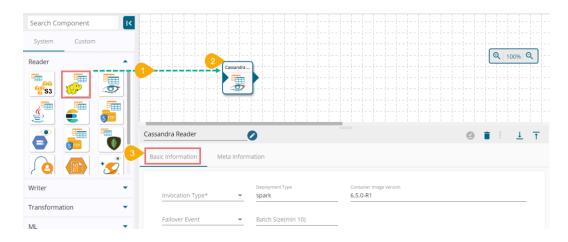


viii) A notification message appears to inform about the component configuration success.



## 9.1.3. Cassandra Reader

- i) Drag & Drop the Cassandra Reader component on the Workflow Editor.
- ii) Click on the dragged reader component to open the component properties tabs below.
- iii) Basic Information: It is the default tab to open for the Cassandra Reader while configuring the component.
  - a. Select an Invocation type from the drop-down menu to confirm the running mode of the reader component. Select 'Real-Time' or 'Batch' from the drop-down menu.
  - b. Deployment Type: It displays the deployment type for the component. This field comes preselected.
  - c. Container Image Version: It displays the image version for the docker container. This field comes pre-selected.
  - d. Failover Event: Select a failover Event from the drop-down menu.
  - e. Batch Size (min 10): Provide the maximum number of records to be processed in one execution cycle (Min limit for this field is 10).



- iv) Open the '**Meta Information**' tab and fill in all the connection-specific details for the Cassandra Reader.
  - 1. Host IP Address (\*): IP Address
  - 2. Port (\*): Host server port number
  - 3. Keyspace (\*): Key for the table
  - 4. Table (\*): Table Name to read data
  - 5. Cluster: Name of the cluster
  - 6. Username (\*): Provide a username for login
  - 7. Password (\*): provide a valid password for login
  - 8. Compression Method: Select a compression method from the drop-down
  - 9. Consistency: the minimum number of Cassandra nodes that must acknowledge the operation
  - 10. Max. Fetch Size: Max number of records per batch
  - 11. Query: Insert an SQL query (it takes query containing a Join statement as well)



Cassandra Reader				٢	Î	:	+	<u>+</u>
Basic Information 4 Me	eta Information							
Host IP Address*	Port* 9040	Keyspace*	Table*					
Cluster	Username*	Password*		Comp	ressio	n Meth	nod	•
Consistency*	Max. Fetch Size	Query						

v) Selected Columns: The users can select some specific columns from the table to read data instead of selecting a complete table; this can be achieved via the 'Selected Columns' section. Select the columns which you want to read and if you want to change the name of the column, then put that name in the alias name section otherwise keep the alias name the same as of column name and then select a Column Type from the drop-down menu. or

Use 'Download Data' and 'Upload File' options to select the desired columns.

- 1. Upload File: The user can upload the existing system files (CSV, JSON) using the '**Upload File**' icon (file size must be less than 2 MB).
- 2. Download Data (Schema): Users can download the schema structure in JSON format by using the 'Download Data' icon.

					27.00
Name	Alias Name	Column Type	*	×	
Add New Column					

vi) After doing all the configurations click the 'Save Component in Storage' icon provided in the component configuration panel to save the component.



vii) A notification message appears to inform about the component configuration success.

# 9.1.4. RDBMS Reader

- i) Drag & Drop the RDBC Reader component on the Workflow Editor.
- ii) Click on the dragged reader component to open the component properties tabs below.
- iii) Basic Information: It is the default tab to open for the RDBMS Reader while configuring the component.
  - a. Select an Invocation type from the drop-down menu to confirm the running mode of the reader component. Select 'Real-Time' or 'Batch' from the drop-down menu.



- b. Deployment Type: It displays the deployment type for the component. This field comes preselected.
- c. Container Image Version: It displays the image version for the docker container. This field comes pre-selected.
- d. Failover Event: Select a failover Event from the drop-down menu.
- e. Batch Size (min 10): Provide the maximum number of records to be processed in one execution cycle (Min limit for this field is 10).

Reader	2 ROBMS Re.	Q 100% Q
	RDBMS Reader     Image: Comparison of Comparis	◎ 🖬 : ⊥ Ŧ
Writer 🔹	Deployment Type Container Image Version Invocation Type*  v spark 6.5.0-R1	
Transformation •		
ML 👻	Failover Event <ul> <li>Batch Size(min 10)</li> </ul>	

- iv) Open the '**Meta Information**' tab and fill in all the connection-specific details for the RDBMS Reader.
  - 1. Host IP Address (\*): IP Address
  - 2. Port (\*): Host server port number
  - 3. Username (\*): Username for login
  - 4. Password (\*): Password for login
  - 5. Database Name (\*): Provide the Database name
  - 6. Driver (\*): Select Database type (MYSQL, MSSQL, Oracle, Postgres)
  - 7. Table Name: Provide the table name to read data
  - 8. Query: Insert an SQL query (it takes query containing a Join statement as well)
  - 9. Limit: Set limit for the number of records

RDBMS Reader					9	Î	:	+	<u>+</u>
Basic Information 4	eta Information								
Host IP Address*	Port*	Userna	ime*	Password*					
Database Name*	Driver*	•	Table Name						
Query			Limit						
		//							

v) Selected Columns: The users can select some specific columns from the table to read data instead of selecting a complete table; this can be achieved via the 'Selected Columns' section. Select the columns which you want to read and if you want to change the name of the column, then put that name in the alias name section otherwise keep the alias name the same as of column name and then select a Column Type from the drop-down menu.



or

Use 'Download Data' and 'Upload File' options to select the desired columns.

- Upload File: The user can upload the existing system files (CSV, JSON) using the 'Upload File' icon (file size must be less than 2 MB).
- 2. Download Data (Schema): Users can download the schema structure in JSON format by using the 'Download Data' icon.

<u></u>	٥
Alias Name Column Type 👻 🗙	

vi) After doing all the configurations click the 'Save Component in Storage' icon to save the component.



vii) A notification message appears to inform about the component configuration success.



### 9.1.4.1. Payload

The RDBMS and Azure readers can consume the data from the previous events and inject it in specific fields of metadata and execute using the payload feature.

Syntax to use payload feature: @data.columnName

- i) Navigate to an RDBMS reader component as a part of a pipeline.
- ii) Open the 'Meta Information' tab for the RDBMS reader.
- iii) Fill the Table Name with the **'@data**.' Syntax.



iv) Open the Preview Data for the RDBMS reader to see the fetched data.



Schulder Geno, even.	
demo_event_1	0 🗉 : 1 T
Basic Info Preview	
Number Of Records: 1	Download as CSV Refresh Data
tableName .	
"covid"	

Note: The user can send payload using the scheduler by giving syntax of the 'key: value pair. In the context of the above example, we can give the following information in the payload field of the scheduler.

{"table\_name": "covid"}

Scheduler	0	9 i ÷ ±
Basic Information Meta In	formation	
Scheduler Name* testScheduler	Paylood(Ex- (Yey"/value)) ("tableName":"covid")	Scheduler Time* 0 0/5 * 1/1 * ? *
	L	Cron Generator
		Minutes Hourly Daily Weekly
		Monthly Yearly
		Every 5 v minute(s) on second 0 v

# 9.1.5. ES Reader

- i) Drag & drop the ES Reader component on the Workflow Editor.
- ii) Click on the dragged reader component to open the component properties tabs below.
- iii) Basic Information: It is the default tab to open for the ES Reader while configuring the component.
  - a. Select an Invocation type from the drop-down menu to confirm the running mode of the reader component. Select '**Real-Time**' or '**Batch**' from the drop-down menu.
  - b. Deployment Type: It displays the deployment type for the component. This field comes preselected.
  - c. Container Image Version: It displays the image version for the docker container. This field comes pre-selected.
  - d. Failover Event: Select a failover Event from the dropdown menu.
  - e. Batch Size (min 10): Provide the maximum number of records to be processed in one execution cycle (Min limit for this field is 10).

Reader		2	ES Reader		Q 100% Q
<b>**</b> \$3					
		ES Reader  Basic Information Meta Inform	0		
Writer	·	Invocation Type*	Deployment Type <b>spark</b>	Container Image Version 6.5.0-R1	
Transform	ation -	Failover Event 👻	Batch Size(min 10)		

- iv) Open the 'Meta Information' tab and fill in all the connection-specific details of ES Reader.
  - 1. Host IP Address (\*): Database Host Address.
  - 2. Port (\*): Database Host port.
  - 3. ES Index Id (\*): Id of the elastic search index.
  - 4. ES Resource Type: Type of the elastic search index (can be given the same as Id).
  - 5. Is Data-Rich True: Required while reading data from indexes having dates.
  - 6. Limit: Set a limit for the number of records.
  - 7. Query: Provide a valid query to fetch data from the dataset.

ES Reader			🥝 🔋 : 🕂 <u>+</u>
Basic Information	nformation		
Host IP Address*	Port* 9200	ES Index Id*	ES Resource Type*
Is Date Rich True	Limit	Query	

v) Selected Columns: The users can select some specific columns from the table to read data instead of selecting a complete table; this can be achieved via the 'Selected Columns' section. Select the columns which you want to read and if you want to change the name of the column, then put that name in the alias name section otherwise keep the alias name the same as of column name and then select a Column Type from the drop-down menu.

Use 'Download Data' and 'Upload File' options to select the desired columns.

- 1. Upload File: The user can upload the existing system files (CSV, JSON) using the 'Upload File' icon (file size must be less than 2 MB).
- 2. Download Data (Schema): Users can download the schema structure in JSON format by using the 'Download Data' 🖸 icon.

Column Type	 ~	

vi) Click the 'Save Component in Storage' 🔽 icon to save the component properties.





vii) A notification message appears to inform about the component configuration success.



### 9.1.6. SFTP Reader

- i) Drag & Drop SFTP Reader component on the Workflow Editor.
- ii) Click on the dragged reader component to open the component properties tabs below.
- iii) Basic Information: It is the default tab to open for the SFTP Reader while configuring the component.
  - a. Select an Invocation type from the drop-down menu to confirm the running mode of the reader component. Select 'Real-Time' or 'Batch' from the drop-down menu.
  - b. Deployment Type: It displays the deployment type for the component. This field comes preselected.
  - c. Container Image Version: It displays the image version for the docker container. This field comes pre-selected.
  - d. Failover Event: Select a failover Event from the drop-down menu.
  - e. Batch Size (min 10): Provide the maximum number of records to be processed in one execution cycle (Min limit for this field is 10).

Reader	57	IP Reader			Q 100% Q
	SFTP Reader Basic Information Me	eta Inform		101100	©∎: <u>⊥</u> ⊤
Writer 💌	Invocation Type*	Ŧ	Deployment Type docker	Container Image Version 6.5.0-R1	
Transformation  ML	Failover Event	•	Batch Size(min 10)	_	

- iv) Open the 'Meta Information' tab and fill in all the connection-specific details for the SFTP Reader.
  - 1. Host IP Address (\*): IP address of SFTP location.
  - 2. Username (\*): Username of SFTP location.
  - 3. Port (\*): Port of SFTP server (default:22).
  - 4. Dynamic Header: Enable this option by putting a checkmark in the box.
  - 5. Authentication (\*): Select an authentication option for login to the SFTP server using the drop-down menu. The field gets displayed based on the Authentication option.
    - a. Password: configure the following fields that appear after selecting the 'Password' option as authentication.
      - i. Password: Provide a valid password in the 'Password' fields to log in.



SFTP Reader	Ø			© i : ÷ .	L
Basic Information	Meta In	formation			
Host IP Address*		Username*	Port* 22	🗹 Dynamic Header	
Authentication Password	<b>•</b>	Password*	Reader Path*	-	_
Channel*	*				

- b. PEM/PPK (file): Select PEM/PPK file database host port for login to SFTP server
  - i. File Name: Provide a file name
  - ii. Choose File: select a PEM/PPK file using this option

FTP Reader			g 🔋 :	<u>+</u> ⊥
Basic Information Me	ta Information			
Host IP Address*	Username*	Port* 22	Dynamic Header	
Authentication PEM / PPK File	File Name	Choose File		
Reader Path*		Channel*	<b>•</b>	

- 6. Reader Path (\*) complete path (absolute path) for the folder (from where the new files need to be read)
- 7. Channel (\*) the supported channel is SFTP.

Basic Information	eta Information			
Host IP Address*	Username*	Port* 22	Dynamic Header	
Authentication	Reader Path*		Channel*	•

v) Selected Columns: The users can select some specific columns from the table to read data instead of selecting a complete table; this can be achieved via the 'Selected Columns' section. Select the columns which you want to read and if you want to change the name of the column, then put that name in the alias name section otherwise keep the alias name the same as of column name and then select a Column Type from the drop-down menu. or

Use 'Download Data' and 'Upload File' options to select the desired columns.

- 1. Upload File: The user can upload the existing system files (CSV, JSON) using the 'Upload File' icon (file size must be less than 2 MB).
- 2. Download Data (Schema): Users can download the schema structure in JSON format by using the 'Download Data' icon.



Selected Columns					- (5) 🖸 🛆 🗙 -
Name	Allas Name	Column Type	*	×	
Add New Column					

vi) After doing all the configurations click the 'Save Component in Storage' icon to save the component.



vii) A notification message appears to inform about the component configuration success.

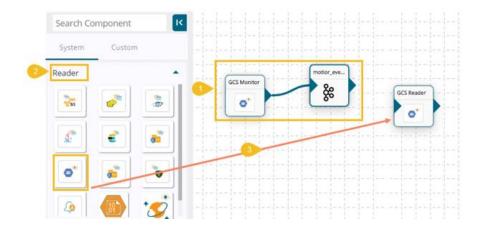


# 9.1.7. GCS Reader

GCS Reader pulls data from the GCS Monitor, so the first step is to implement GCS Monitor.

Note: The users can refer to the GCS Monitor section of this document for the details.

- i) Navigate to the Pipeline Workflow Editor page for an existing pipeline workflow with GCS Monitor and Event component.
- ii) Open the Reader section of the Component Pallet.
- iii) Drag the GCS Reader to the Workflow Editor.



- iv) Click on the dragged GCS Reader component to get component properties tabs below.
- v) Basic Information: It is the default tab to open for the GCS Reader while configuring the component.a. Select an Invocation type from the drop-down menu to confirm the running mode of the
  - reader component. Select 'Real-Time' or 'Batch' from the drop-down menu.
  - b. Deployment Type: It displays the deployment type for the component. This field comes preselected.
  - c. Container Image Version: It displays the image version for the docker container. This field comes pre-selected.
  - d. Failover Event: Select a failover Event from the drop-down menu.



e. Batch Size (min 10): Provide the maximum number of records to be processed in one execution cycle (Min limit for this field is 10).

GCS Monitor		monitor.ev				<b>2</b>   100' 	% Q
GCS Reader		2	100000	۲	Î	:	<u>t</u>
Basic Information M	eta Inform	ation					
Invocation Type*	•	Deployment Type docker	Container Image Version 6.5.0-R1				
Failover Event	-	Batch Size(min 10)					

- vi) Provide the following Meta Information for the GCS Reader component:
  - a. Bucket Name: Destination bucket name (Copy Bucket Name of the GCS Monitor)
  - b. Directory Path: Destination copy folder path (Copy Directory Path of the GCS Monitor)
  - c. Choose File: Upload the same Service account keys using the JSON file
  - d. File Name: It displays the file name of the uploaded file

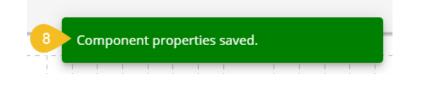
GCS Reader	0		2 🖬 : 🕂 土
Basic Information	Meta Information		
Bucket Name*	Directory Path*	Choose File	File Name*

Note: Provide the same destination configuration information that was used in the GCS Monitor.

vii) After doing all the configurations click the 'Save Component in Storage' icon to save the component.



viii) A message appears to confirm the component properties update.

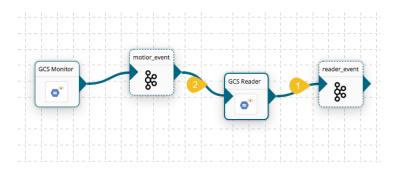


#### 9.1.7.1. GCS Reader in a Workflow

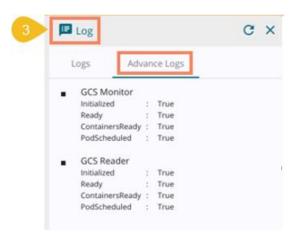
- i) Create a new Event to store all the read data and connect it with the dragged GCS Reader component.
- ii) Connect the GCS Reader to the existing GCS Monitor Pipeline to create the



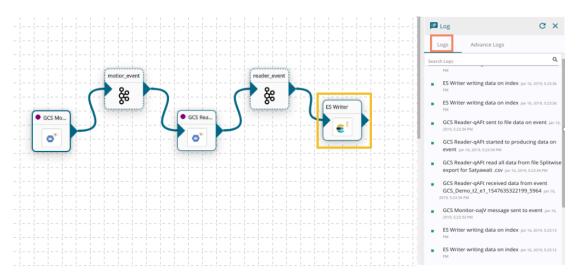
#### below-given workflow.



iii) Open the Advanced Log Tab to check the component status.



- iv) Upload a CSV or JSON file on the monitoring location and check the Logs section displaying the ongoing operations.
- v) Connect a Writer component to store the processed data (E.g., the ES Writer component is used in this pipeline workflow).



Note: GCS Reader component gets a message from an Event component that was initially sent by the GCS Monitor to the Event, so the users must configure the GCS Monitor component before using the GCS Reader component.



# 9.1.8. SFTP Stream Reader

- i) Drag & Drop the SFTP Stream Reader component on the Workflow Editor.
- ii) Click on the dragged reader component to open the component properties tabs below.
- iii) Basic Information: It is the default tab to open for the SFTP Stream Reader while configuring the component.
  - a. Select an Invocation type from the drop-down menu to confirm the running mode of the reader component. Select 'Real-Time' or 'Batch' from the drop-down menu.
  - b. Deployment Type: It displays the deployment type for the component. This field comes preselected.
  - c. Container Image Version: It displays the image version for the docker container. This field comes pre-selected.
  - d. Failover Event: Select a failover Event from the drop-down menu.
  - e. Batch Size (min 10): Provide the maximum number of records to be processed in one execution cycle (Min limit for this field is 10).

Reader		SFTP Stra		
	SFTP Stream Reader Basic Information Meta	Information		❷ 🖬 ÷ ⊥ T
Writer   Transformation	Invocation Type*	Deployment Type     docker     Batch Size(min 10)	Container Image Version 6.5.0-R1	

- iv) Open the 'Meta Information' tab and provide the connection-specific information.
  - 1. Host IP Address (\*): IP address of SFTP location.
  - 2. Username (\*): Username of SFTP location.
  - 3. Port (\*): Port of SFTP server (default:22).
  - 4. Dynamic Header: Enable this option by putting a checkmark in the box.
  - 5. Password (\*): Provide a valid password in the 'Password' field to log in.
  - 6. Reader Path (\*): Complete path (absolute path) for the folder (from where the new files need to be read)
  - 7. Channel (\*): The supported channel is SFTP.
  - 8. File Type (\*): Select a file from the drop-down menu.

SFTP Stream Reader	0		o 🖬 🗄	<u>+</u> <u>+</u>
Basic Information 4	Meta Information			
Host IP Address*	Username*	Port* 22	Dynamic Header	
Password*	Reader Path*		Channel*	*
File Type *	·			

Note: The user needs to configure some extra fields based on the selected File Type as displayed below.



- a. The following fields appear if the selected file type is CSV
  - 1. Header: Put a checkmark in the box to get Column names/ headers from the CSV file.
  - 2. Infer Schema: Put a checkmark in the box to get inferring datatypes.
  - 3. Multiline: Enable this option to get a column with multiple lines in a single cell of the data. E.g., the address column can have multiple lines in a single cell.

	pe -		
CSV	<b>•</b>	🗖 Infor Schomo	Multiline

- b. The following fields appear if the selected file type is JSON
  - 1. Charset-Character set encoding, E.g., UTF-8
  - 2. Multiline: Enable this option to get a column with multiple lines in a single cell of the data.

File Type *		
JSON 🔻	Charset	Multiline

- v) Selected Columns: The users can select some specific columns from the table to read data instead of selecting a complete table; this can be achieved via the 'Selected Columns' section. Select the columns which you want to read and if you want to change the name of the column, then put that name in the alias name section otherwise keep the alias name the same as of column name and then select a Column Type from the drop-down menu. or
  - Use 'Download Data' and 'Upload File' options to select the desired columns.
  - 3. Upload File: The user can upload the existing system files (CSV, JSON) using the 'Upload File' icon (file size must be less than 2 MB).
  - 4. Download Data (Schema): Users can download the schema structure in JSON format by using the 'Download Data' icon.

Selected Columns					5 🗇 🕹
Name	Alias Name	Column Type	*	×	
Add New Column					

vi) After doing all the configurations click the 'Save Component in Storage' icon to save the component.



vii) A notification message appears to inform about the component configuration success.





# 9.1.9. MongoDB Reader

- i) Drag & Drop the Mongo DB Reader on the Workflow Editor.
- ii) Click on the dragged reader component to open the component properties tabs below.
- iii) Basic Information: It is the default tab to open for the Mongo DB reader while configuring the component.
  - a. Select an Invocation type from the drop-down menu to confirm the running mode of the reader component. Select 'Real-Time' or 'Batch' from the drop-down menu.
  - b. Deployment Type: It displays the deployment type for the component. This field comes preselected.
  - c. Container Image Version: It displays the image version for the docker container. This field comes pre-selected.
  - d. Failover Event: Select a failover Event from the drop-down menu.
  - e. Batch Size (min 10): Provide the maximum number of records to be processed in one execution cycle (Min limit for this field is 10).



- iv) Open the 'Meta Information' tab and fill in all the connection-specific details of Mogo DB.
  - 1. Connection Type: Select either of the connection type choices out of 'Standard' and 'SRV'.
  - 2. Host IP Address (\*): Hadoop IP address of the host.
  - 3. Port (\*): Port number (It appears only with the 'Standard' Connection Type).
  - 4. Username (\*): Provide username.
  - 5. Password (\*): Provide a valid password to access the Mongo DB.
  - 6. Database Name (\*): Provide the name of the database from where you wish to read data.
  - 7. Collection Name (\*): Provide the name of the collection.
  - 8. Query: Insert an SQL query (it takes a query containing a Join statement as well).
  - 9. Limit: Set a limit for the number of records.
  - 10. Additional Parameters: Provide the additional parameters.



Mongodb Reader	0		9 📋 :	<u>+</u> <u>↓</u>
Basic Information 4 Meta Ir	nformation			
Connection Type* 💌	Host*	User Name*	Password*	
Database Name*	Collection Name*	Query	Limit	
Additional Parameters				

The Meta Information fields vary based on the selected 'Connection Type' option.

i. Meta Information Tab with Standard as Connection Type

Mongodb Reader		Ø		g 🔋 :	+	<u>+</u>
Basic Information	Meta Ir	nformation				
Connection Type* Standard	•	Port*	Host*	User Name*		
Password*		Database Name*	Collection Name*	Query		
Limit		Additional Parameters				

ii. Meta Information Tab with SRV as Connection Type

sic Information Meta II	nformation		
Connection Type* SRV	Host*	User Name*	Password*
Database Name*	Collection Name*	Query	Limit

 v) Selected Columns: The users can select some specific columns from the table to read data instead of selecting a complete table; this can be achieved via the 'Selected Columns' section. Select the columns which you want to read and if you want to change the name of the column, then put that name in the alias name section otherwise keep the alias name the same as of column name and then select a Column Type from the drop-down menu.

or

Use 'Download Data' and 'Upload File' options to select the desired columns.

1. Upload File: The user can upload the existing system files (CSV, JSON) using the '**Upload File**' icon (file size must be less than 2 MB).



2. Download Data (Schema): Users can download the schema structure in JSON format by using the '**Download Data**'



vi) After doing all the configurations click the 'Save Component in Storage' icon to save the component.



vii) A notification message appears to inform about the component configuration success.



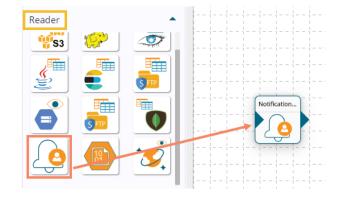
## 9.1.10. Notification Subscriber

Notification Subscriber Component as the name suggests is meant to subscribe to the messages published as a notification. This can be very helpful in acting in scenarios where the immediate reaction is required in some conditions.

Example: A pipeline is monitoring an IoT sensor that sends the temperature of a machine. If the temperature reaches a certain threshold; the process engineer is to be notified immediately to maintain the same to stabilize the machine. Here, the user can subscribe to the same SNS and further use it in a pipeline for sending this directly to the respective dashboard.

The steps to configure the Notification Subscriber component is described in this section.

- i) Navigate to the Pipeline Editor.
- ii) Drag the Notification Subscriber from the Reader section provided under the Components Pallet and drop it to the Pipeline Workspace.



- iii) Click the dragged Notification Subscriber component to get the Configuration fields:
- iv) The Basic Information tab opens by default.



- a. Select the invocation type (Real-Time/ Batch).
- b. Deployment Type: It comes preselected based on the component.
- c. Container Image Version: It comes preselected based on the component.
- d. Failover Event: Select a failover Event from the drop-down menu.
- e. Batch Size (min 10): Provide the maximum number of records to be processed in one execution cycle (Min limit for this field is 10).

			<b>Q</b>   100%   <b>Q</b>
Notificatio			
			© ∎ : <u>⊥</u> ⊤
Basic Information Meta Inform	Deployment Type	Container Image Version	
Invocation Type*   Failover Event	docker Batch Size(min 10)	6.5.0-R1	

- v) Open the 'Meta Information' tab and provide the connection-specific fields.
  - a. Notification Type: Select a notification type using the drop-down menu. The provided options are:
    - i. SNS
    - ii. Google Pub/Sub
  - b. Based on the selected Notification Type the remaining configuration fields get displayed.
  - c. The user needs to complete the configuration process for the selected notification type and click the 'Save Component in Storage' clicon to complete the configuration process for the Notification Subscriber.

Notification Subscriber	Ø	$\frac{+}{+}$ $\pm$
Basic Information Meta Information		
Notification Type*		

### 9.1.10.1. SNS as Notification Type

- i) Navigate to the Meta Information tab provided for the Notification Subscriber.
- ii) Select the 'SNS' as the Notification Type option using the drop-down menu.
- iii) The following configuration fields appear:
  - a. Access Key: Access keys consist of two parts: an access key ID (for example, AKIAIOSFODNN7EXAMPLE) can be retrieved from AWS key management.
  - Secret Key: A secret access key (for example, wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY) can also be retrieved from the AWS key management.



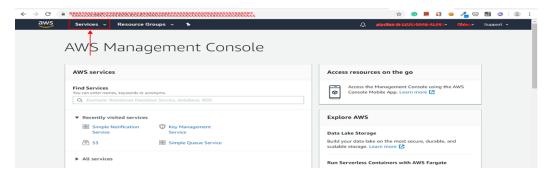
- c. SQS URL: The field gets filled after the topic creation. Navigate to Topics and then click the desired topic.
- d. Region: Select an option from the drop-down menu where the topic has been created.

Notification Subscribe	er 🖉		g 🖬 :	$\frac{1}{2}$ $\frac{1}{2}$
Basic Information	Meta Information			
Notification Type*	Access Key*	Secret Key*	Sqs URL	
Region	<b>•</b>			

- iv) The user needs to follow some more steps to create and subscribe to a topic. The below-given description explains those steps:
  - a. Navigate to the URL: <u>https://aws.amazon.com/console/</u>
  - b. Click the 'Sign In to the Console' option.

C https://aws.amazon.com/console/	🗞 🖈 😨 Contact Sales Support = English = My /	Account 🗸 Sign In to the Console
cts Solutions Pricing Documentati	on Learn Partner Network AWS Marketplace Explore More Q	
PRODUCTS & SERVICES	AWS Management Console	
AWS Console		
RELATED LINKS	Access and manage Amazon Web Services through a simple and intuitive web-based user interface. You can also use the AWS Console	t Started with AWS for Free
Documentation	mobile app to quickly view resources on the go	Create a Free Account
Articles & Tutorials		Or Sign In to the Console
Developer Tools	Bereive	twelve months of access to the AWS
Public Data Sets	Free	Usage Tier and enjoy AWS Basic
Amazon Machine Images (AMIs)		port features including, 24x7x365 omer service, support forums, and
Videos & Webinars		more.
What's New		
	-	<u> </u>

- c. Sign in with your credentials.
- d. The AWS Management Console opens.
- e. Click the 'Services' option from the header drop-down.
- f. Search SQS service using the 'Find Services' space.



- g. The Amazon SNS page opens.
- h. Click the 'Create New Queue' option.



iours.					o 🔏 🖾 🕲 🔮	
aws Services - Resource Groups -	•			@ 6876-9948-4674 -	N. Virginia • Support •	
Create New Queue Actions 👻					3	٥
Filter by Prefix: Q, Enter Text			×		$\ \leqslant ~<~1$ to 5 of 5 items $~>~$	Ы
Name	- Queue Type	Content-Based Deduplication -	Messages Available -	Messages in Flight -	Created	
2335gapa	Standard	N/A	0	0	2019-05-20 16:29:05 GMT+05	30
XXX	Standard	NA	0	0	2019-07-09 11:32:07 GMT+05	30
RANAD.	Standard	N/A	0	0	2019-07-08 12:41:22 GMT+05	30
approceptsi	Standard	NA	0	0	2019-05-13 15:47:57 GMT+05	30
Station	Standard	NA	0	0	2019-07-09 18:01:27 GMT+05	30

- i. Click the 'Quick-Create Queue' option after giving SQS name.
  - s, choose Configure Queue.



- j. Click on the created SQS to see the details.
- k. Copy the SQS URL from there.

new			Standard	N/	/A	0
newsqs			Standard	N/	A	0
queue13	05		Standard	N/	A	0
QS Queue s	elected				000	
Details	Permissions	Redrive Policy	Monitoring	Tags	Encryption	Lambda Trigg
	Name:	queue1305				
	URL:	https://sqs.us-east-1.anx	CR7C0040	4674/999994	WEX.	
		https://odo.do/ 0dot //dil				
		arn:aws:sqs:us-east-		-	nan l	
	ARN:		1000404673/046663730	-		
	ARN: Created:	arn:aws:sqs:us-east-	<b>600401674:queue130</b> T+05:30	-		
	ARN: Created:	am:aws:sqs:us-east- 2019-05-13 15:47:57 GM 2019-05-13 15:48:30 GM	<b>600401674:queue130</b> T+05:30	-		
	ARN: Created: Last Updated:	am:aws:sqs:us-east- <b>163</b> 2019-05-13 15:47:57 GM 2019-05-13 15:48:30 GM 0 seconds	<b>600401674:queue130</b> T+05:30	-	NUCK.	

- I. Use right-click on the created SQS.
- m. Select the 'Subscribe Queue to SNS Topic' to subscribe to the desired SNS Topic.



	aws Serv	vices 🗸 🛛 Resource Grou	ıps v 🍾		
	new		Standard	N/A	
	newsqs		Standard	N/A	
	queue1305	Send a Message	- andard	N/A	
1 SQ S	Queue selected	View/Delete Messages			0
De	etails Permiss	s Configure Queue Add a Permission	ring	Tags Encryption	Lambda 1
		Purge Queue Delete Queue	ı/687699484 ∴queue1305	674/queue1305	
	C	Subscribe Queue to SNS To			
	,	Delay. 0 seconds	a Function		
Cor	Queue ntent-Based Dedupli	e Type: Standard ication: N/A			

- n. The 'Subscribe to a Topic' window opens.
- o. Provide the following information:
  - i. Topic Region
  - ii. Choose a Topic
  - iii. Topic ARN
- p. Click the 'Subscribe' option.

Subscribe to a	Горіс	×
	om the <i>Choose a Topic</i> drop-down or enter a topic's ARN in the <i>Topic ARN</i> text box an o allow your queue(s) to receive SNS notifications from the topic and to subscribe your	
Topic Region (1) Choose a Topic (1) Topic ARN (1)	US East (N. Virginia)   Select a topic from the list of your topics	
L	Cancel Subscrit	

#### Note:

- a. Choose the Topic from where your messages are being published.
- b. SNS Region: The same can be seen in the SNS ARN URL. The user needs to use the same.

# 9.1.10.2. Google Pub/Sub as the selected Notification Type

- i) Navigate to the Meta Information tab provided for the Notification Subscriber.
- ii) Select the 'Google Pub/Sub' as the Notification Type option using the drop-down menu.



Notification Subscriber	0			Ø	<b>i</b> :	<u>+</u>	Ť	
Basic Information	Meta Information							
Notification Type* Google Pub/Sub	▼ Pro	iect ID*	Subscription Name*	Choose Fi	le			
Authentication File Na	me							

- iii) The user needs to follow some more steps to get the Pub/Subtopic ID
- iv) Navigate to the <a href="https://console.cloud.google.com/">https://console.cloud.google.com/</a>
- v) Log in with your credentials.
- vi) The Landing page opens.
- vii) Click on the marked area.
- viii) Create a Project or use the desired one.

C https://console.cloud.google.com/home/d addboard/google.com/home/d add					😁 📕 🎝	0 🐴		8 0	
	💲 project1 👻				2	9	0	•	6
DASHBOARD ACTIVITY								🖍 cus	TOMIZE
									- 1
Project info	:	API APIS	:	Google C	loud Platform	status			:
Project name project1		Requests (requests/sec)	1.0	All services r	normal				
Project ID Reciect&colficte1			0.8	→ Go to Cloud s	status dashboard				
Project number		No data is available for the selected time	frame. <sup>0.6</sup>	0					
→ Go to project settings			0.4	No sign of an	ny errors. Have you :	set up Erro	r Repor	rting?	•

#### ix) Search for the Pub/Sub-option in the Search bar.

≡ Google Cloud Platform ♣ NewProject01 ◄	Q. Pub/sub	5 9 9 🕦 i 🚱
DASHBOARD ACTIVITY	Pub/Sub     Cloud Pub/Sub	CUSTOMIZE

- x) Create a Topic using the following steps:
  - a. Click the 'CREATE TOPIC' option.
  - b. Provide a name for the new topic
  - c. Select the 'Google-managed key' as the Encryption option.
  - d. Click the 'CREATE TOPIC' option.

4	Topics 1	T DELETE		
p	= Filter table		<b>⊘ C Ⅲ</b> 5	Select a
	Topic nam	Encryption		PERMISSIO

xi) The user gets either of the following screens:



## a. While using an existing Topic ID

Topic ID	
projects/newproject	01-246106/topics/newtopic
Encryption key	
Google-managed key	/
Labels	
-	
	SNAPSHOTS

## Alternatively, while using the newly created Topic ID

Ŧ	Filter table			0 C	
	Topic name ↑	Encryption	Topic ID	Labels	
	newtopic	Google-managed	D REPORT OF THE PROPERTY OF TH		:

## b. Create Auth-JSON for the current Project.

÷ ÷	C Anttps://console.clo Google Cloud Platform	dd.google.com/cloudpubsub/topic/detail/newtopic?project=newproject01-246106	🔹 🏠
♠	Home	+ PUBLISH MESSAGE 📩 IMPORT 👻 🏦 EXPORT 👻 👕 DELETE	
Ŷ	Marketplace	1	newtopic
<b>T</b> anal	Billing		PERMISSION
API	APIs & Services	Dashboard	Edit or delete Member" to g
Ť	Support	Library	Show
Θ	IAM & admin	Credentials	
•	Getting started		= Filter tre
۲	Security	,	Role / Member
COMP	PUTE		

- c. Create a Service account key by following the given steps:
  - i. Click the 'Service account key' option from the Credentials tab.



	Google Cloud Platform 🔹 NewProject01 👻	۹
API	Credentials	
4 <b>Ç</b> Þ	Credentials OAuth consent screen Domain verification	
ᇤ	Create credentials 👻 Delete	
0+	API key Identifies your project using a simple API key to check quota and access OAuth client ID Requests user consent so your app can access the user's data	authentication documentation .
	Service account key Enables server-to-server, app-level authentication using robot accounts	ite V Service account
	Help me choose Asks a few questions to help you decide which type of credential to use	

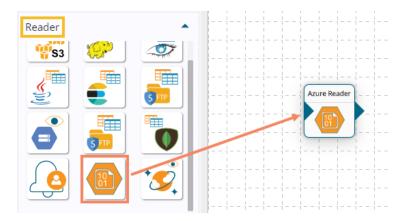
- ii. Select a Service account using the drop-down menu.
- iii. Download the JSON file using the radio button.
- iv. Click the 'Create' option.

Use the copied Topic ID or the service account as metadata input.

=	Google Cloud Platform	🗣 NewProject01 👻
←	Create service account ke	:y
Servio	e account	
Sele	ct	-
	pe oads a file that contains the private key. Sto overed if lost.	re the file securely because this key ca
I JS	SON commended	
O P Fo	12 or backward compatibility with code using th	e P12 format
Crea	Cancel	

# 9.1.11. Azure Reader

i) Drag and drop the Azure Reader component to the Workflow Editor.



ii) Click on the dragged Azure Reader to get the component properties tabs.



- iii) **Basic Information**: It is the default tab to open for the component while configuring the component.
  - a. Invocation Type: Select an invocation mode out of '**Real-Time**' or '**Batch**' using the drop-down menu.
  - b. Deployment Type: It displays the deployment type for the reader component. This field comes pre-selected.
  - c. Container Image Version: It displays the image version for the docker container. This field comes pre-selected.
  - d. Failover Event: Select a failover Event from the drop-down menu.
  - e. Batch Size (min 10): Provide the maximum number of records to be processed in one execution cycle (Min limit for this field is 10).

Search Component				
System Custom				
Reader	Aure Blob.			Q 100% Q
	Azure Blob Reader Basic Information Meta Inform	0		◎ 🗎 : ⊥ Ŧ
Writer •	Invocation Type*	Deployment Type spark	Container Image Version 6.5.0-R1	
Transformation •	Failover Event 👻	Batch Size(min 10)		

- iv) Open the '**Meta Information**' tab and fill in all the connection-specific details for the Azure Reader.
  - 1. Read Using (\*): Select an option from the drop-down menu to connect to data. The supported options are '**Connection String**' and '**Secret Key**'.
  - 2. Account Name (\*): Name of the Azure account.
  - 3. Account Key (\*): Provide Account Key to login.
  - 4. Container (\*): Container details.
  - 5. Read Directory: Enable this option by putting a checkmark in the box to be read from the Directory.
  - 6. Blob Name: Provide a blob name. The Blob field appears if the '**Read Directory**' option is not enabled.
  - 7. Query: Insert relevant query.
  - 8. Limit: Set a limit for the number of records.
  - 9. File Type (\*): Select a file type from the drop-down menu (CSV, JSON, PARQUET, AVRO are the supported file types).

Azure Reader	Ø			🙆 🛢 : 🕂 土
Basic Information	Meta Inform	ation		
Read Using*	*	Account Name*	Account Key*	Container*
Read Directory		Query		Limit
File Type *	•			

The configuration fields may vary based on the selection of the file type.

i. CSV



Configure the following fields when the selected file type is CSV:

- a. Header- Enable or Disable the Header option to confirm whether the stored data has a header or not.
- b. Infer Schema- Enable or disable the Infer Schema option to confirm whether the schema is provided or not.
- c. Path- Provide a specific path for the file.

File Type * CSV	🗌 Header	🔲 Infer Schema
Path		

ii. JSON

Configure the following fields when the selected file type is JSON:

- a. Multiline-Enable this option, if the stored JSON has line breaks in-between
- b. Charset-Character set encoding, E.g., UTF-8
- c. Path-provide a specific path for the file.

File Type * JSON ~	Multiline	Charset
Path		

#### iii. PARQUET

Configure the following fields when the selected file type is PARQUET:

a. Path-provide a specific path for the file.

File Type * PARQUET -	Path

iv. AVRO

Configure the following fields when the selected file type is AVRO:

- a. Compression: Select a compression option from the drop-down menu out of the 'Deflate' and 'Snappy' options.
- b. Compression Level- If the selected Compression option is 'Deflate' then select the compression level from 0-9 (where 0 and 9 are included).
- c. Path-provide the specific path for the file.
- d. Limit- set limit for the number of rows.

File Type * AVRO	*	Compression	*	Compression Level	*
Path					

v) Selected Columns: There is also a section for the selected columns in the Meta Information tab if the user can select some specific columns from the table to read data instead of selecting a complete table so this can be achieved by using the 'Selected Columns' section. Select the columns which you want to read and if you want to change the name of the column, then put that name in the alias name section otherwise keep the alias name the same as of column name and then select a Column Type from the drop-down menu.



Or

Use 'Download Data' and 'Upload File' options to select the desired columns.

- Upload File: The user can upload the existing system files (CSV, JSON) using the 'Upload File' icon (file size must be less than 2 MB).
- 2. Download Data (Schema): Users can download the schema structure in JSON format by using the 'Download Data' () icon.

Alias Name	Column Type	• ×	
	Alias Name	Alias Name Column Type	Alias Name Column Type 👻 🗙

- vi) Click the 'Save Component in Storage' icon after doing all the configurations to save the reader component.
- vii) A notification message appears to inform you that the component properties have been saved.



# 9.1.12. Azure Cosmos DB Reader

i) Drag and drop the Azure Cosmos DB Reader component to the Workflow Editor.



- ii) Click on the dragged Azure Cosmos DB Reader to get the component properties tabs.
- iii) **Basic Information**: It is the default tab to open for the component while configuring the component.
  - a. Invocation Type: Select an invocation mode out of '**Real-Time**' or '**Batch**' using the drop-down menu.
  - b. Deployment Type: It displays the deployment type for the reader component. This field comes pre-selected.
  - c. Container Image Version: It displays the image version for the docker container. This field comes pre-selected.
  - d. Failover Event: Select a failover Event from the drop-down menu.
  - e. Batch Size (min 10): Provide the maximum number of records to be processed in one execution cycle (Min limit for this field is 10).



Reader		×		Q 100% Q
	Azure CosmosDB Reader Basic Information Meta Info	ormation		© ∎ : <u>⊥</u> †
Writer  Transformation	Invocation Type*	Deployment Type spark	Container Image Version 6.5.0-R1	
ML	Failover Event -	Batch Size(min 10)	_	

- iv) Open the '**Meta Information**' tab and fill in all the connection-specific details for the Azure Cosmos DB Reader.
  - 1. Endpoint (\*): Provide endpoint information.
  - 2. Master Key (\*): Provide the master key.
  - 3. Database Name (\*): Provide the name of the database.
  - 4. Container Name (\*): Container details.
  - 5. Query: Insert relevant query.
  - 6. Limit: Set a limit for the number of records.

Azure CosmosDB Rea	der 🖉			0	Î	:	+	<u>+</u>
Basic Information	Meta Information							
Endpoint*		-	MasterKey*					
Database Name*	Container Name*	_	Query					
Limit								

v) Selected Columns: There is also a section for the selected columns in the Meta Information tab if the user can select some specific columns from the table to read data instead of selecting a complete table so this can be achieved by using the 'Selected Columns' section. Select the columns which you want to read and if you want to change the name of the column, then put that name in the alias name section otherwise keep the alias name the same as of column name and then select a Column Type from the drop-down menu. Or

Use 'Download Data' and 'Upload File' options to select the desired columns.

- 3. Upload File: The user can upload the existing system files (CSV, JSON) using the 'Upload File' icon (file size must be less than 2 MB).
- 4. Download Data (Schema): Users can download the schema structure in JSON format by using the 'Download Data' icon.

Selected Columns				□ <b>△</b> ×
Name	Alias Name	Column Type	• ×	
Add New Column				

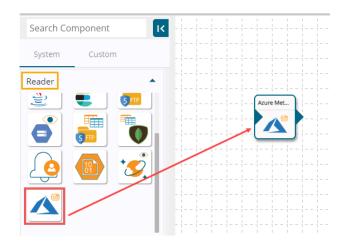


- vi) Click the 'Save Component in Storage' Cicon after doing all the configurations to save the reader component.
- vii) A notification message appears to inform you that the component properties have been saved.

Component properties saved.

# 9.1.13. Azure Metadata Reader

i) Drag and drop the Azure Metadata Reader component to the Workflow Editor.



- ii) Click on the dragged Azure Metadata Reader to get the component properties tabs.
- iii) **Basic Information**: It is the default tab to open for the component while configuring the component.
  - a. Invocation Type: Select an invocation mode out of '**Real-Time**' or '**Batch**' using the drop-down menu.
  - b. Deployment Type: It displays the deployment type for the reader component. This field comes pre-selected.
  - c. Container Image Version: It displays the image version for the docker container. This field comes pre-selected.
  - d. Failover Event: Select a failover Event from the drop-down menu.
  - e. Batch Size (min 10): Provide the maximum number of records to be processed in one execution cycle (Min limit for this field is 10).





iv) Open the '**Meta Information**' tab and fill in all the connection-specific details for the Azure Metadata Reader.

Azure Metadata Read		nformation		© î :	<u>+</u> ↑	<u>+</u>
Read Using*	•	Account Name*	Container*	Path Type -null-	•	
Blob Name						

Based on the selected option using the field '**Read Using'** the displayed fields may change. Available option (\*): Select either of the options out of '**Shared Access Signature'** or '**Secret Key'** 

The Meta Information tab when the selected Read Using option is 'Shared Access Signature':

- 1. Read Using: the selected option is 'Shared Access Signature'
- 2. Shared Access Signature: the user needs to provide the shared access signature.
- 3. Account Name (\*): Provide the name of the account.
- 4. Container (\*): Container details.
- 5. Path Tye: Select a Path type from the given choices (Null, Directory Path, Blob Name)
- 6. Blob Name: Provide a blob name.

		g 🔋 :	<u>+</u> ⊥
Meta Information			
<ul> <li>Shared Access Signa</li> </ul>	Account Name*	Container*	
▼ Blob Name			
	Meta Information	Meta Information  Shared Access Signa Account Name*	Image: Container*       Image: Container*         Shared Access Signa       Account Name*       Container*

The Meta Information tab when the selected Read Using option is 'Secret Key':

- 1. Read Using: the selected option is 'Secret Key'.
- 2. Account Key (\*): Provide account key.
- 3. Account Name (\*): Provide Account Name.
- 4. Container (\*): Container details.
- 5. Path Type: Select a Path type from the given choices (Null, Directory Path, Blob Name)
- 6. Blob Name: Provide a blob name.



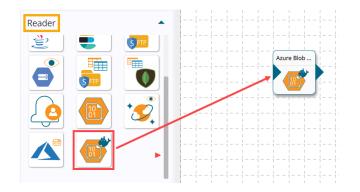
Azure Metadata Read	ler	0		<b>0</b> :	+	<u>+</u>
Basic Information	Meta Ir	formation				
Read Using* Secret Key	-	Account Key*	Account Name*	Container*		
Path Type -null-	•	Blob Name				

- v) Click the 'Save Component in Storage' Cicon after doing all the configurations to save the reader component.
- vi) A notification message appears to inform you that the component properties have been saved.



## 9.1.14. Azure Blob Reader

i) Drag and drop the Azure Blob Reader component to the Workflow Editor.



- ii) Click on the dragged Azure Blob Reader to get the component properties tabs.
- iii) **Basic Information**: It is the default tab to open for the component while configuring the component.
  - a. Invocation Type: Select an invocation mode out of '**Real-Time**' or '**Batch**' using the drop-down menu.
  - b. Deployment Type: It displays the deployment type for the reader component. This field comes pre-selected.
  - c. Container Image Version: It displays the image version for the docker container. This field comes pre-selected.
  - d. Failover Event: Select a failover Event from the drop-down menu.
  - e. Batch Size (min 10): Provide the maximum number of records to be processed in one execution cycle (Min limit for this field is 10).



Reader	2 Azure Blob			<u>୍</u>   100%   ସ୍
2 💶 💷				
e 👼 💿				
	Azure Blob Reader	0		Ø ∎ : ± T
	Basic Information Meta Inform	nation		
Writer -		Deployment Type	Container Image Version	
Transformation 🔹	Invocation Type* 👻	docker	6.5.0-R1	
MI	Failover Event 👻	Batch Size(min 10)		

iv) Open the '**Meta Information**' tab and fill in all the connection-specific details for the Azure Blob Reader.

Based on the selected option using the field '**Read Using'** the displayed fields may change. Available option (\*): Select either of the options out of '**Shared Access Signature'**, '**Secret Key' or 'Principal Secret'** 

Azure Blob Reader	0			0	i :	<u>↓</u> ↑	<u>+</u>
Basic Information Meta Inform	nation						
Shared Access Signature	Shared Access Signature*	Account Name*		Container*			
Secret Key Principal Secret	File Type* 💌						
Selected Columns Source Field	Destination Field	Column Type	×		• >	<	
Add New Column							

The Meta Information tab when the selected Read Using option is 'Shared Access Signature':

- 1. Read Using: the selected option is 'Shared Access Signature'
- 2. Shared Access Signature: the user needs to provide the shared access signature.
- 3. Account Name (\*): Provide the name of the Azure Blob account.
- 4. Container (\*): Container details.
- 5. Read Directory: Select this option using the check box to read the blob name from the directory.
- 6. Blob Name: Provide a blob name. (It appears if the '**Read Directory**' option is not check marked).
- 7. File Type: Select a file type from the drop-down menu. (The supported file types are CSV, JSON, PARQUET, AVRO)

Azure	Blob Reader				Ø	<u>↓</u> ↑	<u> </u>
Basio	c Information Meta Inform	ation					
	Read Using* Shared Access Signat 👻	Shared Access Signature*	Account Name*		Container*		_
	Read Directory	Blob Name	File Type*	•			



Note: If the 'Read Directory' option is marked/enabled the 'Blob Name' field will not appear.

Azure Blob Reader	Ø		g i :	<u>+</u> <u>↓</u>
Basic Information Meta Information	mation			
Read Using* Shared Access Signat ▼	Shared Access Signature*	Account Name*	Container*	
Read Directory	File Type*			

The Meta Information tab when the selected Read Using option is 'Secret Key':

- 1. Read Using: the selected option is 'Secret Key'.
- 2. Account Key (\*): Provide azure blob account key.
- 3. Account Name (\*): Provide the name of the Azure Blob account.
- 4. Container (\*): Container details.
- 5. Read Directory: Select this option using the check box to read the blob name from the directory.
- 6. Blob Name: Provide a blob name.
- 7. File Type: Select a file type from the drop-down menu. (The supported file types are CSV, JSON, PARQUET, AVRO)

Azure Blob Reader			🙆 🔋 : 🕂 🕹
Basic Information Meta	Information		
Read Using* Secret Key	<ul> <li>Account Key*</li> </ul>	Account Name*	Container*
Read Directory	Blob Name	File Type* 👻	

Note: If the 'Read Directory' option is marked/enabled the 'Blob Name' field will not appear.

Azure Blob Reader		2			Ø	Î	<u>↓</u>	<u> </u>
Basic Information	Meta Inform	ation						
Read Using*								
Secret Key	•	Account Key*		Account Name*	 Container	*		
Read Direc	tory	File Type*	*					

The Meta Information tab when the selected Read Using option is 'Principal Secret':

- 1. Read Using: the selected option is 'Principal Secret'.
- 2. Client ID (\*): Provide client ID.
- 3. Tenant ID (\*): Provide Tenant ID.
- 4. Client Secret (\*): Provide client secret key
- 5. Account Name (\*): Provide the name of the Azure Blob account.
- 6. Container (\*): Container details.
- 7. Reader Directory: Enable or mark the box if wish to read the blob name from the directory.



- 8. Blob Name: Provide a blob name. (This field appears if the 'Read Directory' option is not marked/enabled.
- 9. File Type: Select a file type from the drop-down menu. (The supported file types are CSV, JSON, PARQUET, AVRO)

Azure Blob Reader	<b>Ø</b>				Ø		<u>↓</u>	<u> </u>
Basic Information	Meta Information							
Read Using* Principal Secret	▼ Client ID*		Tenan	t ID*				
Client Secret*					Account Na	ame*		
Container*	Read [	Directory	Blob Name		File Type*		•	

Note: If the 'Read Directory' option is marked/enabled the 'Blob Name' field will not appear.

ire Blob Reader	_0		9	: +	<u>+</u> <u>+</u>
asic Information Meta Inf	ormation				
Read Using* Principal Secret	Client ID*	Tenant ID*			
Client Secret*			Account Name*	r	_
Container*	Read Directory	File Type* 💌			

- v) Click the 'Save Component in Storage' icon after doing all the configurations to save the reader component.
- vi) A notification message appears to inform you that the component properties have been saved.



Note:

- a. The (\*) symbol indicates that the field is mandatory.
- b. Either table or query must be specified for the data readers except for SFTP Reader.
- c. Selected Columns- There should not be a data type mismatch in the Column Type for all the Reader components.
- d. The user can pass credentials as Environment Variable in some reader components instead of the actual credential values.

The supported readers are RDBMS, Mongo DB, Azure Reader, Azure Metadata Reader. E.g., The following image displays the Environment Variable used to configure the 'Meta Information' tab for the Mongodb Reader.



Mongodb Reader	0			2 🕯 : ÷ 🕹
Basic Information	Meta Information	Configuration		
Connection Type* Standard	-	Host* @ENV.MONGO_DEV_HOST	Port* 27017	User Name* @ENV.MONGO_DEV_USERNAME
Password*		Database Name* bi_testing	Collection Name* dm	Query Select age,salary from dm
Limit		Additional Parameters	_	

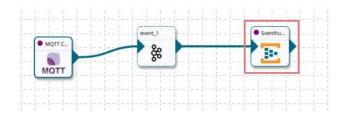
# 9.2. Writer

The Data Pipeline provides Writer components to write input data to the required place and create a pipeline workflow.

- i) Navigate to the Pipeline Workflow Editor.
- ii) Expand the Writer section using the Components Pallet.
- iii) The following Writer components are available under the System tab of the Component Pallet.



Writers are generally used with the combination of a Reader and an Event in the Pipeline Workflow. As the Eventhub Publisher component is shown in the below image:



# 9.2.1. S3 Writer

The S3 writer component writes data to the AWS S3 location.

- i) Drag & Drop the '**S3 Writer**' component to the Workflow Editor and connect to a pipeline workflow containing a reader and an Event component to get the data.
- ii) Click the dragged Writer component to get the Properties tabs below.
- iii) Basic Information: It is the default tab to open for the S3 writer while configuring the component.
  1) Select an Invocation type from the drop-down menu to confirm the running mode of the reader component. Select 'Real-Time' or 'Batch' from the drop-down menu.



- Deployment Type: It displays the deployment type for the component. This field comes preselected.
- 3) Container Image Version: It displays the image version for the docker container. This field comes pre-selected.
- 4) Failover Event: Select a failover Event from the drop-down menu.
- 5) Batch Size: Provide the maximum number of records to be processed in one execution cycle (Min limit for this field is 10).



- iv) Open the 'Meta Information' tab and fill in all the connection-specific details for the S3 Writer.
  - a. Bucket Name (\*): Provide Bucket name
  - b. Access Key (\*): Access key shared by AWS to login
  - c. Secret Key (\*): Secret key shared by AWS to login
  - d. Table (\*): Mention the Table or object name which is to be read
  - e. Zone (\*): Provide the zone information
  - f. File Type: Select a file type from the drop-down menu (CSV, JSON, PARQUET, AVRO are the supported file types)
  - g. Save Mode: supported save mode is 'Append' at present.

S3 Writer						+	<u>+</u>
Basic Information	Meta Information						
Bucket Name*	Access Key*		Secret Key*		Table*		
Zone*	File Type	•	Save Mode	<b>T</b>			

Selected Columns- Select the columns that you wish to store in the S3 location. Provide Column V) Name, Alias Name, Column Type for the selected column(s).

(Add multiple columns to the Selected Columns section using the 'Add New Column' option.)

Selected Columns				<b>( ( ) ( )</b>
Name	Alias Name	Column Type	• ×	
Add New Column				

Save the component properties. vi)



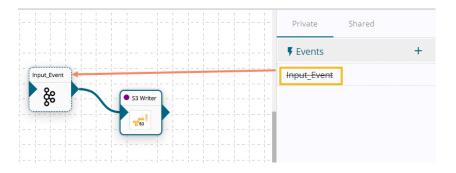


vii) A message appears to assure that the component properties have been saved.



#### Note:

- a. The S3 Writer requires data input from an Event and writes the data to the S3 location.
  - i) Create one event and drag it to the workspace.
  - ii) Connect the created input Event to the dragged S3 writer component.



b. The data in the input event can come from any Ingestion, Readers, or shared events.

## 9.2.2. RDBMS Writer

The RDBMS writer helps to write data to the supported relational databases.

- i) Drag & drop the RDBMS Writer component to the Workflow Editor and connect with a pipeline workflow containing a reader and an event component to get the data.
- ii) Click the RDBMS writer to open component properties tabs below.
- iii) Basic Information: It is the default tab to open for the RDBMS writer while configuring the component.
  - 1) Select an Invocation type from the drop-down menu to confirm the running mode of the reader component. Select either the 'Real-Time' or 'Batch' option from the drop-down menu.
  - 2) Deployment Type: It displays the deployment type for the component. This field comes preselected.
  - 3) Container Image Version: It displays the image version for the docker container. This field comes pre-selected.
  - 4) Failover Event: Select a failover Event from the drop-down menu.
  - 5) Batch Size: Provide the maximum number of records to be processed in one execution cycle (Min limit for this field is 10).



System Custom			0	100% C	
Reader 🗸	RDBMS Write			100%	
Writer	S3.Peader Sample Pip.				
🐝 🔔 😥	<b>86</b>				
	RDBMS Writer	0		<u>+</u>	Ť
	3 Basic Information Meta Information				
😒 💽 🥿	Deployment Type Container Image Version Invocation Type*				
Transformation	Failover Event < Batch Size(min 10)				

- iv) Open the 'Meta Information' tab and fill in all the connection-specific details for the RDBMS Writer.
  - a. Host IP Address (\*)
  - b. Port (\*)
  - c. Username (\*)
  - d. Password (\*)
  - e. Database Name (\*)
  - f. Table Name (\*)
  - g. Driver (\*) (MySQL/MSSQL/Oracle/Postgres)
  - h. Save Mode (Append/Overwrite/Upsert)

BMS Writer	0			9 🕯	:	+
Basic Information Meta Info	rmation					
Host IP Address*	Port*	Username*	Password	j*		
Host IP Address* @ENV.DATABASE_HOST	Port* @ENV.D	Username* @ENV.DATABASE_USERNA		j+		

v) Selected Columns: The users can select some specific columns from the table to read data instead of selecting a complete table; this can be achieved via the 'Selected Columns' section. Select the columns which you want to read and if you want to change the name of the column, then put that name in the alias name section otherwise keep the alias name the same as of column name and then select a Column Type from the drop-down menu. or

Use the 'Download Data' and 'Upload File' options to select the desired columns.

- 1. Upload File: The user can upload the existing system files (CSV, JSON) using the '**Upload File**' icon (file size must be less than 2 MB).
- 2. Download Data (Schema): The user can download the schema structure in JSON format by using the '**Download Data**' icon.

Selected Columns					
Name	Alias Name	Column Type	*	×	
Add New Column					

vi) Save the configured properties for the RDBMS writer.





vii) A message appears to notify that the component properties got saved.



Note: The data input event can take data from Ingestion, Readers, or shared Event.

## 9.2.3. HDFS Writer

The HDFS writer writes the data to the HDFS location.

- i) Drag the HDFS Writer component to the Workflow Editor and connect in a proper pipeline workflow to get the data.
- ii) Click the HDFS Writer to open the component properties tabs below.
- iii) Basic Information: It is the default tab to open for the HDFS Writer while configuring the component.
  - 1) Select an Invocation type from the drop-down menu to confirm the running mode of the reader component. Select either 'Real-Time' or 'Batch' from the drop-down menu.
  - 2) Deployment Type: It displays the deployment type for the component. This field comes preselected.
  - 3) Container Image Version: It displays the image version for the docker container. This field comes pre-selected.
  - 4) Failover Event: Select a failover Event from the drop-down menu.
  - 5) Batch Size: Provide the maximum number of records to be processed in one execution cycle (Min limit for this field is 10).



- iv) Open the '**Meta Information**' tab and fill in all the connection-specific details for the HDFS Writer.
  - 1. Host IP Address (\*): Hadoop IP address of the host
  - 2. Port (\*): Port number
  - 3. Table (\*): Table or object name which must be read
  - 4. Zone (\*): HDFS Zone location

HDFS Writer	_0		٢	<b>i</b> :	+	<u>+</u>
Basic Information Meta In	nformation					
Host IP Address*	Port* 8020					
Table*	Zone*					

5. Selected Columns: The users can select some specific columns from the table to read data instead of selecting a complete table; this can be achieved via the 'Selected Columns' section. Select the columns which you want to read and if you want to change the name of the column, then put that name in the alias name section otherwise keep the alias name the same as of column name and then select a Column Type from the drop-down menu. or

Use 'Download Data' and 'Upload File' options to select the desired columns.

- Upload File: The user can upload the existing system files (CSV, JSON) using the 'Upload File' icon (file size must be less than 2 MB).
- Download Data (Schema): Users can download the schema structure in JSON format by using the 'Download Data' icon.

Selected Columns					
Name	Alias Name	Column Type	*	×	
Add New Column					

6. Partition Column- Provide a unique Key column name to partition data in Spark.

Partition Columns		
Name	×	
Add New Column		

- 7. File Format (\*): Select a file type from the drop-down menu (CSV, JSON, PARQUET, AVRO are the supported file types)
- 8. Save Mode (\*): It displays the supported save mode. 'Append' is the supported save mode is at present.
- v) After doing all the configurations click the '**Save Component in Storage**' icon provided in the Writer configuration panel to save the component.
- vi) A message appears to notify about the component properties.



Note: The Input Event can take data from ingestion, Readers, or Shared Event.



## 9.2.4. Cassandra Writer

The Cassandra Writer writes data to the Cassandra Database.

- i) Drag & drop the Cassandra Writer component to the Workflow Editor and connect to a pipeline workflow to get the data.
- ii) Click the dragged Cassandra Writer component to open the component properties tabs below.
- iii) Basic Information: It is the default tab to open for the Cassandra Writer while configuring the component.
  - 1) Select an Invocation type from the drop-down menu to confirm the running mode of the reader component. Select 'Real-Time' or 'Batch' from the drop-down menu.
  - 2) Deployment Type: It displays the deployment type for the component. This field comes preselected.
  - 3) Container Image Version: It displays the image version for the docker container. This field comes pre-selected.
  - 4) Failover Event: Select a failover Event from the drop-down menu.
  - 5) Batch Size: Provide the maximum number of records to be processed in one execution cycle (Min limit for this field is 10).



- iv) Open the '**Meta Information**' tab and provide the connection-specific details.
  - a. Host IP address (\*): Provide hostname-based to connect the data connector.
  - b. Port number (\*): The server port number gets displayed by default.
  - c. Keyspace (\*): Provide a keyspace.
  - d. Table (\*): Provide a table name where you wish to save the data.
  - e. Cluster: Provide a number to signify the number of clusters.
  - f. Username (\*): Provide the Username of the selected connection.
  - g. Password (\*): Provide a password for the database.
  - h. Compression Method: Select a method from the drop-down menu (the supported method is SNAPPY)
  - i. Consistency: Provide a number to define consistency.
  - j. No. of Rows per Batch: Provide the number of rows to be fetched batch-wise.
  - k. UUID Column Name- Enter the UUID column name (Optional)



Cassandra Writer					٢	Î	:	+++++++++++++++++++++++++++++++++++++++	Ŧ
Basic Information	eta Information								
Host IP Address*	Port* 9042	Keyspace*		Table*					
Cluster	Username*	Passv	word*		Compr	essior	n Metho	d	*
Consistency	No. of Rows Per B	latch UUID	Column Name						

v) Selected Columns: The users can select some specific columns from the table to read data instead of selecting a complete table; this can be achieved via the 'Selected Columns' section. Select the columns which you want to read and if you want to change the name of the column, then put that name in the alias name section otherwise keep the alias name the same as of column name and then select a Column Type from the drop-down menu. or

Use 'Download Data' and 'Upload File' options to select the desired columns.

- 1. Upload File: The user can upload the existing system files (CSV, JSON) using the 'Upload File' icon (file size must be less than 2 MB).
- 2. Download Data (Schema): Users can download the schema structure in JSON format by using the '**Download Data**' icon.

Selected Columns					A
Name	Alias Name	Column Type	•	×	
Add New Column					

vi) **Partition Columns**: Provide the name of the partition column. Use the 'Add New Column' option to add a new partition column.

Partition Columns	
Name	×
Add New Column	

#### Note:

- i. Users can add multiple Selected Columns and Partition Columns by clicking the 'Add New Column' option.
- ii. UUID Column Name, Selected Columns, and Partition Columns are optional.
- vii) Save Mode- Select the mode to save the data from the drop-down menu (The supported save mode is the '**Append'**).

Append 🔻	Save Mode*	
	Append	•



- viii) Click the 'Save Component in Storage' 🥌 icon for the Cassandra Writer component.
- ix) A message appears to notify the same.



Note: The input event can read data from Ingestion, Readers, and Shared Event.

## 9.2.5. ES Writer

The ES writer writes the data to the Elastic Search engine.

- i) Drag & drop the ES Writer component to the Workflow Editor and connect to a pipeline workflow to get the data.
- ii) Click the dragged ES Writer component to open the component properties tabs below.
- iii) Basic Information: It is the default tab to open for the ES Writer while configuring the component.
  - 1) Select an Invocation type from the drop-down menu to confirm the running mode of the reader component. Select either '**Real-Time**' or '**Batch**' from the drop-down menu.
  - 2) Deployment Type: It displays the deployment type for the component. This field comes preselected.
  - 3) Container Image Version: It displays the image version for the docker container. This field comes pre-selected.
  - 4) Failover Event: Select a failover Event from the drop-down menu.
  - 5) Batch Size: Provide the maximum number of records to be processed in one execution cycle (Min limit for this field is 10).



- iv) Open the 'Meta Information' tab and fill in all the connection-specific details for the ES Writer.
  - a. Host IP Address
  - b. Port number
  - c. ES Index Id
  - d. ES Resource Type,
  - e. Save Mode: It displays the supported save modes. It supports only the 'Append' option.
  - f. Mapping Id- Provide mapping id (Optional)



ES Writer	Ø			2 🖬 : 🕂	<u>+</u>
Basic Information	Meta In	formation			
Host IP Address*		Port* 9200	ES Index Id*	ES Resource Type*	
Save Mode	•	Mapping Id			

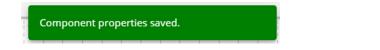
v) Selected Columns: The users can select some specific columns from the table to read data instead of selecting a complete table; this can be achieved via the 'Selected Columns' section. Select the columns which you want to read and if you want to change the name of the column, then put that name in the alias name section otherwise keep the alias name the same as of column name and then select a Column Type from the drop-down menu. or

Use 'Download Data' and 'Upload File' options to select the desired columns.

- 1. Upload File: The user can upload the existing system files (CSV, JSON) using the 'Upload File' icon (file size must be less than 2 MB).
- 2. Download Data (Schema): Users can download the schema structure in JSON format by using the 'Download Data' icon.

Selected Columns					<b>D A</b> ×
Name	Alias Name	Column Type	~	×	
Add New Column					

- vi) Click the 'Save Component in Storage' 🥌 icon for the ES Writer component.
- vii) A message appears to notify the same.



Note: The input event can read data from Ingestion, Readers, and Shared Event.

# 9.2.6. MongoDB Writer

The MongoDB writer writes the data to the database.

- i) Drag & drop the MongoDB Writer component to the Workflow Editor and connect it to a pipeline workflow to get the data.
- ii) Click the dragged MongoDB Writer component to open the component properties tabs below.
- iii) Basic Information: It is the default tab to open for the MongoDB Writer while configuring the component.
  - 1) Select an Invocation type from the drop-down menu to confirm the running mode of the reader component. Select 'Real-Time' or 'Batch' from the drop-down menu.
  - 2) Deployment Type: It displays the deployment type for the component. This field comes preselected.
  - 3) Container Image Version: It displays the image version for the docker container. This field comes pre-selected.
  - 4) Failover Event: Select a failover Event from the drop-down menu.



5) Batch Size: Provide the maximum number of records to be processed in one execution cycle (Min limit for this field is 10).

Reader		sample_pip		<u>ି</u> ( 100%   ସ୍
Writer	• 53 Reader	k Mongadh		
	3 Basic Information Meta	● Information		❷ <b>î</b> : <u>↓</u> Ţ
r E	Invocation Type*	■ Deployment Type ■ spark	Container Image Version 6.5.0-R1	
Transformation ML	Failover Event	▼ Batch Size(min 10)	_	

- iv) Open the 'Meta Information' tab and configure all the connection-specific details for the MongoDB Writer.
  - 1. Connection Type: Select either of the options out of 'Standard' and 'SRV' connection types.
  - 2. Port number (\*): Provide the Port number (It appears only with the 'Standard' connection type).
  - 3. Host IP Address (\*): IP address of the host.
  - 4. Username (\*): Provide username.
  - 5. Password (\*): Provide a valid password to access the Mongo DB.
  - 6. Database Name (\*): Provide the name of the database where you wish to write data.
  - 7. Collection Name (\*): Provide the name of the collection.
  - 8. Save Mode: Select an option from the drop-down menu (the supported options are Upsert, Ignore and Append).
  - 9. Composite Keys (\*): This field appears only when the selected save mode is '**Upsert**'. The user can enter multiple composite keys separated by commas.
  - 10. Additional Parameters: Provide details of the additional parameters.

ongodb Writer			
Basic Information Meta Ir	oformation		
Connection Type* 💌	Host*	User Name*	Password*

The Meta Information fields vary based on the selected 'Connection Type' option.

i. Meta Information Tab with Standard as the Connection Type



ngodb Writer	0		9	÷ +	
asic Information Meta	Information				
Connection Type* Standard	Port* 27017	Host* @ENV.MONGO_DEV_H	User Name* @ENV.MON0	GO_DEV_U	
Password*	Database Name* bi_testing	Collection Name* envrkv	Save Mode Upsert	•	1
Composite Keys [*keys seperated		Parameters			1

ii. Meta Information Tab with SRV as the Connection Type

godb Writer			9 📋 :	<u>+</u> ↑ .
sic Information Met	a Information			
Connection Type* SRV	Host* @ENV.MONGO_DEV_H	User Name* @ENV.MONGO_DEV_U	Password*	
Database Name* bi_testing	Collection Name* envrkv	Save Mode Upsert 🗸		
Composite Keys [*keys seperate Cust id	ed by com Additional Parar	neters		

v) Selected Columns: The users can select some specific columns from the table to read data instead of selecting a complete table; this can be achieved via the 'Selected Columns' section. Select the columns which you want to read and if you want to change the name of the column, then put that name in the alias name section otherwise keep the alias name the same as of column name and then select a Column Type from the drop-down menu. or

Use 'Download Data' and 'Upload File' options to select the desired columns.

- 1. Upload File: The user can upload the existing system files (CSV, JSON) using the '**Upload File**' icon (file size must be less than 2 MB).
- 2. Download Data (Schema): Users can download the schema structure in JSON format by using the '**Download Data**' icon.

Selected Columns				<b>( ( ) (</b>
Name	Alias Name	Column Type	<u> </u>	
Add New Column				

- vi) Click the 'Save Component in Storage' Sicon for the MongoDB Writer component.
- vii) A message appears to notify the same.



Note: The input event can read data from Ingestion, Readers, and Shared Event.

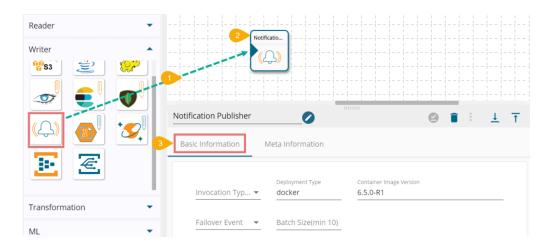


# 9.2.7. Notification Publisher

Notification Publisher Component as the name suggests is meant to publish the messages coming to Kafka as a notification. It is very helpful in acting scenarios where an immediate reaction is required in some conditions.

Example: A pipeline is monitoring an IoT sensor that sends the temperature of a machine. If the temperature reaches a certain threshold; the process engineer is to be notified immediately to maintain the same to stabilize the machine. Here, the Notification Publisher component is useful.

- i) Drag and drop the Notification Publisher component to the Workflow Editor. (The user needs to connect the Notification component to a pipeline workflow to get the data).
- ii) Click the 'Notification Publisher' component to open the component properties below.
- iii) Basic Information: It is the default tab to open for the Notification Publisher while configuring the component.
  - 1) Select an Invocation type from the drop-down menu to confirm the running mode of the reader component. Select 'Real-Time' or 'Batch' from the drop-down menu.
  - 2) Deployment Type: It displays the deployment type for the component. This field comes pre-selected.
  - 3) Container Image Version: It displays the image version for the docker container. This field comes pre-selected.
  - 4) Failover Event: Select a failover Event from the drop-down menu.
  - 5) Batch Size: Provide the maximum number of records to be processed in one execution cycle (Min limit for this field is 10).



- iv) Open the '**Meta Information**' tab and fill in all the connection-specific details for the Notification Writer.
  - a. Type: Select a notification type using the drop-down menu. The provided options are:
    - i. SNS
    - ii. Google Pub/Sub
  - b. Based on the selected Notification Type the remaining configuration fields get displayed.
  - c. The user needs to complete the configuration process for the selected notification type and click the 'Save' <sup>Selected</sup> icon to complete the configuration process for the Notification Publisher.

Notification Publisher	_0	۲	<u>+</u> ↑	<u>+</u>	
Basic Information	Meta Information				
Notification Type*	]				

# 9.2.7.1. SNS as Notification Type

- i) Navigate to the Meta Information tab provided for the Notification Publisher.
- ii) Select the 'SNS' as the Notification Type option using the drop-down menu.
- iii) The following configuration fields appear:
  - a. Access Key: Access keys consist of two parts: an access key ID (for example, AKIAIOSFODNN7EXAMPLE) can be retrieved from AWS key management.
  - b. Secret Key: A secret access key (for example, wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY) can also be retrieved from the AWS key management.
  - c. Topic Arn: The field gets filled after the topic creation. Navigate to Topics and then click the desired topic.
  - d. Region: Select an option from the drop-down menu where the topic has been created.

Notification Publish	er 💋		2 i ÷ ÷	<u>+</u>
Basic Information	Meta Information			
Notification Type*	<ul> <li>Access Key*</li> </ul>	Secret Key*	Topic Arn	
Region	<b>•</b>			

- iv) The user needs to follow some more steps to create and subscribe to a topic. The below-given description explains those steps:
  - a. Navigate to the URL: https://aws.amazon.com/console/
  - b. Click the 'Sign In to the Console' option.

C A https://aws.amazon.com/con	sole/	\$6 ☆ @ ■ 5] ø A <mark>6</mark> ⊡ 15 @
VS Jucts Solutions Pricing Docume	Contact Sale entation Learn Partner Network AWS Marketplace Explore More Q	s Support + English + My Account + Sign In to the Conso
PRODUCTS & SERVICES	AWS Management Console	
AWS Console	Access and manage Amazon Web Services through a significant services and manage Amazon Web Services through a significant services and manage Amazon Web Services through a significant services and manage Amazon Web Services through a significant services and manage Amazon Web Services through a significant services and manage Amazon Web Services through a significant service services and manage Amazon Web Services through a significant service service services and manage Amazon Web Services through a significant service service service service services and manage Amazon Web Services through a significant service servi	Get Started with AWS for Free
RELATED LINKS	intuitive web-based user interface. You can also use the A mobile app to quickly view resources on the go	WS Console Create a Free Account
Articles & Tutorials	mobile app to quickly view resources on the go	Or Sign In to the Console
Developer Tools		Receive twelve months of access to the AWS
Public Data Sets		Free Usage Tier and enjoy AWS Basic Support features including, 24x7x365
Amazon Machine Images (AMIs)		customer service, support forums, and
Videos & Webinars What's New		more.

- c. Sign in with your credentials.
- d. The AWS Management Console opens.
- e. Click the 'Services' option from the header drop-down.



f. Search SNS service using the 'Find Services' space.

	🖈 🙁 📕 🎝 😕 🚰 😂 😩 :				
aws Services - Resource Groups - *	A pipelina di 6076-9948-8674F      Obioux Support      ✓				
AW <mark>S</mark> Management Console					
AWS services	Access resources on the go				
Find Services You can over manes, keywards or acremyms. Q. Example: Relational Database Service, database, RDS	Access the Management Console using the AWS Console Mobile App. Learn more				
▼ Recently visited services	Explore AWS				
Simple Notification UKey Management Service Service	Data Lake Storage				
🔁 53 🚟 Simple Queue Service	Build your data lake on the most secure, durable, and scalable storage. Learn more 🗹				
► All services	Run Serverless Containers with AWS Fargate				

- g. The Amazon SNS page opens.
- h. Click the '**Create topic**' option.

C      Antipolitaneouspace	สน่หสหมอยสองสนองสนังหนุ่งผู้ผู้สืบสหมดิจะอุ่งหมดตอสมาร์สีที่/topics	*) 😌 📕 🌒 🥹 者 🔍 🗟 🔘 😩
aws Services - F	Resource Groups 🗸 🔭	A histore Contention of a state of the support +
Amazon SNS $\times$	Amazon SNS > Topics	
Dashboard <mark>Topics</mark> Subscriptions	Topics (2) Q. Search	Edit Delete Publish message Create topic
	Name	ARN V
	O TestTopic	XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
	O shobhit_demo_topic	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

- i. The Create topic page opens.
- j. Provide essential information for the topic.

etails	
me	This will appear on the emails or other protocols as Subject
Document_topic	
ximum 256 characters. Can include alphanum splay name - optional use this topic with SMS subscriptions, entered	Sisplay name. Only the first 10 characters are displayed in an SMS message. Info
splay name - optional	

k. Click the 'Create Subscription' option.



aws Services v F	Resource Groups 🗸 🕇			û pipeline @ 6876-9948-4674 × Ohio × Support
mazon SNS ×	Amazon SNS > Topics > Doci	ument_topic		
ashboard o <mark>pics</mark>	Document_topic			Edit Delete Publish message
ibscriptions	Details			
	Name Document_topic ARN बराउठाठावराउठावराउठावराउठावराउठावराउठावराउठावराउठावराउठावराउठावराउठावराउठावराउठावराउठावराउठावराउठावराउठावराउ	Gällikeonektosk	Display name sns_notification Pipeline Topic owner \$XXX53XEV5KNK	
	Subscriptions Access pol	icy Delivery retry policy (HTTP/5) Delivery status loggi	ng Encryption Tags	
	Subscriptions (0)		Edit Delete Request confirmation	Confirm subscription
	iD	▼ Endpoint	▼ Status	▼ Protocol ▲
		You don't have a	bscriptions found ny subscriptions to this topic. ate subscription	

- I. Search or provide the topic ARN.
- m. Select a protocol from the available list.
- n. Click the 'Create Subscription' option.

aws Services - F	Resource Groups 🗸 🔸	۵	pipeline @ 6876-9948-4674 🔻	Ohio 👻	Su
Amazon SNS $\times$	Amazon SNS > Subscriptions > Create subscription				
Dashboard Topics	Create subscription				
Subscriptions	Details				
	Topic ARN Q. arr:aws:sns:us-east-2:687699484674:Docu X Protocol				
	The type of endpoint to subscribe           Select protocol				
	НТТР				
	HTTPS firm it. Info				
	Email				
	Email-JSON				
	Amazon SQS				
	AWS Lambda				
		Can	cel Create subscription		

NOTE: The user may get a '**Pending Confirmation**' notification of subscriptions as shown in the below image. In such cases navigate to the email id and confirm the subscription request.

aws Services - Re	esource Groups 🗸 🔭	¢	) pipeline @ 6876-9948-4674 🔻	Ohio 👻 Support 👻
Amazon SNS ×	Subscription: 1d346dc3-b115-4641-a12e-	a493fab8403e	Edit	Delete
Dashboard Topics	Details			
Subscriptions	ARN am:aws:sns:us-east-2:687699484674:Document_topic:1d346dc3-b115-4641- a12e-a493fab8403e Endpoint prem.prakash@bdb.ai Topic Document_topic	Status Pending confirmation Protocol EMAIL		



#### o. Copy the ARN of the Topic.

mytest0107	Edit Delete Publish message
Details	
Name mytest0107	Display name -
ARN am:aws:sns:us-east-1:687699484674:mytest0107	Topic owner

# 9.2.7.2. Google Pub/Sub as the selected Notification Type

- i) Navigate to the Meta Information tab provided for the Notification Publisher.
- ii) Select the '**Google Pub/Sub**' as the Notification Type option using the drop-down menu.

Notification Publisher			$\ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ $	Î	:	+ +	<u>+</u>
Basic Information	Meta Information						
Notification Type*	Pub/Sub Topic ID*	Choose File	A	uther	nticati	on File	

- iii) The user needs to follow some more steps to get the Pub/Subtopic ID
- iv) Navigate to the <a href="https://console.cloud.google.com/">https://console.cloud.google.com/</a>
- v) Log in with your credentials.
- vi) The Landing page opens.
- vii) Click on the marked area.
- viii) Create a Project or use the desired one.

Google Cloud Platform Pro	ject1 👻		- 🛛 🔊 🔍 🚺 E			
Project info	1	RPI APIS	ı	Google Cloud Platform status		
Project name project1		Requests (requests/sec)	1.0	All services normal		
Project ID Scripplicklesi			0.8	ightarrow Go to Cloud status dashboard		
Project number		No data is available for the selected time fra	0.6			

ix) Search for the Pub/Sub-option in the Search bar.

≡ Google Cloud Platform S• NewProject01 ▼	Q. Pub/sub	5 9 9 💿 i 🚱
DASHBOARD ACTIVITY	Pub/Sub	CUSTOMIZE

- x) Create a Topic using the following steps:
  - a. Click the '**CREATE TOPIC**' option.
  - b. Provide a name for the new topic
  - c. Select the 'Google-managed key' as the Encryption option.
    - d. Click the '**CREATE TOPIC**' option.



**** ***	Topics 1 CREATE TOPIC	👕 DELETE	
Þ	➡ Filter table		Image: Organization     Image: Organization       Image: Organization     Image: Organization
≡	□ Topic name ↑ No rows to display	Encryption	Create a topic
0	No rows to display	_	
			A topic forwards messages from publishers to subscribers.
			rewtopic     Topic ID: projects/newproject01-246106/topics/newtopic
		3	Encryption  Google-managed key
			No configuration required O Customer-managed key
			Manage via Google Cloud Key Management Service
			CANCEL <sup>4</sup> CREATE TOPIC

- xi) The user gets either of the following screens:
  - a. While using an existing Topic ID

Topic ID		
projects/newproject0	1-246106/topics/newtopic	
Encryption key		
Google-managed key		
Labels		

Alternatively, while using the newly created Topic ID

Top	bics	+ CR	EATE TOPIC	ELETE				
Ŧ	Filter table					0	С	
	Topic name	↑	Encryption	Topic ID		Labels		
	newtopic		Google-managed	PARAKARANANANANANANANANANANANANA	6			÷
						-		
							-	

b. Create Auth-JSON for the current Project.



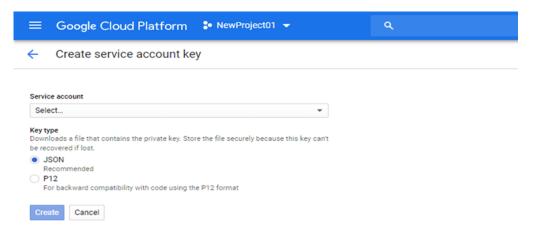
	· · · · · · · · · · · · · · · · · · ·				
←	→ C A https://console.clo	ud.google.com/cloudpubsub/t	copic/detail/newtopic?project=newpro	oject01-246106	\$
=	Google Cloud Platform	S• NewProject01 →			
♠	Home	+ PUBLISH MESSAGE	📩 IMPORT 👻 🏦 EXPORT 👻	<b>DELETE</b>	
È	Marketplace	î.			newtopic
<b>Description</b>	Billing				PERMISSIONS
8P3	APIs & Services	> Dashboard			Edit or delete pe Member" to gran
Ť	Support	> Library			Show in
Θ	IAM & admin	> Credentials			Ţ Filter tree
۲	Getting started				Role / Member 1
٢	Security	> <sub>&gt; TS</sub>			<ul> <li>Owner (1)</li> </ul>
COM	IPUTE				

# c. Create a Service account key by following the given steps:i. Click the 'Service account key' option from the Credentials tab.

=	Google Cloud Platform 🔹 NewProject01 👻	٩
API	Credentials	
¢	Credentials OAuth consent screen Domain verification	
册	Create credentials - Delete	
6	API key Identifies your project using a simple API key to check quota and access OAuth client ID Requests user consent so your app can access the user's data	authentication documentation .
	Service account key Enables server-to-server, app-level authentication using robot accounts	ite ✓ Service account
	Help me choose Asks a few questions to help you decide which type of credential to use	

- ii. Select a Service account using the drop-down menu.
- iii. Download the JSON file using the radio button.
- iv. Click the 'Create' option.

Use the copied Topic ID or the service account as metadata input.





- v) Click the 'Save Component in Storage' icon to save the component Properties for the Notification Publisher.
- vi) A message appears to notify the same.



Note: The input event can read data from Ingestion, Readers, and Shared Event.

## 9.2.8. Azure Writer

The Azure Writer writes the data to the Azure blob storage.

- i) Drag and drop the Azure Writer component to the Workflow Editor (Connect the Azure Writer component to a pipeline workflow to get the data).
- ii) Click on the dragged Azure Writer to get the component properties tabs.
- iii) **Basic Information**: It is the default tab to open for the component while configuring the component.
  - 1) Invocation Type: Select an invocation mode out of '**Real-Time**' or '**Batch**' using the drop-down menu.
  - 2) Deployment Type: It displays the deployment type for the reader component. This field comes pre-selected.
  - 3) Container Image Version: It displays the image version for the docker container. This field comes pre-selected.
  - 4) Failover Event: Select a failover Event from the drop-down menu.
  - 5) Batch Size: Provide the maximum number of records to be processed in one execution cycle (Min limit for this field is 10).



- iv) Open the '**Meta Information**' tab and fill in all the connection-specific details for the Azure Writer.
  - 1. Write Using (\*): Select an option from the drop-down menu to connect to data. The supported options are '**Connection String**' and '**Secret Key**'.
  - 2. Account Name (\*): Name of the Azure account.
  - 3. Account Key (\*): Provide Account Key to login.
  - 4. Container (\*): Container details.



- 5. Blob Name (\*): Provide a blob name. The Blob field appears if the '**Read Directory**' option is not enabled.
- 6. File Format (\*): Select a file format option from the drop-down menu (CSV, JSON, PARQUET, AVRO are the supported file types).
- 7. Save Mode: Choose a mode to save data out of Append or Overwrite.

ure Writer	Ø					9	:	+	<u>+</u>
Basic Information	Meta Inform	ation							
Write Using*	•	Account Name*		Account Key*		Container*			
Blob Name*		File Format *	-	Save Mode	*				

v) Selected Columns: There is also a section for the selected columns in the Meta Information tab if the user can select some specific columns from the table to read data instead of selecting a complete table so this can be achieved by using the 'Selected Columns' section. Select the columns which you want to read and if you want to change the name of the column, then put that name in the alias name section otherwise keep the alias name the same as of column name and then select a Column Type from the drop-down menu. Or

Use 'Download Data' and 'Upload File' options to select the desired columns.

- 1. Upload File: The user can upload the existing system files (CSV, JSON) using the 'Upload File' icon (file size must be less than 2 MB).
- 2. Download Data (Schema): Users can download the schema structure in JSON format by using the 'Download Data' icon.

Selected Columns				🗇 🛆 🗙
Name	Alias Name	Column Type	• ×	
Add New Column				

- vi) Click the 'Save Component in Storage' icon after doing all the configurations to save the writer component.
- vii) A notification message appears to inform you that the component properties have been saved.



# 9.2.9. Azure Cosmos DB Writer

The Azure Cosmos DB Writer writes the data to the Cosmos DB.

- i) Drag and drop the Azure Cosmos DB Writer component to the Workflow Editor (connect the Azure Cosmos DB writer component to a pipeline workflow to get the data).
- ii) Click on the dragged Azure Cosmos DB Writer to get the component properties tabs.
- iii) **Basic Information**: It is the default tab to open for the component while configuring the component.



- 1) Invocation Type: Select an invocation mode out of '**Real-Time**' or '**Batch**' using the drop-down menu.
- 2) Deployment Type: It displays the deployment type for the reader component. This field comes pre-selected.
- 3) Container Image Version: It displays the image version for the docker container. This field comes pre-selected.
- 4) Failover Event: Select a failover Event from the drop-down menu.
- 5) Batch Size: Provide the maximum number of records to be processed in one execution cycle (Min limit for this field is 10).

Reader 🗸	Azure Cos	
Writer		
<b>*\$3</b> 🔮 🜮		
	Azure CosmosDB Writer	2 i : <u>1</u> T
	Basic Information Meta Information	
	Deployment Type	Container Image Version
Transformation	Invocation Typ  spark	6.5.0-R80
ML 👻	Failover Event   Batch Size(min 10)	)

- iv) Open the '**Meta Information**' tab and fill in all the connection-specific details for the Azure Cosmos DB Writer.
  - 1. Endpoint (\*):
  - 2. Master Key (\*): Provide the master key.
  - 3. Database Name (\*): Provide Account Key to login.
  - 4. Container Name (\*): Container details.
  - 5. Partition Key: Provide the partition key.
  - 6. Save Mode: Select a Save mode using the drop-down menu.

Azure CosmosDB Writ	ter 🖉		🙆 📋 :	<u>+</u> <u>↓</u>
Basic Information	Meta Information			
Endpoint*		MasterKey*		
Database Name*	Container Name*	Partition Key	Save Mode	*

v) Selected Columns: There is also a section for the selected columns in the Meta Information tab if the user can select some specific columns from the table to read data instead of selecting a complete table so this can be achieved by using the 'Selected Columns' section. Select the Source Field using the drop-down menu. Provide the name of the Destination Field. Select a Column Type from the drop-down menu.

Or

Use the 'Download Data' and 'Upload File' options to select the desired columns.

- Upload File: The user can upload the existing system files (CSV, JSON) using the 'Upload File' icon (file size must be less than 2 MB).
- 2. Download Data (Schema): Users can download the schema structure in JSON format by using the 'Download Data' icon.



Selected Columns					<b>C A</b> ×
Name	Alias Name	Column Type	*	×	
Add New Column					

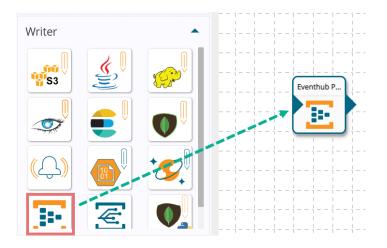
- vi) Click the 'Save Component in Storage' icon after doing all the configurations to save the writer component.
- vii) A notification message appears to inform you that the component properties have been saved.



## 9.2.10. Eventhub Publisher

This Component consumes the data from the previous event and publishes the data on a given EventHub name. This data can be consumed by other applications. EventHub Subscriber in the data pipeline comes under the Writer component group.

i) Drag and drop the Eventhub Publisher component to the Workflow Editor (connect the Eventhub Publisher to a pipeline workflow to get the data).



- ii) Click on the dragged Eventhub Publisher component to get the component properties tabs.
- iii) Configure the Basic Information tab.
  - 1) Invocation Type: Select an invocation mode out of '**Real-Time**' or '**Batch**' using the dropdown menu.
  - 2) Deployment Type: It displays the deployment type for the reader component. This field comes pre-selected.
  - 3) Container Image Version: It displays the image version for the docker container. This field comes pre-selected.
  - 4) Failover Event: Select a failover Event from the drop-down menu.
  - 5) Batch Size: Provide the maximum number of records to be processed in one execution cycle (Min limit for this field is 10).

Eventhub Publisher			g	Î	0 0 0	<u> </u>	1
Basic Information	Meta Information						
Invocation Typ 🔻	Deployment Type docker	Container Image Version 6.5.0-R80					
Failover Event 🔻	Batch Size(min 10)						

- iv) Configure the Meta Information tab by providing the required fields.
  - a. **Connection String**: There is a connection string to connect to the event hub service (this can be found in Azure platform access manages].
  - b. **Consumer Group**: A consumer group is a view (state, position, or offset) of an entire event hub.
  - c. Eventhub name: Name of the subscribed event hub.
  - d. **Checkpoint location:** it is an HDFS location URL ("hdfs://<ip>/user/filename/Test"). It can be used when a component fails, and you want to read from the exact event where you left.

Basic Information Meta	aInformation			
Connection String*		Consumer Group		
EventHub Name*	Checkpoint Location			

- v) Click the 'Save Component in Storage' icon after doing all the configurations to save the writer component.
- vi) A notification message appears to inform you that the component properties have been saved.

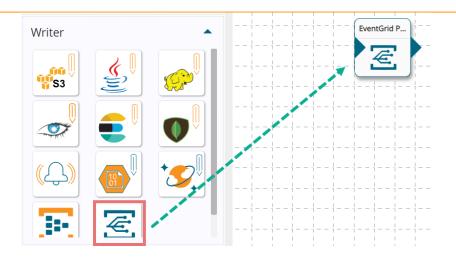


## 9.2.11. EventGrid Producer

The Azure Event Grid provides the capability for building event-based applications.

i) Drag and drop the EventGrid Producer component to the Workflow Editor (connect the EventGrid Producer to a pipeline workflow to get the data).





- ii) Click on the dragged EventGrid Producer component to get the component properties tabs.
- iii) Configure the **Basic Information tab**.
  - 1) Invocation Type: Select an invocation mode out of '**Real-Time**' or '**Batch**' using the dropdown menu.
  - 2) Deployment Type: It displays the deployment type for the reader component. This field comes pre-selected.
  - 3) Container Image Version: It displays the image version for the docker container. This field comes pre-selected.
  - 4) Failover Event: Select a failover Event from the drop-down menu.
  - 5) Batch Size: Provide the maximum number of records to be processed in one execution cycle (Min limit for this field is 10).

EventGrid Producer			$\boldsymbol{\boldsymbol{\heartsuit}}$	Î	:	<u> </u>	Ť
Basic Information	Meta Information						
Invocation T 🝷	Deployment Type docker	Container Image Versi 6.5.0-R1	on				
Failover Event 💌	Batch Size(min						

- iv) Configure the Meta Information tab by providing the required fields.
  - a. Topic Endpoint
  - b. Topic Secret Key

EventGrid Producer		۷	Î	0	Ţ	Ť
Basic Information	Meta Information					
Topic Endpoint*	Topic Secret Key*					



- v) Click the 'Save Component in Storage' icon after doing all the configurations to save the writer component.
- vi) A notification message appears to inform you that the component properties have been saved.

Component properties saved.

## 9.3. Transformation

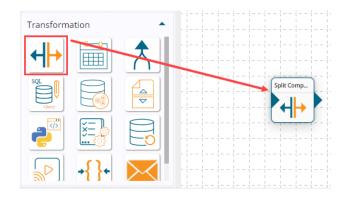
Transformation components allow users to transform the data. BDB Data Pipeline provides the following Transformation components to transform a variety of data.



# 9.3.1. Split Component

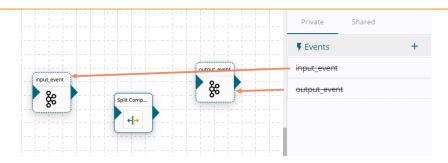
The Split component helps users to split the selected column(s) from the input data set based on the given regular expression.

i) Drag & drop the Split Component to the Workflow Editor.

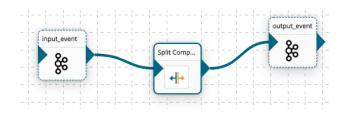


- ii) The transformation component requires an input event (to get the data) and sends the data to an output event.
- iii) Create two Events and drag them to the Workspace.





iv) Connect the input event and the output event (The data in the input event can come from any Ingestion, Readers, or shared events).



- v) Click the dragged Split component to get the component properties tabs below.
- vi) **Basic Information**: It is the default tab to open for the Split Component while configuring the component.
  - a. Select an Invocation type from the drop-down menu to confirm the running mode of the reader component. Select '**Real-Time**' or '**Batch**' from the drop-down menu.
  - b. Deployment Type: It displays the deployment type for the component. This field comes preselected.
  - c. Container Image Version: It displays the image version for the docker container. This field comes pre-selected.
  - d. Failover Event: Select a failover Event from the drop-down menu.
  - e. Batch Size (min 10): Provide the maximum number of records to be processed in one execution cycle (Min limit for this field is 10).

tiput, Event	Spit Comp.		
Split Component	0		◎ 🖬 : ⊥ 〒
Basic Information	Neta Information		
Invocation Type*	Deployment Type     docker	Container Image Version 6.5.0-R1	
Failover Event	▼ Batch Size(min 10)		

- vii) Open the 'Meta Information' tab and fill in all the connection-specific details for the Notification Writer.
  - 1) Fill the Regular Expression by which the data gets split (E.g., Comma (,) is used as the Regular Expression in the below-given image).
  - 2) Column Name- Provide the input column name.
  - 3) New Columns- Provide Output columns name separated by a comma. These output columns get created after transformation.
  - 4) Provide Default Value in the given field.



- 5) To perform multiple Split transformation actions, click the '**Add New Row**' option, and follow the above steps.
- 6) Enable the 'Keep Original Column' option to keep the original column.

	1010101	٢	•	:	+	
Information						
Column Name* full_name	New Columns [*column name seperated by comma (,)]* full_name, last_name			alue		
	Column Name*	Information Column Name* New Columns [*column name seperated by comma (,.]*	Column Name* New Columns [*column name seperated by comma (,)]*	Column Name* New Columns (*column name separated by comma (.))* Default V	Column Name* New Columns (*column name seperated by comma (.)]* Default Value	Column Name* New Columns (*column name seperated by comma (.))* Default Value

- viii) Click the 'Save Component in Storage' icon to save the configured properties for the Split component.
- ix) A message appears to notify the successful update of the component.

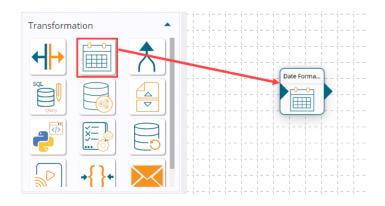


x) The Split component is ready to read the data coming from the input event, transform the data, and give the output data with the newly created column(s).

## 9.3.2. Date Formatter Component

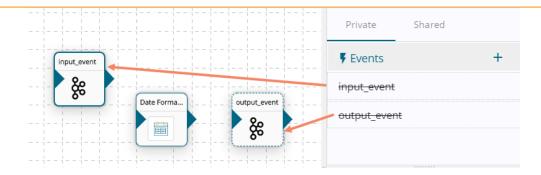
Users can alter the Date format by using this transform component.

i) Drag & drop the Replace Text component to the Workflow Editor.



- ii) The transformation component requires an input event (to get the data) and sends the data to an output event.
- iii) Create two Events and drag them to the Workspace.





iv) Connect the input event and the output event (The data in the input event can come from any Ingestion, Reader, or shared events).



- v) **Basic Information**: It is the default tab to open for the Date Formatter while configuring the component.
  - a. Select an Invocation type from the drop-down menu to confirm the running mode of the reader component. Select '**Real-Time**' or '**Batch**' from the drop-down menu.
  - b. Deployment Type: It displays the deployment type for the component. This field comes preselected.
  - c. Container Image Version: It displays the image version for the docker container. This field comes pre-selected.
  - d. Failover Event: Select a failover Event from the drop-down menu.
  - e. Batch Size (min 10): Provide the maximum number of records to be processed in one execution cycle (Min limit for this field is 10).

	te Formatter		te Formatter	Input_Event	Date Form	LEve		-
			e Formatter			11-1-1-1-1-1-1-1-1-1-1-1-1-1-1-1-1-1		
e Formatter		asic Information Meta Information	Deployment Type Container Image Version	e Formatter	0		0 i i <u>i</u>	1
		nivotauon ryper + spark 0.5.0-K1		Invocation Type*				

- vi) Open the Meta Information tab and provide the connection-specific details.
  - a. Fill in the Column Name.
  - b. Select the input format.
  - c. Select the output format.
  - d. Provide a name for the Output Column.



(This component transforms date type for selected input column to provided output data type.)

e. To perform multiple Date Formatter transformations, click the 'Add New Row' option, and follow the above steps.

Date Formatter	Ø			٢	Î	:	+	Ţ
Basic Information	Meta Inf	ormation						
Columns								
Column Name	*	Input Format*	 Output Format*					
Output Column	n Name*	×						
Add New Ro	w							

vii) **Selected Columns:** The users can select some specific columns from the table to read data instead of selecting a complete table; this can be achieved via the 'Selected Columns' section. Select the columns which you want to read and if you want to change the name of the column, then put that name in the alias name section otherwise keep the alias name the same as of column name and then select a Column Type from the drop-down menu.

or

Use 'Download Data' and 'Upload File' options to select the desired columns.

- 1. Upload File: The user can upload the existing system files (CSV, JSON) using the 'Upload File' icon (file size must be less than 2 MB).
- 2. Download Data (Schema): Users can download the schema structure in JSON format by using the 'Download Data' 🖸 icon.

Selected Columns				
Name	Alias Name	Column Type	<u> </u>	
Add New Column				

viii) Limit: Provide a limit of the records.

Limit		

- ix) Click the 'Save Component in Storage' icon to save the configured properties of the Date Formatter component.
- x) A message appears to notify the successful update of the component.



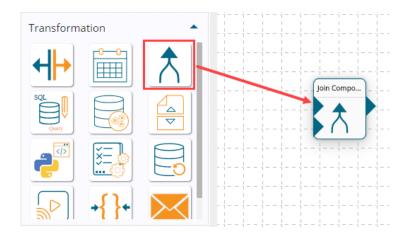


xi) The Date Formatter component is ready to read the data coming from the input event, transform the data, and send the newly formatted Date column(s) to the output event.

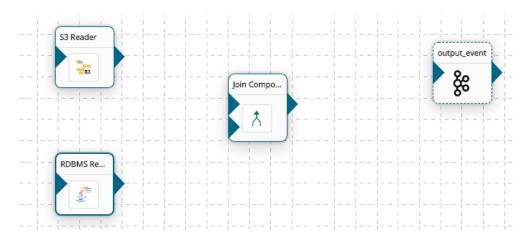
# 9.3.3. Join Component

The Join transform allows users to join two or more input data sets as per the user-defined join conditions.

i) Drag and Drop the Join component to the Workflow Editor.

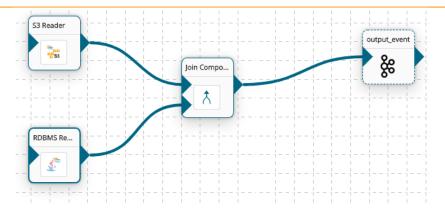


- ii) This transformation component requires two input data sources and sends the data to an output event.
- iii) Create two input data sources and one output Event and drag them to the workspace.



iv) Connect the input data sources and the output event. The data in input data sources can select only from Readers.





v) Configure the Input Data sources.

Note: Refer to the Reader section of the document for more information.

S3 Reader	Join Compo.	Output.Eve.			100%	٩
S3 Reader			۵		<u>+</u>	Ť
Invocation Type*	Deployment Type spark	Container Image Version 6.5.0-R1				
Failover Event 👻	Batch Size(min 10)					

- vi) Click the Join component to get the component properties field for the Join Component.
- vii) The Basic Information tab opens by default.
  - a. Select an Invocation type from the drop-down menu to confirm the running mode of the reader component. Select '**Real-Time**' or '**Batch**' from the drop-down menu.
  - b. Deployment Type: It displays the deployment type for the component. This field comes preselected.
  - c. Container Image Version: It displays the image version for the docker container. This field comes pre-selected.
  - d. Failover Event: Select a failover Event from the drop-down menu.
  - e. Batch Size (min 10): Provide the maximum number of records to be processed in one execution cycle (Min limit for this field is 10).



S3 Reader		Output, Eve		Q   100	% Q
Join Component			9 1	: <u>+</u>	Ť
Basic Information Meta Informa	ation				
Invocation Type* 👻	Deployment Type spark	Container Image Version 6.5.0-R1			_
Failover Event 👻	Batch Size(min 10)				

- viii) Open the Meta Information tab and provide the connection-specific details.
  - a. Select the first data source.
  - b. Select a Join Type from the drop-down menu.
  - c. Select the Second Data Source.
  - d. Configure the Join Columns- Enter the First Column Name, Second Column Name, and select a Condition from the drop-down menu.
    - Note: The Condition field can be ignored for the single join.
  - e. To use multiple joins, click on the 'Add New Column' option.
  - f. Use a checkmark in the box beside the '**Null Safe**' option to include the 'null' values for join columns.
- ix) Click the 'Save Component in Storage' 🥙 icon to save the Join component properties.

Join Columns First Column Name*				
Einst Column Name?				
source1_name	Second Column Name* source2_first_name	Condition	- ×	
Add New Column				
Add New Column				

x) A message appears to notify the successful update of the component.

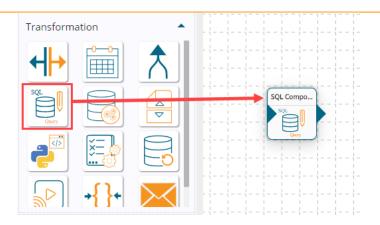


xi) The Join component is ready to read the data coming from the provided inputs, transform the data, and send the joined data to the output event.

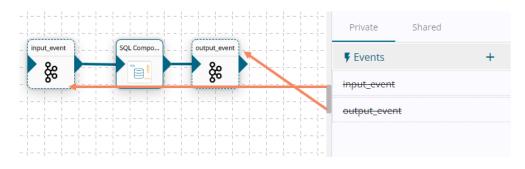
## 9.3.4. SQL Component

This component helps users to get data as per the entered query.i) Drag and Drop the SQL component to the Workflow Editor.





- ii) The transformation component requires an input event (to get the data) and sends the data to an output event.
- iii) Create two Events and drag them to the Workspace.
- iv) Connect the input event and the output event (The data in the input event can come from any Ingestion, Reader, or shared events).



- v) Click the SQL component to get the component properties tabs.
- vi) The **Basic Information** tab opens by default.
  - a. Select an Invocation type from the drop-down menu to confirm the running mode of the reader component. Select '**Real-Time**' or '**Batch**' from the drop-down menu.
  - b. Deployment Type: It displays the deployment type for the component. This field comes preselected.
  - c. Container Image Version: It displays the image version for the docker container. This field comes pre-selected.
  - d. Failover Event: Select a failover Event from the drop-down menu.
  - e. Batch Size (min 10): Provide the maximum number of records to be processed in one execution cycle (Min limit for this field is 10).

SQL Component	0	1111111	۵	<u>+</u>	Ť
Basic Information Met	ta Information				
Invocation Type*	Deployment Type     spark	Container Image Version 6.5.0-R1			
Failover Event	▼ Batch Size(min 10)				

- vii) Open the Meta Information tab and provide the connection-specific details.
  - a. Enter a valid data query to fetch data.
  - b. Provide the Table Name.



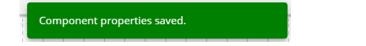
- c. Schema File Name: This option helps for schema conversion.
- d. Choose File: The user can browse a file and select it.
- e. **Selected Columns**: The users can select some specific columns from the table to read data instead of selecting a complete table; this can be achieved via the 'Selected Columns' section. Select the columns which you want to read and if you want to change the name of the column, then put that name in the alias name section otherwise keep the alias name the same as of column name and then select a Column Type from the drop-down menu. or

Use 'Download Data' and 'Upload File' options to select the desired columns.

- Upload File: The user can upload the existing system files (CSV, JSON) using the 'Upload File'
   icon (file size must be less than 2 MB).
- 2. Download Data (Schema): Users can download the schema structure in JSON format by using the 'Download Data' icon.

SQL Component	Ø		e	:	+	<u>+</u>
Basic Information	Meta Information					
Query* select first_name, la Table Name* employee	ast_named from employeeSchema File Name	Choose File	11			
Selected Colum	Alias Name	Column Type 🔹 🗙	[	×		

- viii) Click the 'Save Component in Storage' 🐸 icon to save the component properties.
- ix) A message appears to notify the successful update of the component.



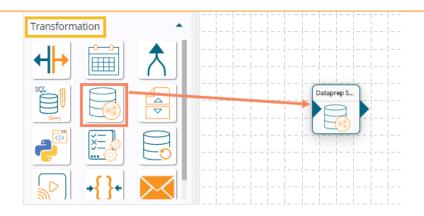
x) The SQL component gets configured to read the data coming from the input event, transform the data, and return it to the output events.

## 9.3.5. Dataprep Script Runner Component

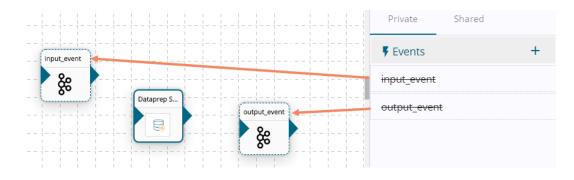
This transform component enables the users to use the exported Dataprep scripts in the Data Pipeline.

i) Drag and Drop the Dataprep Script Runner component to the Workflow Editor.





- ii) The Transformation component requires an input event (to get the data) and sends the processed data to an output event.
- iii) Create two Events and drag them to the Workspace.



iv) Connect the input event and the output event. (The data in input event can come from any Ingestion, Reader, or shared events)



- v) Click the Dataprep Script Runner component to get the component properties tabs.
- vi) **Basic Information**: It is the default tab to open for the Dataprep Script Runner while configuring the component.
  - a. Select an Invocation type from the drop-down menu to confirm the running mode of the reader component. Select '**Real-Time**' or '**Batch**' from the drop-down menu.
  - b. Deployment Type: It displays the deployment type for the component. This field comes preselected.
  - c. Container Image Version: It displays the image version for the docker container. This field comes pre-selected.
  - d. Failover Event: Select a failover Event from the drop-down menu.
  - e. Batch Size (min 10): Provide the maximum number of records to be processed in one execution cycle (Min limit for this field is 10).



	Input, Event	000000. Konno 86				
Dataprep Script Runner	I lee lee lee lee lee lee lee lee lee le		9		<u>1</u>	T
Invocation Type*	Deployment Type     spark	Container Image Version 6.5.0-R1				
Failover Event	Batch Size(min 10	1)				

- vii) Select the 'Meta Information' tab and fill all the required inputs fields.
  - a. Script Name: Select a script that is exported from the Dataprep Application.
  - b. Created On: By selecting a script the date of creation reflects by default.
  - c. Transformation Details: The transformation details get displayed below.
- viii) Click the '**Save Component in Storage**' Sicon to save the component properties.

Dataprep Script Run	ner 🖉				$\bigcirc$	: :	+	<u>+</u>
Basic Information	Meta Information							
Script Name* seer_problems	5	<b>*</b>	Created On 3/10/2021					
Transformatio	n Details							
Steps	Action Type			Column Names				
1	SEARCH_AND_REPLACE			Closed_Date				
2	SEARCH_AND_REPLACE			Title				
3	SEARCH_AND_REPLACE			Root_Cause				
4	CAST_TO_TYPE							

ix) A message appears to notify the successful update of the component.

Component properties saved.

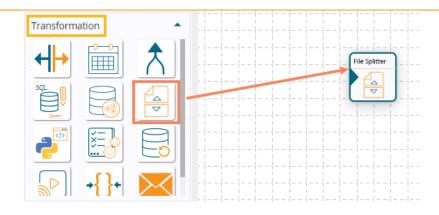
x) The Dataprep Script Runner component gets configured to read the data coming from the input event, transform the data, and return it to the configured output nodes.

#### 9.3.6. File Splitter

This transform component enables the users to split files.

i) Drag and Drop the File Splitter component to the Workflow Editor.





- ii) The Transformation component requires an input event (to get the data). The dragged File Splitter will not have the output connection. The user needs to configure it from the Meta Information tab.
- iii) Create one input event, drag it to the Workflow editor, and connect it with the File Splitter component.



- iv) Click the File Splitter component to get the component properties tabs.
- v) **Basic Information**: It is the default tab to open for the File Splitter component while configuring the component.
  - a. Select an Invocation type from the drop-down menu to confirm the running mode of the reader component. The supported invocation type for this component is Real-time.
  - b. Deployment Type: It displays the deployment type for the component. This field comes preselected.
  - c. Container Image Version: It displays the image version for the docker container. This field comes pre-selected.
  - d. Failover Event: Select a failover Event from the drop-down menu.
  - e. Batch Size (min 10): Provide the maximum number of records to be processed in one execution cycle (Min limit for this field is 10).

File Splitter		0		Ø	:	Ţ	Ť
Basic Information Me	ta Inforr	nation					
		Deployment Type	Container Image Version				
Invocation Type*	*	docker	6.5.0-R1				
Failover Event	Ŧ	Batch Size(min 10)					

- vi) Select the 'Meta Information' tab and fill all the required properties fields.
  - a. Split Type: Select a split type from the drop-down. The supported Split Types are given below:



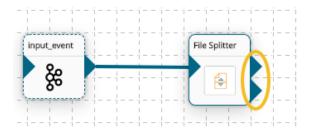
- 1. By File Format
- 2. By File Name
- 3. By RegExp
- 4. By Excel Sheet Name
- 5. By Excel Sheet Number

By File Format	
By File Name	
By RegExp	
By Excel Sheet Name	
By Excel Sheet Number	

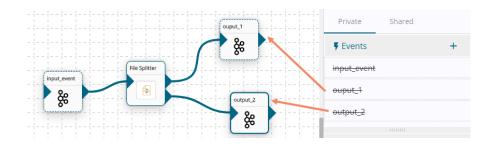
b. No. of Outputs: Select a number from the drop-down menu to create the Output nodes.

					2	Î	:	+	<u>+</u>
Meta Inf	ormation								
-	No. of Outputs* 2	*							
			No. of Outputs*	No. of Outputs*	No. of Outputs*	Meta Information	Meta Information	Meta Information	No. of Outputs*

vii) By selecting the '**No. of Outputs**' for the File Splitter component the dragged File Splitter component gets those many output nodes.



viii) Create and connect the output events to the output nodes of the File Splitter component (E.g., In the present case, 2 output events are created and connected to the File Splitter component).



ix) Open the Meta Information tab and provide the connection-specific details.



- x) The Details field displays both the connected output events.
- xi) Select File Type for the given output event as displayed in the following image (the supported file types are PDF, CSV, Excel, Others).

File Splitter			
Basic Information	Meta Informatio	on	
Split Type*		No. of Outputs*	
By File Format	•	2	*
Details			
Out Event*		File Type *	
ouput_1	-	CSV	*
Out Event*		File Type *	
output_2	Ŧ	EXCEL	*

xii) Click the 'Save Component in Storage' icon to save the component properties.



xiii) A message appears to notify the successful update of the component.



xiv) The File Splitter component gets configured to read the data coming from the input event, transform the data, and return it to the configured output nodes.

# 9.3.7. Python Script (Custom Python Script)

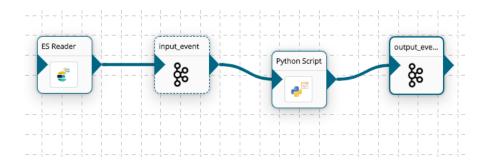
Python Script component works as a normal python compile.

i) Drag and drop the Python Script to the Workflow Editor.





- ii) The Python script will read data from an input event and will pass the processed data into an output event, so create and connect two events to the Python Script component.
- iii) One reader is needed to pass the data to the input event. (in this case, the ES Reader component is used).
- iv) Connect the ES Reader, the Kafka Events, and Python Script components as displayed below:



- v) Click the dragged Python Script component to get the component properties tabs:
- vi) **Basic Information**: It is the default tab to open for the Python Script component while configuring the component.
  - a. Select an Invocation type from the drop-down menu to confirm the running mode of the reader component. The supported invocation type for this component is Real-time.
  - b. Deployment Type: It displays the deployment type for the component. This field comes preselected.
  - c. Container Image Version: It displays the image version for the docker container. This field comes pre-selected.
  - d. Failover Event: Select a failover Event from the drop-down menu.
  - e. Batch Size (min 10): Provide the maximum number of records to be processed in one execution cycle (Min limit for this field is 10).

Python Script			111111	9	Î	:	<u>+</u>	Ť
Basic Information Meta	Inform	ation						
Invocation Type*	•	Deployment Type docker	Container Image Version 6.5.0-R1					
Failover Event	•	Batch Size(min 10)						

- vii) Open the 'Meta Information tab to open the fields and configure them.
  - Component Name: Provide a name for the Python Script component.
     Note: The component name should be without space and special characters. Use the underscore symbol to show space in between words.
  - b. Python Script: Insert the Python script containing at least one function. The function should not have an argument, data frame argument, or custom argument.
  - c. Start Function Name: It displays all the function names used in the python script in a drop-down menu. Select one function name with which you want to start.
  - d. In Event Data Type: Provide input data type as a data frame or list.
  - e. External Library: Provide the external library name in this field. Insert multiple library names separated by commas.
  - f. Input Data: Use custom argument names as keys and provide the required value.
- viii) Click the 'Save Component in Storage' 🥝 icon.

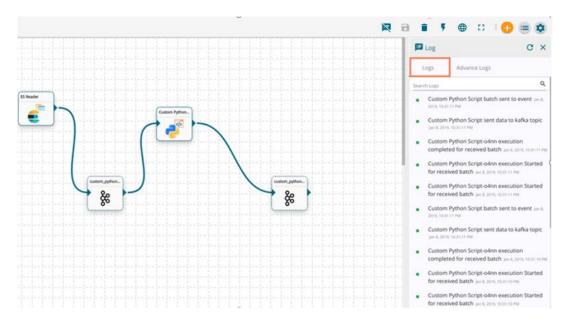


Python Script				1			${}$	Î		+	Ť
Basic Information	Meta Infor	mation									
Component Name* Smaple Python Script		<sup>Script*</sup> import requests import pandas as pd									ľ
Start Function* testingpythonn	*	Import json	•	External Librar	ries [*Library	name se	eperate	d by c	:omm	a (,)]	~
Input Data											
Key	Value	×									
Add New Colun	n										

- ix) Click the 'Update Pipeline' icon to save the Pipeline workflow. (After getting the success message)
- x) Activate the Pipeline workflow.



xi) Open the Log section to see the logs.



xii) The Python Script component is ready to read the data coming from the input event, it transforms the data and returns output data.

#### Note:

- a. The below-given instructions should be followed while writing a Python script in the Data Pipeline:
  - The Python script needs to be written inside a valid Python function. E.g., The entire code body should be inside the proper indentation of the function (Use 4 spaces per indentation level.)
  - The Python script should have at least one main function. Multiple functions are acceptable, and one function can call another function.



- It should be written above the calling function body (if the called function is an outer function)
- It should be written above the calling statement (if called function is an inner function)
- Spaces are the preferred indentation method.
- Do not use "type" as the function argument as it is a predefined keyword.
- The code in the core Python distribution should always use UTF-8.
- Single-quoted strings and double-quoted strings are considered the same in Python.
- All the packages used in the function need to import explicitly before writing the function.
- The Python script should return data in the form of a data frame or list only. The form of data should be defined while writing the function.
- If the user uses some Kafka event data for transformation, then the first argument of the function should be a data frame or list.
- If the user needs to use some external library, the user needs to mention the library name in the external libraries field. If the user wants to use multiple external libraries, the library names should be separated by a comma.
- If you need to pass some external input in your main function, then you can use the input data filed. The key name should be the same according to the variable's name and value that is put as per the requirement.
- We can use that component as a reader, transformation, and writer.
- b. Python Script Examples

The Custom Python Script transform component supports 3 types of scripts in the Data Pipeline.

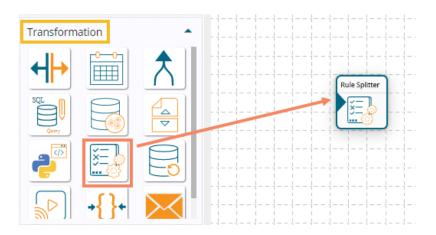
- **1.** Without argument (Like Read Data): If you don't have any in Event then you can use no argument function. For Example,
  - i. import json
  - **ii.** import requests
  - iii. import pandas as pd
  - iv. def getmovies\_result():
  - v. data = requests.get("http://www.omdbapi.com/?s=water&apikey=ba5d53d4")
  - **vi.** loaded\_json = json.loads(data.content)
  - **vii.** data = loaded\_json['Search']
  - viii. df = pd.DataFrame.from\_dict(data, orient='columns')
  - ix. return df
- 2. With argument (Like Transformation): If you have data frame to execute some operation. Then use first argument as data frame. For Example,
  - i. def getcsvdata(df):
  - ii. cond1 = df['Unit Price'] > 450
  - iii. filter\_df = df[cond1]
  - iv. return filter\_df
- **3.** Custom Argument with Data frame: If there is custom argument with data frame means Input Event then first argument is always data frame variable. For Example,
  - i. def getcsvdata(df, range):
  - ii. cond1 = df['Unit Price'] > range
  - iii. filter\_df = df[cond1]
  - iv. return filter\_df

# 9.3.8. Rule Splitter

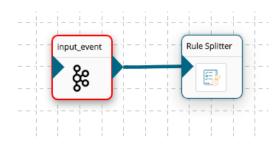
This component in the Pipeline helps the users to apply a filter on the data as per the condition and move to separate events.

i) Drag and drop the Rule Splitter component to the Workflow Editor.





- ii) The transformation component requires an input event (to get the data) and sends the data to an output event.
- iii) Create an input event and connect it to the Rule Splitter component.



- iv) Click the Rule Splitter component to get the tabs containing the component properties.
- v) The Basic Information tab opens by default.
  - a. Select an Invocation type from the drop-down menu to confirm the running mode of the reader component. Select '**Real-Time**' or '**Batch**' from the drop-down menu.
  - b. Deployment Type: It displays the deployment type for the component. This field comes preselected.
  - c. Container Image Version: It displays the image version for the docker container. This field comes pre-selected.
  - d. Failover Event: Select a failover Event from the drop-down menu.
  - e. Batch Size (min 10): Provide the maximum number of records to be processed in one execution cycle (Min limit for this field is 10).

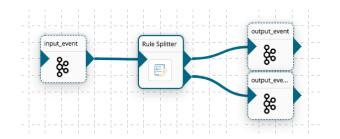
Rule Splitter	0		Ø	Î	:	<u>+</u>	Ť
Basic Information Meta Info	ormation						
	Deployment Type	Container Image Version					
Invocation Type* 👻	spark 	6.5.0-R1					
Failover Event 🔹	Batch Size(min 10)						

- vi) Open the **Meta Information tab** and configure the required information:
- vii) Select the Number of Outputs from the drop-down menu.



Rule Splitter	0	
Basic Information	Meta Information	
Number of Outputs*	<b>.</b>	

- viii) Based on the No. of Outputs the Rule Splitter component gets the output nodes.
- ix) Connect the output events to the Rule Splitter and create a workflow as given below:



- x) Open the 'Meta Information tab and provide the connection-specific details.
- xi) Configure the 'Event Relation' fields.
  - i. Column Name- Provide a name for the column
  - ii. Condition- Select a condition from the provided options
  - iii. Value- Enter a defining value for the column
  - iv. Data Type (String/Integer/Double/Date)
  - v. Rule Condition (AND/OR)
- xii) Click the 'Save Component in Storage' C icon after providing the required information.

vent Relation						
Out Event*						
output_event	•					
Condition(s)						
Column Name*	Condition*	Value*	Data Type *	Rule Condition*		
col 1	>	▼ 25	Integer	AND	-	×
Add New Condition						
Add New Condition						
Out Event*						
output_event2	~					
output_event2						
Condition(s)						
Condition(s)						
Condition(s) <sup>Column Name*</sup> col 2	Condition*	value* ▼ Value	Data Type * String	Rule Condition*	-	×

xiii) A message appears to notify the successful update of the component.



xiv) The Rule Splitter component is ready to read the data coming from the input event, transform the data, and return output data (based on the configured output nodes).



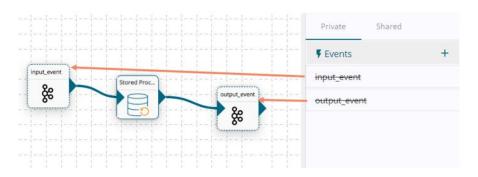
## 9.3.9. Stored Procedure Runner

Stored Procedure Runner Component can be used to run a stored procedure using a Pipeline. It is a **Batch** component. As soon as the data comes from the previous components the Stored Procedure Runner gets triggered and performs its task only once for that batch. This Transformation component cannot be used with Stream data because the velocity of data in the case of streaming is very high.

i) Drag and drop the Stored Procedure Runner component to the Workflow Editor.



- ii) The transformation component requires an input event (to get the data) and sends the data to an output event.
- iii) Create one input event and an output event, connect them to the Stored Procedure Runner component.



- iv) Click the Stored Procedure Runner component to get the tabs containing the component properties.
- v) The Basic Information tab opens by default.
  - a. Select an Invocation type from the drop-down menu to confirm the running mode of the reader component (Only the '**Batch**' invocation type is supported).
  - b. Deployment Type: It displays the deployment type for the component. This field comes preselected.
  - c. Container Image Version: It displays the image version for the docker container. This field comes pre-selected.
  - d. Failover Event: Select a failover Event from the drop-down menu.
  - e. Batch Size (min 10): Provide the maximum number of records to be processed in one execution cycle (Min limit for this field is 10).



Stored Procedure Runner		2		Ø	Î	:	<u>+</u>	Ť
Basic Information Met	ta Inform	nation						
Invocation Type*	Ŧ	Deployment Type docker	Container Image Version 6.5.0-R1					
Failover Event	*	Batch Size(min 10)						

- vi) Open the **Meta Information tab** and provide the connection-specific details.
- vii) Configure the mandatory fields by providing the required information:
  - a. Host (JDBC Host)
  - b. Port
  - c. Username
  - d. Password
  - e. Database Name
  - f. Procedure Name (Name of the Stored Procedure)
  - g. Driver: Select a driver using the drop-down menu
  - h. Get Data: Enable this option using a checkmark to get returned data from the Stored Procedure

Stored Procedure Run	nner 🖉		9 🖬 🗄	+	Ţ
Basic Information	Meta Information				
Host IP Address*	Port*	Username*	Password*		
Database Name*	Procedure Name*	Driver*	📕 🗌 Get Data		

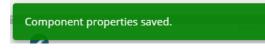
- viii) Configure the Stored Procedure specific Input and Output parameters:
  - a. Input Parameters
    - i. Parameter Name
    - ii. Parameter Value
    - iii. Parameter Type
  - b. Output Parameters
    - i. Parameter Name
    - ii. Parameter Type

Parameter Name parameter	Parameter Value values		Parameter Type integer	~	>
Add New Column					
Output Parameters					
Dutput Parameters	Parameter Type Integer	- )	×		

ix) Click the 'Save Component in Storage' 🗹 icon.



x) A message appears to notify the successful update of the component properties.



xi) The Stored Procedure Runner component is ready to read the data coming from the input event, transform the data, and return output data to the output events.

Note: Data from an Input Event cannot be taken as Input Parameter for the Stored Procedure Runner component.

## 9.3.10. Stream Component

The steps to configure the Stream component is described in this section.

i) Drag and drop the Stream Component to the Pipeline Workflow Editor.



- ii) Click the dragged Stream component to get the Configuration fields:
- iii) The **Basic Information tab** opens by default.
  - a. Select the invocation type (Real-Time/Batch)
  - b. Deployment Type: It comes preselected based on the component.
  - c. Container Image Version: It comes preselected based on the component.
  - d. Failover Event: Select a failover Event from the drop-down menu.
  - e. Batch Size (min 10): Provide the maximum number of records to be processed in one execution cycle (Min limit for this field is 10).

Stream Component	2		9	0 0	<u>+</u>	Ť
Basic Information Meta Inform	ation					
Invocation Type* •	Deployment Type docker	Container Image Version 6.5.0-R1				
Failover Event 💌	Batch Size(min 10)	_				

- iv) Open the Meta Information tab and provide the connection-specific details.
  - a. Input Fields: The user needs to provide all/selected columns that the IoT device is producing along with their Data Type.
    - 1. Field Name: Provide a name for the field.
    - 2. Filed Type: Select a field type from the drop-down menu (The supported field types are String, Long, Integer, Float, Double, Date).



- b. Window Field: The user needs to provide the field using which we want to do bucketing of data. This field must be a timestamp value.
- c. Window Time: Time duration of Window (Bucket).
- d. Allowed Lateness: Allowed time up to which the component accepts late data.
- e. Aggregation Field: The user needs to provide a field using which component performs aggregation operations on operation fields.
- f. Operations: The user needs to select the field name, operation name, and provide the output field label. Multiple operations are also supported.

Stream Component 📀				۲		+++++++++++++++++++++++++++++++++++++++	1
Basic Information Meta I	nformation						
Input Fields							
Field Name	Field Type	- ×					
Add New Field							
Window Field* 🔻	Window Tim	Slide Time(in	Allowed Late	Aggrega	tion Field*		•

- g. Operations: Configure the following details.
  - 1. Field Name: The configured input fields appear in the drop-down. Select a field from the drop-down menu.
  - 2. Operation Name: Select an option to select the Operation name.
  - 3. Output Label: Provide an output label.

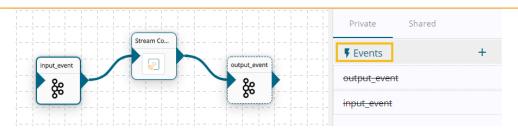
Operations					
Field Name*		Operation Name		Output Label*	
Field	-	Max	-	Field	×
Add New Operat	ion				

- v) Click the 'Save Component in Storage' 🐸 icon.
- vi) A message appears to notify the action.



vii) The Stream Component is ready to be used in the Pipeline workflow. It can read data from an input event and pass the processed streaming data to an output event, so accordingly connect the required event components to create a pipeline workflow.





Note:

- a. The user can add a new field and new operation by clicking the 'Add New Field' and 'Add New Operation' options, respectively.
- b. Windowing: Windowing is an operation in spark structured streaming where the stream of data gets divided into a small amount of windows and then performs aggregation operations on the data.

For detailed information, please visit this link: <u>https://spark.apache.org/docs/2.3.0/structured-streaming-programming-guide.html#window</u> operations-on-event-time

## 9.3.11. Flatten JSON

The pipeline provides the Flatten JSON component for flatting a nested JSON as well as filtering data from a nested JSON. It can filter and remove the unnecessary dumping of data on a Kafka topic. It can run in the following scenarios:

- 1. Real-Time
- 2. Batch Process

The steps to configure the Flatten JSON component are described in this section.

i) Drag and drop to the Flatten Json component to the pipeline Workflow Editor.



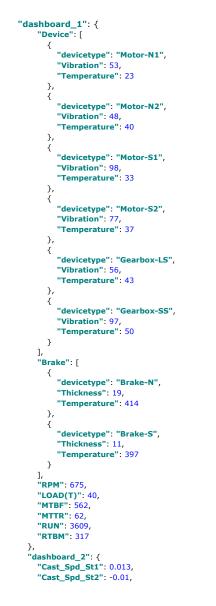
- ii) Click the dragged Flatten JSON component to get the Configuration fields:
- iii) The Basic Information tab opens by default.
  - a. Select the invocation type (Real-Time/Batch)
  - b. Deployment Type: It comes preselected based on the component.
  - c. Container Image Version: It comes preselected based on the component.
  - d. Failover Event: Select a failover Event from the drop-down menu.
  - e. Batch Size (min 10): Provide the maximum number of records to be processed in one execution cycle (Min limit for this field is 10).



Flatten Json		Ø	Î	:	<u>+</u>	Ť
Basic Information Meta Information						
Deployment Type	Container Image Version					
Invocation Type* 🔻 docker	6.5.0-R1					
Failover Event <ul> <li>Batch Size(min 10)</li> </ul>						

- iv) Open the Meta Information tab to provide the connection-specific details.
  - a. Name: The user must provide the full path of data by using dot(.) which he wants to retrieve from a nested JSON.

For instance - We have a nested JSON of the following format, and we want to retrieve the device type of the Device in dashboard\_1 we must give the name **dashboard\_1**. **Device.devicetype**.



v) Selected **Columns**: The users can select some specific columns from the table to read data instead of selecting a complete table; this can be achieved via the 'Selected Columns' section.



Select the columns which you want to read and if you want to change the name of the column, then put that name in the alias name section otherwise **keep the alias name the same as of column name** and then select a Column Type from the drop-down menu. **or** 

Use 'Download Data' and 'Upload File' options to select the desired columns.

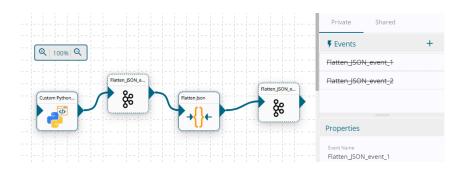
- 1. Upload File: The user can upload the existing system files (CSV, JSON) using the '**Upload File**' icon (file size must be less than 2 MB).
- 2. Download Data (Schema): Users can download the schema structure in JSON format by using the '**Download Data**' icon.
- vi) Click the 'Save Component in Storage'<sup>(S)</sup> icon to save the configuration.
- vii) A message appears to notify the action.

Component properties saved.

viii) The Flatten JSON is ready to be used in the Pipeline workflow.

#### Note:

- a. The Flatten JSON component cannot generate data in rows.
- b. The Flatten JSON component must have an in-event and out-event in the workflow, as displayed below:



## 9.3.12. Email Component

The Email Component helps to configure the Email content together with the Email server and recipients.

i) Drag and drop the Email Component to the Pipeline Workflow Editor.



ii) Click the dragged Email component to get the Configuration fields:



- iii) The **Basic Information tab** opens by default.
  - a. Select the invocation type (Real-Time/Batch)
  - b. Deployment Type: It comes preselected based on the component.
  - c. Container Image Version: It comes preselected based on the component.
  - d. Failover Event: Select a failover Event from the drop-down menu.
  - e. Batch Size (min 10): Provide the maximum number of records to be processed in one execution cycle (Min limit for this field is 10).

Email Component		٥		6	3	Î	:	<u>+</u>	Ť
Basic Information	ta Inform	nation							
		Deployment Type	Container Image Version						
Invocation Type*	~	docker	6.5.0-R1						
Failover Event	-	Batch Size(min 10)							

- iv) Open the **Meta Information tab** and provide the connection-specific details.
  - a. Subject (\*): Subject of the Email
  - b. Text Message: Provide the email content in the given space.

c Infor		on	N	Vleta	Infor	rmati	on	_													
oject*												 					 				
HTML			7		e			-	-	-	-			1-			Artal				
	C A		I Class					≡				-	:=	;=	defau	r.	\$ Arial	•	3	Ŧ	
Ente	er te	xt he	re																		

- c. Attachment: Select an option out of 'Yes' or 'No'.
- d. Receivers (\*): Provide receivers' email ids separated by commas.
- e. Host (\*): Provide host details.
- f. TLS: Enable or disable TLS by putting a checkmark in the given box.
- g. Email Username: Provide the name of the email user.
- h. Encryption Type: Select an encryption type from the drop-down menu.
- i. Email Password: Provide the password of the email
- j. Enable SSL: Enable SSL by putting a checkmark in the given box.
- k. Email From: Provide the email address of the senders.
- I. Email Port: Provide the Port number of the email.
- m. Disable Email Sending: Put a tick mark to disable the sending of the notification emails.
- n. Email Input: Provide the following details.
  - 1) Path
  - 2) Input Variable
  - 3) Data Type



Attachment Yo You You You You You You You You You	Receivers [*receiver email se	par	Host* mail.smtp2go.com				
TLS	Email User Name		Encryption Type				
Email Password	Enable SSL		Email From				
Email Port	Disable Email Sending						
Email Input							
Path	Input Variable	Data Type	• ×				
Add New Input							

- v) Click the 'Save Component in Storage' icon.
- vi) A message appears to notify the action.



vii) The Email Component is ready to be used in the Pipeline workflow. It can read data from an input event and pass the processed data to an output event, so accordingly connect the required event components to create a pipeline workflow.

## 9.4. ML

ML Model Runner Components allow us to use the models created on R and Python Workspaces of the Data Science Workbench inside the Data Pipeline.



## 9.4.1. R Model Runner

i) Drag and Drop R Model (runner) component to the Workflow Editor.

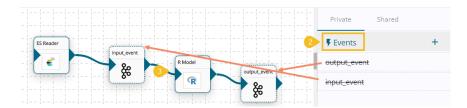


ii) Model Runner requires data input from an Event and sends the processed data to another Event, so create two Events.



iii) Drag them to the workflow editor and connect to the dragged R Model (runner) component as displayed below:

Note: The data in Input Event can come from any Ingestion, Reader, or shared events.



- iv) Click on the R Model Component to get the Component Properties tabs.
- v) Basic Information: It is the default tab to open for the component.
  - a. Select an Invocation type from the drop-down menu to confirm the running mode of the reader component. Select '**Real-Time'** or '**Batch**' from the drop-down menu.
  - b. Deployment Type: It displays the deployment type for the component. This field comes preselected.
  - c. Container Image Version: It displays the image version for the docker container. This field comes pre-selected.
  - d. Failover Event: Select a failover Event from the drop-down menu.
  - e. Batch Size (min 10): Provide the maximum number of records to be processed in one execution cycle (Min limit for this field is 10).

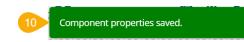
	Event	Output liver	
	<b>- <u> </u></b>  -   -		
R Model		9 1	1
	eta Information	9 i :	Ŧ
	eta Information	o 🔋 🔋	Ŧ
	eta Information	Container Image Version	Ŧ
			Ţ

- vi) Click the Meta Information tab to open.
- vii) All the exported R models appear under the '**Model Name**' menu. Search and select the R model you want to use.
- viii) Configure the URL link (The URL link comes predefined and can be changed by the user).
- ix) Click the 'Save Component in Storage' icon.

R Model	9>❷ 🛢 🗄 🕂 ⊥
Basic Information 6 Meta Information	
Search List	
Model_app_prod_bdb_ai	8 URL* https://qa.bdb.ai
7 reg_model	
pipeline_outliersal	
pipeline_outlierexp	
pipeline_predictsalary	



x) A message appears to confirm the action.



xi) The R Model runner is configured. The component reads the data coming to an input event, runs the model, and gives the output data with processed columns to the output event.

## 9.4.2. Python Model Runner

i) Drag and drop the Python Model (runner) component to the Workflow Editor.



- ii) The Python Model runner requires input data from an Event and sends the processed data to another Event. So, create two Events and drag them onto the Workspace.
- iii) Connect the input and output events with the Python Model runner component as displayed below.
- iv) The data in the input event can come from any Ingestion, Readers, or shared events.



- v) Click the Python Model runner component to get the component properties tabs below.
- vi) Basic Information: It is the default tab to open for the component.
  - a. Select an Invocation type from the drop-down menu to confirm the running mode of the reader component. Select '**Real-Time**' or '**Batch**' from the drop-down menu.
  - b. Deployment Type: It displays the deployment type for the component. This field comes preselected.
  - c. Container Image Version: It displays the image version for the docker container. This field comes pre-selected.
  - d. Failover Event: Select a failover Event from the drop-down menu.
  - e. Batch Size (min 10): Provide the maximum number of records to be processed in one execution cycle (Min limit for this field is 10).



S Reader	vent	Output River &	
Python Model Basic Information Meta Information			Ø <b>i</b> : <u>⊥</u> ⊺
Invocation Type* 💌	Deployment Type docker	Container Image Version 6.5.0-R1	
Failover Event	Batch Size(min 10)		

- vii) Open the Meta Information tab and click on the Model name field.
- viii) The model type displays the Python Model and NN Model options.
- ix) Select a model Type option.

Python Model	0
Basic Information	Meta Information
Python Model	
NN Model	

- x) Search the model you want to use and select the model.
- xi) Click the 'Save Component in Storage' 🥝 icon.

Python Model	0			${}^{{}_{{}_{{}_{{}_{{}_{{}_{{}_{{}_{{}_{$	1	<u>+</u> ↑	<u> </u>
Basic Information	Met	a Information					
Model Type Python Model	*	Model Name* pyhtontest_dm	•				

xii) A notification message appears.



xiii) The Python Model runner component reads the data coming to the input event, runs the model, and gives the output data with predicted columns to the output event.

#### 9.5. Ingestion

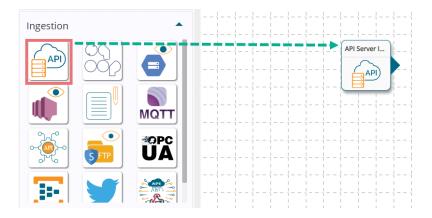
Ingestion components allow the users to ingest data in the pipeline from outside the pipeline as in files from some SFTP location or manual ingestion of data using a service in real-time.





# 9.5.1. API Server Ingestion

i) Drag and Drop the **API Server Ingestion** component to the Workflow Editor.



- ii) Click on the dragged ingestion component to get the component properties tabs.
- iii) Configure the **Basic Information** tab.
  - a. Select an Invocation type from the drop-down menu to confirm the running mode of the reader component. The supported invocation type is '**Real-Time**'.
  - b. Deployment Type: It displays the deployment type for the component. This field comes preselected.
  - c. Container Image Version: It displays the image version for the docker container. This field comes pre-selected.
  - d. Failover Event: Select a failover event from the drop-down menu.
  - e. Batch Size (min 10): Provide the maximum number of records to be processed in one execution cycle (Min limit for this field is 10).

API Server Ingestion		0		٢	Î	:	<u>+</u>	Ť
Basic Information	Meta Inforn	nation						
Invocation Type*	Ŧ	Deployment Type docker	Container Image Version 6.5.0-R1					
Failover Event	•	Batch Size(min 10)						

- iv) Open the 'Meta Information' tab.
  - a. Provide the Ingestion Id
  - b. Provide the Ingestion Secret (Key)



- c. Write a Message (optional)
- d. The Component Instance Id URL generates (after providing the Ingestion Id and Secret Key).
- v) Click the 'Save Component in Storage' 🥌 icon to save the API Server Ingestion.

PI Server Ingestion	${ \  \  \  \  \  \  \  \  \  \  \  \  \ $	Î	:	+	Ŧ
Basic Information Meta Information					
Ingestion Id* 5b2c8240-463c-46df-5 exYvsb+N1nllMsO4fv3Xpvm9Ayyx7bRrDq/YdnAd+zir					
Message					
Component Instance Id URL					

- vi) A notification message appears to inform that the component has been saved.
- vii) Connect the API Ingestion component to an Event and save the Pipeline.

	🗩 🕒 🗊 🤻 🖶 🖂 🗐	•
	Private Shared	
API Serv     input_event	F Events	+
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~	input_event	

viii) As we save the pipeline, an auto-generated Component **Ingestion URL** appears inside the Meta Information tab to ingest the data.

API Server Ingestion	2		2	Î	:	<u>+</u> ↑	<u>+</u>
Basic Information	leta Information	Configuration					
Ingestion Id* 5b2c8240-463c-46df-9	Ingestion Secret* exYvsb+N1nIIM	IsO4fv3Xpvm9Ayyx7bRrDq/YdnAd+zir					
Message							
Component Instance Id URL comp1585912240204-inst	-6987.api-server-ing	estion.pipeline.bdbizviz.com/api/v1/in		Сор	y Link		

- ix) Click the 'Save Component in Storage' 🥙 icon.
- x) A notification message appears to inform about the action.



xi) The Ingestion component can be used in a pipeline workflow as an input component after component configuration.

Note:

a. The users can use the Component Ingestion URL with the following format in the program or add it anywhere in the third-party portal.



New • Import Runner 🖗 *	Builder Team Library 😪 🧿 🖲		6 V (†
	Bring your whole team to Postman for free! Update the app to experience the new features. Deatload		
Q apl-s ×	Comp1545293236482 • + ++	Environment	~ ©
History Collections	POST V comp1545293236482-inst-5523.api-server-ingestion.pipeline.bdbizvtz.com/api/v1/ingestion/event Pa	irams Send ~	Save
August 21	Authorization Headers (1) Body  Pre-request Script Tests		Cookles Co
PORT         comp1534837278009-inst-2613.api-serv           er-ingestion pipeline biblishitz.com/api/v         fingestion/event           PORT         comp1534722534521-inst-1734.api-serv           er-ingestion pipeline biblishitz.com/api/v         fingestion pipeline biblishitz.com/api/v	<pre>Official &amp; xwwwform-urlencedd * raw &amp; Binary (SORYapplication/poin) ~</pre>	, \"sales_id\" : sale_23	1324,
R051 comp1534772534521-inst-1734.api-serv	7 x		
er-ingestion.pipeline.bdbizviz.com/api/v 1/ingestion/event			
1/ingestion/event POST comp1534772534521-inst-1734.api-serv er-ingestion pipeline.bdbizviz.com/api/v			

b. Click the List Ingestions option to open the list of ingestions. Select an Ingestion from the list. The Ingestion Details opens on the top-right side of the page with Ingestion ID and Ingestion Secret (Key).

Ingestion List				Search Ingestion Q 🕂
A Home	Name	Ingestion Type	Actions	Ingestion Details
:≡ List Pipelines	abc	system	/ 1	Ingestion Name abc
+ Create Pipeline	test	system	/ 1	Ingestion ID 5b2c8240-463c-46df-9
List Components	stream	system	/ 1	Column Meta Info
Scheduler	Nandita Vidya	system	2 II.	Name Data Type Parent Node
Log Destination	anuj	system	1	Id string -
O Data Channel	Anuj_ingestion	system	× •	
Settings		Items per page: 10 💌	1-6of6  < < >	

#### Service Info:

- IngestionID and Ingestion Secret (key) must be the same as filled in a pipeline component.
- A message accepting data in Array as String format appears.
- In the message, all the objects must follow the same column info as declared in the settings page at the time ingestion gets created.

```
URL → comp1545293236482-inst-5523.api-server-

ingestion.pipeline.bdbizviz.com/api/v1/ingestion/event (will be Auto-generated Component Ingestion

URL)

Type → POST

Content-Type → application/json

Body →

{

"apiVersion": "1.0",

"ingestionId": "b1d9609a-38eb-48c7-b1c1-d3baa78ea37b",

"ingestionSecret":

"DbKNKwB7sTekPmKwNdjOvx2wQgUwsek+fQJDne4b6XOt1nQc8zwdODnU4xHj/L+I",
```



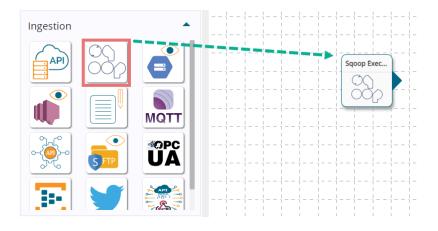
```
"message": "[{\"id\": 213350, \"sales_id\": sale_231323, \"description\": \"Pipeline lead\", \"status\"
: \"finalized\"},{\"id\": 213351, \"sales_id\": sale_231324, \"description\": \"Pipeline lead\", \"status\"
: \"initialize discussion\"}]"
```

}

#### 9.5.2. Sqoop Executer

The Sqoop component in BDB Pipeline helps the users to transfer tables and databases from RDBMS (Supported by Sqoop) to HDFS (Hadoop System Distributed File).

i) Drag and drop the Sqoop component to the pipeline Workflow Editor.



- ii) Click the Sqoop Executer to see the component properties tabs below.
- iii) The **Basic Information tab** opens by default by clicking the Sqoop Executer component.
  - a. Select an Invocation type from the drop-down menu to confirm the running mode of the reader component. The supported invocation type is '**Real-Time**'.
  - b. Deployment Type: It displays the deployment type for the component. This field comes preselected.
  - c. Container Image Version: It displays the image version for the docker container. This field comes pre-selected.
  - d. Failover Event: Select a failover event from the drop-down menu.
  - e. Batch Size (min 10): Provide the maximum number of records to be processed in one execution cycle (Min limit for this field is 10).

Sqoop Executer		0		Ø	Î	:	Ť	Ť
Basic Information	Meta Inform	nation						
Invocation Type*	•	Deployment Type docker	Container Image Version 6.5.0-R1					
Failover Event	•	Batch Size(min 10)						

- iv) Open the **Meta Information tab** and provide the following information:
  - a. Username: Provide the username.
  - b. Host Provide a host or IP address.
  - c. Port- Provide a Port number (the default number for these fields is 22).
  - d. Authentication- Select an authentication type from the drop-down menu out of the given choices



- i. Password- Provide the correct password for this authentication option
- ii. PEM/PPK File-choose a file and provide the file name if the user selects this authentication option.
- e. Command-Enter the relevant Sqoop command.

Sqoop Executer	0			٢		:	+	<u>+</u>
Basic Information	Meta Information							
Username*	host*	Port 22	 Password		1			
Command*			PEM / PPK File	e				

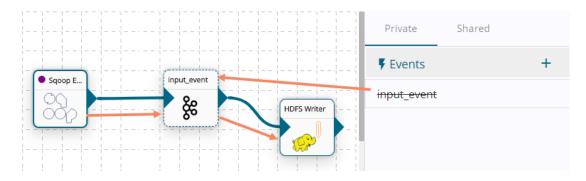
v) Click the 'Save Component in Storage' icon to save the component.



vi) A notification message appears to inform about the action.



vii) Create one Event and connect it to the Sqoop component for receiving the success message. Connect an HDFS writer to the input event to create a workflow. Once the data gets transferred to the HDFS, the user can use it for further processing in the pipeline.



#### 9.5.3. GCS Monitor

#### 9.5.3.1. Creating GCS (Google Cloud Storage) Credential

The users require Google credentials to access the GCS bucket via the GCS monitor. The steps to create a service account key are described below:

- i) Open Google Cloud Console or open the link- <u>https://console.cloud.google.com/home</u>
- ii) Click on the menu button (from the top left side) to open the list of options
- iii) Select and click the **APIs & Services** option from the menu list

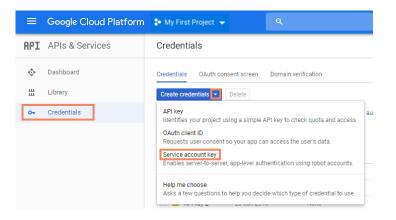


- iv) A context menu opens as displayed in the image below
- v) Select the **Credentials** option from the context menu

Google Cloud Platforn	n 🔹 My First Project 👻		ii 🖬 🗩 🖗 🌲	1
Home			/	CUSTOMIS
Getting started	ect info	: RPI APIS	i Google Cloud Platform status i	
Marketplace APIs & Services	t name st Project	Requests (requests/sec)	All services normal	
APIs & Services	Dashboard     Library		A A s Go to Cloud status dashboard	
IAM & admin Security	Credentials	mmmm	A Billing :	
Billing	project settings		Estimated charges     INR ₹0.00     For the billing period 1–16 Jan 2019	
Support	, e	1	2 → View detailed charges	
IPUTE	ce data from the last 7 days	Requests: 4.733		
App Engine	>	→ Go to APIs overview	Error Reporting	

Note: The user can ignore the above-given steps to create credentials if a Service Account key is already created.

- vi) The 'Credentials' opens as displayed in the below image.
- vii) Click the 'Create Credentials' drop-down option.
- viii) A menu opens with 3 types of authorization options:
  - 1. API Key
  - 2. OAuth Client ID
  - 3. Service account key
- ix) Click the 'Service Account Key' authorization option.



- x) The 'Create Service Account Key' page opens.
- xi) The users need to provide the following parameters:
  - 1. Services account: Select 'New Service Account' from the drop-down menu.
  - 2. Service account name: Provide a user-defined name for the Service Account. (E.g., 'bizviz' in the image below)
  - 3. Role: Select the 'Storage' option from the drop-down menu.
    - a. A new context menu opens.
    - b. Select 'Storage Object Admin' from the context menu.
  - 4. Key type: Select JSON by choosing the checkmark provided next to the option.
- xii) The Service Account Key gets created.



	Platform 💲 My F				٩
<ul> <li>Create service a</li> </ul>	account key				
Service account					
New service account		*			
2 Service account name 💿	3	Role 💿			
bizviz		Storage Object Ad 👻			
Service account ID		Selected			
bizviz	@dark-lexicon-228111.iam	.gs 🗸 Storage Object Admin			
Very type           Downloads as file that contains th cannot be recovered if lost.           • JSOM           • Recommended           • P12           For backward compatibility w		Kubernetes Engine Logging Monitoring	<pre>&gt; * &gt; * &gt; * &gt; * &gt; * * * * * * * * * * *</pre>	Storage Admin	
		Manage roles		<ul> <li>Storage Object Admin Storage Object Creator Storage Object Viewer</li> </ul>	Full control of GCS objects.

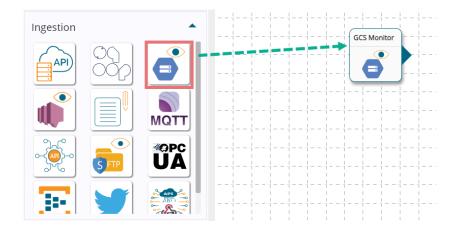
Note: After the successful creation of the Service Account, the users should get the Credential JSON file name according to their project name it indicates that the Service Account Key file got successfully created.

#### 9.5.3.2. GCS Monitor in Data Pipeline

GCS Monitor monitors one folder all the time. If any file has been uploaded, the GCS Monitor reads the file name and sends it to the event. The GCS Monitor also copies that file to the copied (defined) location and deletes it from the monitoring folder. It repeats the same process for all the files.

Steps to create one pipeline using the GCS Monitor component.

i) Drag and drop the GCS Monitor component to the Workflow Editor.



- ii) Click the dragged GCS Monitor component to access the component properties.
- iii) The Basic Information tab opens by default.
  - a. Select an Invocation type from the drop-down menu to confirm the running mode of the reader component. The supported invocation type is '**Real-Time**'.
  - b. Deployment Type: It displays the deployment type for the component. This field comes pre-selected.



- c. Container Image Version: It displays the image version for the docker container. This field comes pre-selected.
- d. Failover Event: Select a failover event from the drop-down menu.
- e. Batch Size (min 10): Provide the maximum number of records to be processed in one execution cycle (Min limit for this field is 10).

GCS Monitor		0		(	9	Î	:	<u>+</u>	Ť
Basic Information	Meta Inform	nation							
Invocation Type*	<b>~</b>	Deployment Type docker	Container Image Version 6.5.0-R1						
Failover Event	*	Batch Size(min 10)							

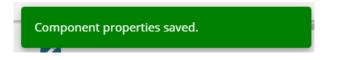
- iv) Open the Meta Information tab and configure it.
  - 1. Bucket Name: Enter the source bucket name
  - 2. Directory Path: Fill monitor folder path using forward-slash (/). E.g., monitor/
  - 3. Copy Directory Path: Fill in the copy folder name where you want to copy the uploaded file. E.g., **monitor\_copy/**
  - 4. Choose file: Upload a Service Account Key(s) file.
  - 5. File Name: After the Service Account Key file is uploaded, the file name gets auto-generated based on the uploaded file.
  - 6. Copy Bucket Name: Fill in the destination bucket name where you need to copy.

GCS Monitor	0		0 1	+	Ţ
Basic Information	Meta Information				
Bucket Name*	Directory Path*	Copy Directory Path*	Choose File		
File Name*	Copy Bucket Name*				

 v) Click the 'Save Component in Storage' icon to save all the Meta Information for the GCS Monitor component (A message appears to notify completion of the action).



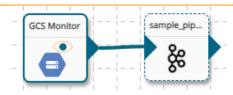
vi) The GCS Monitor component gets configured, and a notification message appears to confirm the action.



vii) The users need to create an event and connect it with the GCS Monitor component.

(Monitor sends filename to the event when any file gets uploaded on monitor folder.)

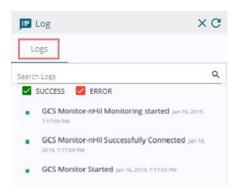




viii) Click the Update Pipeline icon (A message appears to notify completion of the action).



ix) Open the Log toggle panel to see the process of the GCS Monitor.



x) The GCS Monitor component can be used in a pipeline workflow.

Note: The GCS monitor can work with GCS Reader. BDB Pipeline supports CSV and JSON files at present.

#### 9.5.4. AWS SNS Monitor

The user needs to create an SNS topic as a precondition for SNS Monitor in BDB Data Pipeline.

#### 9.5.4.1. Steps to Configure SNS and SQS for SNS Monitor

- i) Log in to an AWS account.
- Navigate to the SNS console using the below-given link: https://us-west-2.console.aws.amazon.com/sns/v2/home?region=us-west-2#/home
- iii) Select the 'Create topic' option.

SNS dashboard Topics	4	SNS dashboard		
Applications Subscriptions		Building a mobile app? Try AWS Mobile Hub.		
fext messaging (SMS)		Common actions	Resources	
		Create topic Create a communication channel to send messages and subscribe to notifications	Topic Subscriptions	lowing Amazon SNS resources in the us-west-2 region 1
		Create platform application Create a platform application for mobile devices	Applications Endpoints	0 0
		O Create subscription Subscribe an endpoint to a topic to receive messages published to that topic	More info	
		Publish message Publish a message to a topic or as a direct publish to a platform endpoint	Getting started Documentation API reference	
		Publish text message (SMS)     Publish a text message to a phone number	Forums Service health	

iv) The 'Create new topic' window opens.



- a. Provide the following information:
  - i. Topic Name: Enter a name for the topic
  - ii. Display Name: Enter the topic name (Required for topics with SMS subscription)
  - iii. Click the 'Create topic' option.

lding a mobile app	? Try AWS Mobile Hub.	
ame will be used t	o create a permanent unique identifier called an Amazon Resource Name (ARN).	
Topic name	Enter topic name	0
Display name	Enter topic display name. Required for topics with SMS subscriptions.	0

- v) Assign permission after the topic gets created.
- vi) Navigate to the '**Topics**' page.
- vii) Click the 'Actions' drop-down menu.
- viii) Select the 'Edit topic policy' option from the menu.

	Pu	blish to topic Cre	ate new topic	Actions -	C (
s iing (SMS)	Filte	r		Edit topic encryption configuration Edit topic display name	
		Name	ARN	Subscribe to topic	
		24jansns	am:aws:		
		monitors3	am:aws:		
		s3Monitor	amaws	Edit topic delivery policy Delivery status	
		s3testdata	arn:aws:		
		test_11	amaws	sns:us-west-2:687699484674.test_11	
	8	testsns	am:aws:	sns.us-west-2:687699484674:testsns	

- ix) The 'Basic view' tab opens.
- x) Update the policy from 'Only me' to 'Everyone' using the radio button.
- xi) Click the '**Update policy**' option.

		o this topic		
Only me (topic owne	r)			
Everyone Only these AWS use	15			
o only mare i are	Comm	a-separated list of AWS	account IDs.	
Allow these users to s	ubscribe to this to	pic		
Only me (topic owne	n			
Everyone				
Only these AWS use	Comm	a-separated list of AWS	account IDs.	
	Comm		account IDs. ple.com* or "http://example	com/**
Only users with endp	Comm			.com/**
Only these AWS use Only users with endp Using these of	points that match	examples. **@exan	pie.com" or "http://example	



#### 9.5.4.2. Create the SQS Queue

The steps to create an SQS Queue are described below:

- i) Log in to an AWS account.
- ii) Navigate to the SQS console. <u>https://console.aws.amazon.com/sqs/home?region=us-west-2#queue-browser:selected=https://sqs.us-west-2.amazonaws.com/687699484674/s3Monitor;prefix=</u>
- iii) Click the 'Create New Queue' option.

ġ	aws se	rvices 🗸	Resource Groups	•	*				₽	anand @ 6876-9948-4674	▼ Orego	ı <b>∗</b> £	Support	•
Creat	te New Queue	Queue Act	tions v										С	٥
Filter	by Prefix: Q, Ent	ter Text						×			< < 1	o 1 of 1 it	ems >	ж
	Name			*	Queue Type	٣	Content-Based Deduplication -	Messages Av	ailable	Messages in Flight ~	Created			~
	24jansqs				Standard		N/A	1		0	2019-01-24	15:21:35	GMT+0	5:30
	monitors3				Standard		N/A	3		0	2019-01-23	10:46:07	GMT+0	5:30
	s3Monitor				Standard		N/A	0		0	2019-01-10	1:42:13	GMT+05	:30
	s3datatest				Standard		N/A.	0		0	2019-01-23	2:10:53	GMT+0	5:30
	testsqs				Standard		N/A	1		0	2019-01-23	2:44:38	GMT+0	5:30

- iv) The 'Create New Queue' page opens.
  - a. Provide a 'Queue Name' in the given space.
  - b. Provide the '**Region**'.
  - c. Select the type of queue from the provided options: **Standard Queue** or **FIFO Queue.**

aws Services -	Resource Groups 🗸	\$		۵	anand @ 6876-9948-4674 👻	Oregon 👻	Support 👻
Create New Queue							
		What do you want to	o name your queue?				
Queue Name 🚯							
Type the queue name.							
Region 🚯 US West (Oregon)							
	Standard Queue				FIFO Queue		
Unlimited Throughput: Standard second (TPS) per API action.	d queues support a nearly unlim	ited number of transactions per	High Throughput: FIFO queues su delete operations per second). Whe can support up to 3,000 messages p	in you b	atch 10 messages per operation (r	naximum), FIF	O queues
At-Least-Once Delivery: A mess of a message is delivered.	sage is delivered at least once, I	out occasionally more than one copy	First-In-First-out Delivery: The ord	der in w	hich messages are sent and receiv	ed is strictly pr	eserved.
Best-Effort Ordering: Occasiona	ally, messages might be delivere	d in an order different from which	Exactly-Once Processing: A mess	sage is	delivered once and remains availab	ole until a consi	umer

- v) The users need to subscribe to the queue to an SNS topic.
- vi) Navigate to the 'Queue Actions' drop-down menu.
- vii) Select the 'Subscribe Queue to SNS topic' option.

					14 4	
ilter by Prefix: Q, Enter Text.			×		≪ ≪ 1 to 1 of 1 items	• > >1
Name	- Queue Type	<ul> <li>Content-Based I</li> </ul>	Deduplication - Message	s Available - Messages in Fli	light - Created	
s3Monitor	Standard	N/A	0	0	2019-01-10 11:42:13 GM	IT+05:30



Note: Once the subscription is done with our SNS topic, the users need to create the event in S3 so that the pipeline can receive the uploaded message that comes to the S3 bucket.

#### 9.5.4.3. Creating an S3 Event

- i) Log in to an AWS account.
- ii) Create the S3 bucket.
- iii) Select the '**Properties**' tab.

Overview	Properties	Permissions	Management	
Versioning	Server	access logging	Static website hosting	
Keep multiple versions of an object in the same bucket.		s log records that provide bout access requests.	Host a static website, which does not require server-side technologies.	
Learn more		Learn mote	Learn more	
Disabled	Disabled		Disablod	
Object-level logging	Defa	ult encryption		
Record object-level API activity using the CloudTrail data events feature (additional cost).		lly encrypt objects when ed in Amazon S3		

#### iv) Select the 'Events' option under properties.

Prevent objects from being deleted. Learn more	Use tags to track your cost against projects or other criteria.	Enable fast, easy and secure transfers of files to and from your bucket.
Disabled	O Tags	Suspended
Events Receive notifications when specific events occur in your bucket. Learn more	Requester pays The requester (instead of the bucket owner) will pay for requests and data transfer Learn more	
1 Active notifications	Disabled	

- v) Create a new Event for the respective S3 bucket by providing the following information:
  - a. Provide 'Name' for the Event.
  - b. Select an option using the checkmark in the given box from the provided list of **Events.**

Name	
input	
Events ()	
✓ PUT	Permanently deleted
POST	Delete marker created
COPY	All object delete events
Multipart upload completed	Restore from Glacier initiated
All object create events	Restore from Glacier completed
Object in RRS lost	



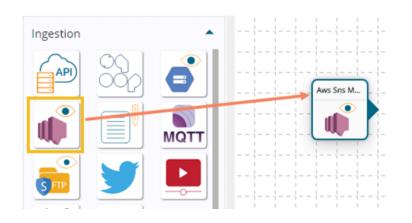
- c. Select the **SNS topic** as an option from the '**Send to'** drop-down menu.
- d. Select the 'S3 Monitor' from the SNS drop-down menu.
- e. Click the 'Save' option.

e.g. images/	
Suffix 🚯	
e.gjpg	
Send to 🜖	
SNS Topic	~
SNS	
s3Monitor	~

After the Event gets created, the users can access the complete setup to use the Data Pipeline SNS Monitor to monitor the S3 location.

#### 9.5.4.4. Using the AWS SNS Monitor Component in the Data Pipeline

i) Drag and drop the AWS SNS Monitor component to the Workflow Editor.



- ii) Click on the component to get the component properties tabs below.
- iii) The Basic Information tab opens by default.
  - a. Select an Invocation type from the drop-down menu to confirm the running mode of the reader component. The supported invocation type is '**Real-Time**'.
  - b. Deployment Type: It displays the deployment type for the component. This field comes pre-selected.
  - c. Container Image Version: It displays the image version for the docker container. This field comes pre-selected.
  - d. Failover Event: Select a failover event from the drop-down menu.
  - e. Batch Size (min 10): Provide the maximum number of records to be processed in one execution cycle (Min limit for this field is 10).

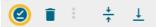


Sns Monitor		0		۷	Î	:	<u>+</u>
asic Information M	eta Inform	nation					
Invocation Type*	•	Deployment Type docker	Container Image Version 6.5.0-R1				
Failover Event	-	Batch Size(min 10)					

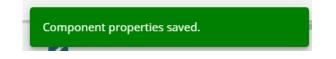
- iv) Open the Meta Information tab and provide all the required details:
  - a. Access Key
  - b. Secret Key
  - c. Region
  - d. SQS URL

Aws Sns Monitor	0		<u>+</u> <u>↓</u>
Basic Information	Meta Information		
Access Key*	Secret Key*	Region*	*
Sqs URL*			

v) Click the 'Save Component in Storage' icon to save the configuration details.



vi) The AWS SNS Monitor component gets configured and a notification message appears to confirm.

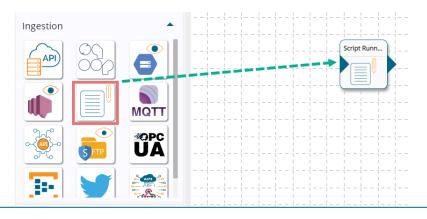


vii) The AWS SNS Monitor component can be used in a pipeline workflow now.

#### 9.5.5. Script Runner

This component can be used for connecting it to a remote server/machine and running script files present there based on some events.

i) Drag and drop Script Runner Component to the Workflow Editor.





- ii) Open the dragged Script Runner component to open the component properties tabs.
- iii) The Basic Information tab opens by default.
  - a. Select an Invocation type from the drop-down menu to confirm the running mode of the reader component. The supported invocation types are '**Real-Time**' and '**Batch**'.
  - b. Deployment Type: It displays the deployment type for the component. This field comes preselected.
  - c. Container Image Version: It displays the image version for the docker container. This field comes pre-selected.
  - d. Failover Event: Select a failover event from the drop-down menu.
  - e. Batch Size (min 10): Provide the maximum number of records to be processed in one execution cycle (Min limit for this field is 10).

Script Runner		0		٢	Î	:	<u>+</u>	Ť
Basic Information	Meta Inform	nation						
		Deployment Type	Container Image Version					
Invocation Type*		docker	6.5.0-R1					
Failover Event	•	Batch Size(min 10)						

- iv) Open the Meta Information tab and configure the required information.
  - a. **Host:** Host IP of the remote server/machine
  - b. Username: Username of the remote server/machine.
  - c. Port: Provide machine Port number.
  - d. Authentication: Select an authentication option from the drop-down menu.
    - 1. Password: By selecting this option the user needs to pass the password.
    - 2. PEM/PPK File: By selecting this option the user needs to pass the authentication file to connect to the server.
  - e. **Script type**: Choose the type of script file that You want to run out of SSH/ PERL/command options.
  - f. File path: Path of the file that is stored at the remote server.
  - g. **File Name:** The script file that you want to execute.
  - h. **Event File Location:** this is the location of the file sent through the file monitor (Non-mandatory).

Script Runner	0				<b>i</b> :	+	<u> </u>
Basic Information	Meta Information						
User Name*	host*	Port 23	Authentica	tion	<b>•</b>		
File Name*	File Path*	Script Type*	▼	Event F	ile Locatior	1	

Note: The displayed fields may vary based on the selected Authentication option.

Component Properties when the Authentication option is Password.



Script Runner			Ø 🖬 :	<u>+</u> <u>↓</u>
Basic Information	ata Information			
User Name*	host*	Port 23	Authentication Password	
Password*	File Name*	File Path*	Script Type*	•
Event File Location				

Component Properties when the Authentication option is PEM/PPK File.

Script Runner	0		Ø 🖬 :	<u>+</u> <u>↓</u>
Basic Information	Meta Information			
User Name*	host*	Port 23	Authentication PEM / PPK File	
PEM File Name	Choose File	File Name*	File Path*	

#### i. Input Arguments

- 1. Manual Arguments (Optional): These are the arguments to the parameter of the script that the user can provide manually.
- 2. Event Arguments (Optional): These are the arguments to the parameter coming from the previous event/Kafka topic.

put Arguments			
Кеу	•	Value	×
Add New Column			

Note: The user need not do anything if the arguments to be passed are coming from the previous event. By selecting Even Arguments, the value is set to true by default.

v) Click the 'Save Component in Storage' icon (A notification message appears to confirm the action completion).



vi) The Script Runner component gets configured, and the notification message appears to inform the same.

Note: The component can connect to the remote machine using the details provided. It will pick the file from the location in that machine using the file name and file path respectively and finally execute the script after passing arguments (if any).

• Limitations

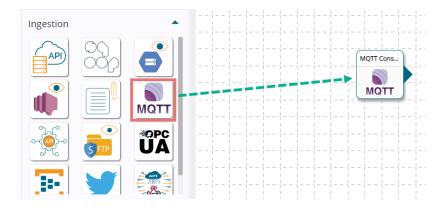


- a. It accepts only lists as input i.e. the in-event data should be a list.
- b. It sends data on the out-event only when there is a print statement as output in the script if not there will be no data on the out event.
- c. The data produced from the script is of a list type.

#### 9.5.6. MQTT Consumer

The MQTT consumer is instrumental in consuming data from IoT devices. It consumes data from provided MQTT broker and ingests the data in our data pipeline for further processing.

i) Drag and drop the MQTT Consumer to the Workflow Editor.



- ii) Click the dragged MQTT Consumer component to get the component properties tabs.
- iii) The Basic Information tab opens by default.
  - a. Select the invocation type (At present only the 'Real-Time' option is provided)
  - b. Deployment Type: It comes preselected based on the component.
  - c. Container Image Version: It comes preselected based on the component.
  - d. Failover Event: Select a failover event from the drop-down menu.
  - e. Batch Size (min 10): Provide the maximum number of records to be processed in one execution cycle (Min limit for this field is 10).

MQTT Consumer		0		Ø	:	<u>+</u>	Ť
Basic Information	Meta Inform	nation					
Invocation Type*	•	Deployment Type docker	Container Image Version 6.5.0-R1				
Failover Event	-	Batch Size(min 10)					

- iv) Click on the Meta Information tab to open the properties fields.
  - a. Host: Broker IP or URL
  - b. Username: If authentication is required then give username
  - c. Broker Port Number
  - d. Authentication: Select an authentication option using the drop-down list.
    - i. Password: Provide a password to authenticate the MQTT component.
    - ii. PEM/PPK File: Choose a file to authenticate the MQTT component
  - e. Quality of Service: The Quality of Service (QOS) level is an agreement between the sender of a message and the receiver of a message that defines the guarantee of delivery for a specific message.
  - f. MQTT Topic: Provide MQTT topic name where the user is producing data.



Click the 'Save Component in Storage' icon (A notification message appears to confirm the v) action).

MQTT Consumer	0		ة (٢)	:	<del>+</del> +	Ŧ
Basic Information	Meta Information					
Host IP Address*	Port* 1883	Username*	Authentication	*		
Quality Of Service*	MQTT Topic*					

Note: The displayed fields may vary based on the selected Authentication option.

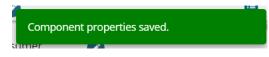
🕑 🖀 🗄 土 千 MQTT Consumer 0 Basic Information Meta Informatio Username\* mqtt-host username 1900 Password -Passw Quality Of Service\* MQTT Topic\* ••••• 2 Demo

Configuration Fields when the Authentication option is Password

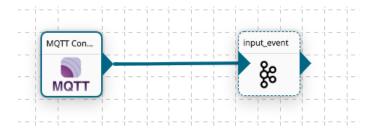
Configuration Fields when the Authentication option is PEM/PPK File

IQTT Consumer 🕜			<b>9</b> i	<u>⊥</u> <u>⊺</u>
Basic Information Meta	Information			
Host IP Address* mqtt-host	Username* username	Port*	Authentication PEM / PPK File -	
File Name	Choose File	Quality Of Service*	MQTT Topic* Demo	

The MQTT gets configured, and the notification message appears to inform the same. vi)



Note: The MQTT Consumer should be connected to an input event to pass the data and create a Pipeline workflow.

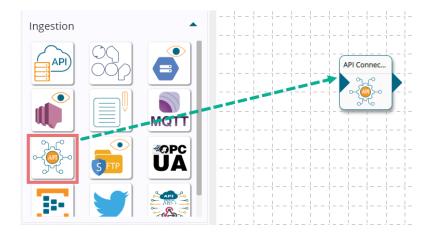


#### 9.5.7. **API Connector**

The API connector helps to ingest data from the Google Analytics and Service Now API connectors.



#### i) Drag and Drop the API Connector component to the Workflow Editor.



- ii) Click on the dragged ingestion component to get the component properties tabs.
- iii) Configure the **Basic Information** tab.
  - a. Select an Invocation type from the drop-down menu to confirm the running mode of the reader component. Select one option out of '**Real-Time**' and '**Batch**'.
  - b. Deployment Type: It displays the deployment type for the component. This field comes preselected.
  - c. Container Image Version: It displays the image version for the docker container. This field comes pre-selected.
  - d. Failover Event: Select a failover event from the drop-down menu.
  - e. Batch Size (min 10): Provide the maximum number of records to be processed in one execution cycle (Min limit for this field is 10).

Connectors	0			0	Î	:	<u> </u>
asic Information	eta Information						
Invocation Type*	■ Deployment Type	Container Image Ve 6.0.0-R3	rsion				
Failover Event	<ul> <li>Batch Size(min</li> </ul>	10)					

- iv) Click on the Meta Information tab to open the component properties fields.
  - a) Customer Type: Select one option from the drop-down menu. The given choices are:
    - i. Service Now
    - ii. Google Analytics

Meta Information fields when the Connector Type is Service Now

- 1. Connector Type: Select the Service Now option from the drop-down menu.
- 2. Service Now DataSet: Select a Service Now Data Set from the drop-down menu.
- 3. Domain: it gets selected based on the chosen Data Set
- 4. Query: It gets reflected in the query statement based on the selected Data Set
- 5. Columns: It gets reflected based on the selected Data Set
- 6. Dataset ID: It gets reflected based on the selected Data Set



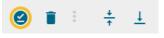
Pl Connectors				🛛 🗊 i 🕂 土
Basic Information	Meta In	formation		
Connector Type*		Service Now DataSet*		Domain*
Service Now				https://qa.bdb.ai
<sub>Query*</sub> api/now/table/pro	blem			
			11	
Columns*	a disalay yal	ue","type":"string","jsonpath":"made_sla.display_value"}]	<u> //</u>	DataSet ID* 411500592

Meta Information fields when the Connector Type is Google Analytics

- 1. Connector Type: Select the Google Analytics option from the drop-down menu.
- 2. Data Connector: Select a Google Analytics data connector from the drop-down menu.
- 3. Start Date: It gets reflected based on the chosen Data Connector.
- 4. End Date: It gets reflected based on the chosen Data Connector.
- 5. Metrics: The selected metrics get reflected based on the choice of the data connector.
- 6. Dimensions: The selected dimensions from the data connector get reflected.
- 7. Domain: The domain address appears based on the selected data connector.
- 8. View id: It gets reflected from the selected data connector.
- 9. Data Set ID: It gets reflected from the selected data connector.

Pl Connectors	2		❷ <b>î</b> : ÷ <u>∔</u>
Basic Information M	eta Information		
Connector Type* Google Analytics	Data Connector*		Start Date* 30daysAgo
			SudaysAgo
End Date*	Metrics*	Dimensions*	Domain*
today	[{"name":"ga:users","type":"	[{"name":"ga:userType","typ	https://qa.bdb.ai
View Id*	DataSet ID*		
120489073	413859855		

v) Click the 'Save Component in Storage' icon to save the component properties.



vi) A notification message appears to inform about the action.

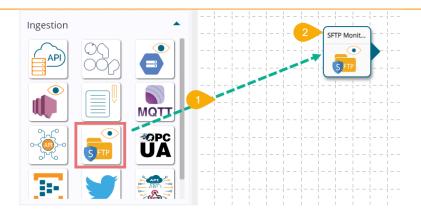


vii) The API Connector component gets configured.

#### 9.5.8. SFTP Python Monitor

- i) Drag and Drop the SFTP Python Monitor ingestion component to the Workflow Editor.
- ii) Click the dragged ingestion component to get the component properties tabs.





#### iii) Configure the **Basic Information Tab**.

- a. Select the invocation type (at present only the 'Real-Time' option is provided).
- b. Deployment Type: It comes preselected based on the component.
- c. Container Image Version: It comes preselected based on the component.
- d. Failover Event: Select a failover event from the drop-down menu.
- e. Batch Size (min 10): Provide the maximum number of records to be processed in one execution cycle (Min limit for this field is 10).

SFTP Monitor	0		۵	Î	:	<u>+</u>	Ť
Basic Information	eta Information						
Invocation Type*	Deployment Type     docker	Container Image Version 6.5.0-R1					
Failover Event	▼ Batch Size(min 10)						

- iv) Configure the Meta Information tab for the dragged SFTP Python Monitor component.
  - a. Host: Broker IP or URL
  - b. Username: If authentication is required then give username
  - c. Port: Provide the Port number
  - d. Authentication: Select an authentication option using the drop-down list.
    - i. Password: Provide a password to authenticate the MQTT component.
    - ii. PEM/PPK File: Choose a file to authenticate the MQTT component
  - e. Directory Path: Fill monitor folder path using forward-slash (/). E.g., monitor/
  - f. Copy Directory Path: Fill in the copy folder name where you want to copy the uploaded file. E.g., **monitor\_copy/**
  - g. Channel: Select a channel option from the drop-down menu (the supported channel is SFTP).

SFTP Python Monitor			0	Î	0 0 0	+	<u>+</u>
Basic Information 4 Met	a Information						
Host IP Address*	Username*	Port* 22	Authentication		-		
Directory Path*	Copy Directory Path*	Channel*	•				



Note: The SFTP Python Monitor may have Meta Information fields based on the selected Authentication option.

#### Meta Information Fields when the Authentication option is Password

SFTP Python Monitor	0		9	E	÷ 4	Ŧ
Basic Information	Meta Information					
Host IP Address*	Username*	Port* 22	Authentication Password	•		
Password*	Directory Path*	Copy Directory Path*	Channel* SFTP		<b>.</b>	

#### Meta Information Fields when the Authentication option is PEM/PPK

SFTP Python Monitor			2 🖬 : 🕂 .	Ţ
Basic Information Met	a Information			
Host IP Address*	Username*	Port* 22	Authentication PEM / PPK File	
File Name	Choose File	Directory Path*	Copy Directory Path*	_
Channel* SFTP		-		

v) Click the 'Save Component in Storage' icon to save configured details of the SFTP Python Monitor component.

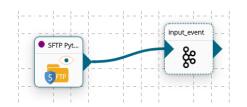


vi) A notification message appears to confirm the same.



Note:

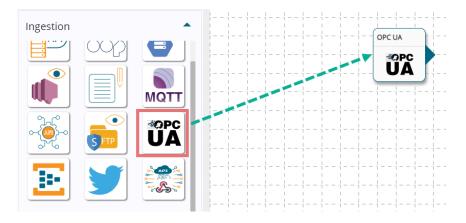
- a. The SFTP Python Monitor component monitors the file coming to the monitored path and copies the file in the Copy Path location for SFTP Reader to read.
- b. The SFTP Python Monitor component requires an Event to send output.





#### 9.5.9. OPC UA

i) Drag and Drop the **OPC UA** ingestion component to the Workflow Editor.



- ii) Click on the dragged ingestion component to get the component properties tabs.
- iii) Configure the Basic Information tab.
  - a. Select an Invocation type from the drop-down menu to confirm the running mode of the reader component. Select one option out of '**Real-Time**' and '**Batch**'.
  - b. Deployment Type: It displays the deployment type for the component. This field comes preselected.
  - c. Container Image Version: It displays the image version for the docker container. This field comes pre-selected.
  - d. Failover Event: Select a failover event from the drop-down menu.
  - e. Batch Size (min 10): Provide the maximum number of records to be processed in one execution cycle (Min limit for this field is 10).

OPC UA		2		0	Î	:	Ť	Ť
Basic Information	Meta Inform	ation						
Invocation Type*	•	Deployment Type docker	Container Image Version 6.5.0-R1					
Failover Event	•	Batch Size(min 10)						

- iv) Configure the Meta Information tab by providing the required fields.
  - a. URL (\*)- Provide URL link
  - b. Message Security Mode- Select a message security mode from the drop-down menu (The supported options are 'Sign' and 'SignAndEncrypt')
  - c. Security Policy- Select a policy using the drop-down menu.
  - d. Certificate File Name- This name gets reflected based on the Choose File option provided for the Certificate file.
  - e. Choose File- Browse a certificate file by using this option.
  - f. PEM File Name: This name gets reflected based on the Choose File option provided for the PEM file.
  - g. Choose File: Browse a PEM file by using this option.
  - h. Source Node: Provide the source node.
  - i. Event Note: Provide the event node.



OPC UA	0		11111		0	•	+	<u>1</u>
Basic Information	Meta Information							
URL*	Message Security Mode	•	Security Policy	•	Certifica	ate File Na	ime	
Choose File	PEM File Name		Choose File		Source	Node*		
Event Node*								

v) Click the 'Save Component in Storage' icon.



vi) A notification message appears to inform about the action.

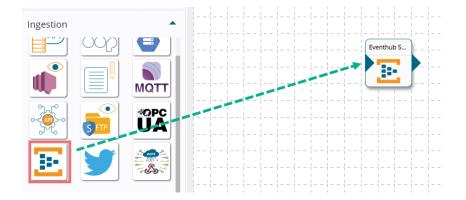


vii) The OPC UA ingestion component gets configured, and a notification message appears to inform the same.

#### 9.5.10. Eventhub Subscriber

This Component consumes the event ingestion service (event hub) provided by azure. The user can directly subscribe to the azure event hub and ingest the events into a data pipeline. This comes under the ingestion component group.

i) Drag and drop the Eventhub Subscriber component to the Workflow Editor.



- ii) Click the dragged Eventhub component.
- iii) Click on the dragged ingestion component to get the component properties tabs.
- iv) Configure the Basic Information tab.
  - a. Select an Invocation type from the drop-down menu to confirm the running mode of the component. The supported invocation option is '**Real-Time**'.
  - b. Deployment Type: It displays the deployment type for the component. This field comes preselected.
  - c. Container Image Version: It displays the image version for the docker container. This field comes pre-selected.
  - d. Failover Event: Select a failover event from the drop-down menu.



e. Batch Size (min 10): Provide the maximum number of records to be processed in one execution cycle (Min limit for this field is 10).

Eventhub Subscriber	0		٢	Î	:	Ŧ	Ť
Basic Information	eta Information						
Invocation Type*	Deployment Type     docker	Container Image Version 6.5.0-R1					
Failover Event	▼ Batch Size(min 1	)					

- v) Configure the **Meta Information tab** by providing the required fields.
- vi) The user can configure the Eventhub Subscriber meta-Information by reading the data using either **Connection String** or **Principal Secret**.

Eventhub Subscriber		9	: -	<u>+</u> _⊥
Basic Information Meta Infor	mation			
Connection String	Consumer Group*	EventHub Na	ime*	
Principal Secret	Enqueued Time(DD/MM/YYYY, HH:MM:SS) UTC	Subscriber N	amespa.	

The Meta Information fields with Connection String as a Reading option.

- a. **Reading Using**: Select the '**Connection Sting**' to connect to the event hub service (this can be found in Azure platform access manages].
- b. **Consumer Group**: A consumer group is a view (state, position, or offset) of an entire event hub.
- c. **Eventhub name**: Name of the subscribed event hub.
- d. **Checkpoint location:** it is an HDFS location URL ("hdfs://<ip>/user/Cloudera/Test"). It can be used when a component fails, and you want to read from the exact event where you left.
- e. Enqueued time: This is to be used if the user wants to read the events from a specific time.
- f. Subscriber Namespace: Subscriber Namespace of the subscribed Eventhub.

sic Information Meta Info	prmation	
Read Using* Connection String <	Connection String*	Consumer Group*
EventHub Name*	Checkpoint Location	Enqueued Time(DD/MM/YYYY, HH:MM:SS) UTC

The Meta Information fields with Principal Secret as a Reading option.

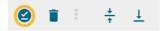
- a. Reading Using: Select the 'Principal Secret' to connect to the event hub service.
- b. **Client ID**: Provide the Client ID.
- c. **Configure Tenant ID**: Provide the Tenant ID.
- d. **Client Secret**: Provide the Client Secret ID.
- e. **Consumer Group**: A consumer group is a view (state, position, or offset) of an entire event hub.



- f. Eventhub name: Name of the subscribed event hub.
- g. Checkpoint location: it is an HDFS location URL ("hdfs://<ip>/user/Cloudera/Test"). It can be used when a component fails, and you want to read from the exact event where you left.
- h. Enqueued time: This is to be used if the user wants to read the events from a specific time.
- i. Subscriber Namespace: Subscriber Namespace of the subscribed Eventhub.

thub Subscriber		2 🔋 : 🕂
ic Information Meta Information		
Read Using* Principal Secret ▼ Client ID*	Tenant ID*	
Client Secret*		
Client Secret* Consumer Group*	EventHub Name*	Checkpoint Location

vii) Click the 'Save Component in Storage' icon.



viii) A notification message appears to inform about the action.

Component properties saved.

ix) The Eventhub Subscriber ingestion component gets configured, and a notification message appears to inform the same.

#### 9.5.11. Twitter Scrapper

Twitter Scrapper helps in scrapping Twitter data. Twitter Scrapper provides two types of scrapping:

- 1. Real-Time
- 2. History Data

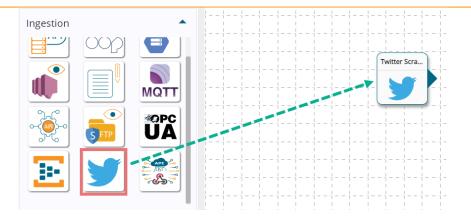
Real-Time – By using this option, the user can fetch the live Tweets by using Twitter API along with specific search filters.

History Data – By using this option, the user can provide or fetch Twitter data for the past 7 days (provided limit is up to 1000 tweets). It also provides the users with a facility to fetch data according to specific

tags.

i) Drag them and drop Twitter Scrapper to the Workflow Editor.





- ii) Click the dragged Twitter Scrapper component to get the component properties tabs.
- iii) The Basic Information tab opens by default.
  - a. Select the invocation type (at present only the 'Real-Time' option is provided)
  - b. Deployment Type: It comes preselected based on the component.
  - c. Container Image Version: It comes preselected based on the component.
  - d. Failover Event: Select a failover event from the drop-down menu.
  - e. Batch Size (min 10): Provide the maximum number of records to be processed in one execution cycle (Min limit for this field is 10).

Twitter Scrapper		٥		Ø	Î	 <u>+</u>	Ť
Basic Information	Meta Inform	nation					
Invocation Type*	•	Deployment Type docker	Container Image Version 6.5.0-R1a				
Failover Event	•	Batch Size(min 10)					

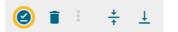
- iv) Click on the Meta Information tab to open the configuration fields for the selected columns.
  - a. Consumer API Key: This is going to be unique depending upon the user's account. It is essential in both the cases of scrapping Real-time and History Data.
  - b. Consumer API Secret Key: This is going to be unique depending upon the user's account. It is essential in both the cases of scrapping Real-time and History Data.
  - c. Filter Text: This field is used to specify separate tags that the user wants to fetch from Twitter. It provides the choice to the user for fetching multiple tags data at a time, to accomplish this user needs to provide the tags name with a comma separator. For instance, #India, #Cricket
  - d. Twitter Data Type: This field provides a dropdown menu from where the consumer can select the scrapping strategy(real-time/history).
  - e. Access Token: This field is consumable only when the user wants to fetch real-time data from a scrapper.
  - f. Access Token Secret: This field is consumable only when the user wants to fetch real-time data from a scrapper.

Note: The Access Token and the Access Token Secret (key) fields appear only when the selected Twitter Data Type is 'real-time'.



Twitter Scrapper			9 🖬 :	<u>+</u>
Basic Information Meta	Information			
Consumer API Key *	Consumer API SecretKey *	Filter Text *	Twitter Data Type * real-time	
Access Token *	Access Token Secret *			

v) Save the Twitter Scrapper component.



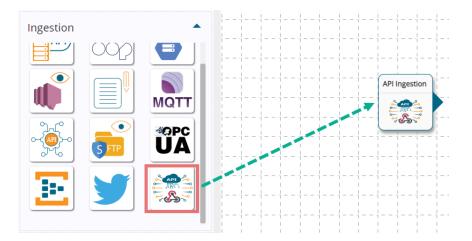
vi) The Twitter Scrapper component properties get saved and a notification message appears to inform the same.



Note: The user should know the Consumer API Key, Consumer Secret Key, Access Token, Access Token Secret from a developer account of Twitter.

#### 9.5.12. API Ingestion

- The API Ingestion component is provided in the Component to ingest external API content.
- i) Drag them and drop API Ingestion to the Workflow Editor.



- ii) Click the dragged API Ingestion component to get the component properties tabs.
- iii) The Basic Information tab opens by default.
  - a. Select the invocation type (at present only the 'Real-Time' option is provided)
  - b. Deployment Type: It comes preselected based on the component.
  - c. Container Image Version: It comes preselected based on the component.
  - d. Failover Event: Select a failover event from the drop-down menu.
  - e. Batch Size (min 10): Provide the maximum number of records to be processed in one execution cycle (Min limit for this field is 10).



API Ingestion		2		Q	Î	Ť	Ť
Basic Information	Meta Inform	ation					
Invocation Type*	•	Deployment Type docker	Container Image Version 6.5.0-R1				
Failover Event	•	Batch Size(min 10)					

- iv) Click on the Meta Information tab to open the configuration fields for the selected columns.
  - a. The Ingestion Type field provides two options:
    - i. Webhook
    - ii. API Ingestion
  - b. Provide an Ingestion Id.
  - c. Provide an Ingestion Secret Key.
  - d. Provide Component Instant Id URL.

PI Ingestion		e	Î	:	Ŧ	Ť
Basic Information	eta Information					
Webhook	Ingestion Id* ca9951ae-da08-4a79-9c68-3ff830186fbd	6				
API Ingestion	y5/vPb0ZNy5dV5mPeaq+MqTCTP00aNDolQIXwxn8					

v) Save the API ingestion component.



vi) The API Ingestion component properties get saved and a notification message appears to inform the same.



#### 9.6. WebSocket

The WebSocket component provided in the Components Pallet of the Data pipeline contains WebSocket Listener and WebSocket Writer components to produce and consume the live streaming data using socket.

#### 9.6.1. WebSocket Listener

The WebSocket Listener component listens to and accepts the incoming WebSocket connection requests.

i) Drag the WebSocket Listener component to the Workflow Editor.



Websocket			 	 	  -	 	 Websocket	-	
<b><u>N</u> N</b>			 	 	- - - -	     	 Web Socket		
Web Socket		{	 	 	- 1	 	 	-	

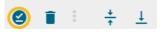
- ii) Open the WebSocket Listener component to get the Component Properties tab below.
- iii) The Basic Information tab opens by default.
  - a. Select an Invocation type from the drop-down menu to confirm the running mode of the reader component. The supported invocation type is '**Real-Time**'.
  - b. Deployment Type: It displays the deployment type for the component. This field comes preselected.
  - c. Container Image Version: It displays the image version for the docker container. This field comes pre-selected.
  - d. Failover Event: Select a failover event from the drop-down menu.
  - e. Batch Size (min 10): Provide the maximum number of records to be processed in one execution cycle (Min limit for this field is 10).

Websocket Listener			Ø	Î	 <u> </u>	Ť
Basic Information Meta II	nformation					
Invocation Type*	Deployment Type     docker	Container Image Version 6.5.0-R1				
Failover Event	▼ Batch Size(min 10)					

- iv) Configure the Meta information tab by providing the required fields.
  - a. Base URL
  - b. Port
  - c. SSL Required- Enable the
  - d. Route Path

Websocket Listener	0		0	Î	:	+	<u>+</u>
Basic Information	Meta Information						
Base URL*	Port 8080	SSL Required	Route Path				

v) Click the 'Save Component in Storage' icon.



vi) A notification message appears to inform about the action.



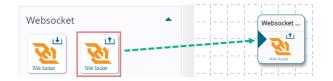


vii) The WebSocket Listener component gets configured, and a message appears to inform the same.

#### 9.6.2. WebSocket Producer

The WebSocket Producer helps the user to get the message received by the Kafka channel.

i) Drag the WebSocket Producer component to the Workflow Editor.



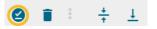
- ii) Open the WebSocket Producer component to get the Component Properties tab below.
- iii) The Basic Information tab opens by default.
  - a. Select an Invocation type from the drop-down menu to confirm the running mode of the reader component. The supported invocation type is '**Real-Time**'.
  - b. Deployment Type: It displays the deployment type for the component. This field comes preselected.
  - c. Container Image Version: It displays the image version for the docker container. This field comes pre-selected.
  - d. Failover Event: Select a failover event from the drop-down menu.
  - e. Batch Size (min 10): Provide the maximum number of records to be processed in one execution cycle (Min limit for this field is 10).

Websocket Producer			Ø	Î	:	Ť	Ť
Basic Information	leta Information						
	Deployment Type	Container Image Version					
Invocation Type*	docker	6.5.0-R1					

iv) Provide the WebSocket ID field to configure the Meta Information tab.

Websocket Producer	_0	🥝 i : 🕂 🛓
Basic Information	Meta Information	
ID		

v) Click the 'Save Component in Storage' icon (A message appears to confirm the same).



vi) A notification message appears to inform about the action.



vii) The WebSocket Producer component gets configured, and a message appears to inform the same.

#### 9.7. Scheduler

The scheduled pipelines are listed together with the scheduler details. It displays the meta-information filled in the scheduler component for the respective pipeline. The page also contains information on how many times the pipeline has been triggered and when the next time the scheduled component will get deployed.

- i) Drag and drop the Scheduler component to the Workflow Editor.
- ii) Connect it with a reader or Data Loading component (Input event).



- iii) Click on the Scheduler component to get the configuration details.
- iv) The Basic tab opens by default.
  - a. Select an Invocation type from the drop-down menu to confirm the running mode of the reader component. The supported invocation type is '**Real-Time**'.
  - b. Deployment Type: It displays the deployment type for the component. This field comes preselected.
  - c. Container Image Version: It displays the image version for the docker container. This field comes pre-selected.
  - d. Failover Event: Select a failover event from the drop-down menu.
  - e. Batch Size (min 10): Provide the maximum number of records to be processed in one execution cycle (Min limit for this field is 10).

Scheduler	O		۲	:	Ţ	Ť
Basic Information M	eta Information					
Invocation Type*	Deployment Type     none	Container Image Version 6.5.0-R1				
Failover Event	▼ Batch Size(min 10)					

- v) Open the Meta Information tab and configure the required fields.
- vi) Provide Scheduler name
- vii) Provide Payload value i(n the predefined format)
- viii) Set Scheduler Time using the Crone Generator (the displayed fields may vary based on the selected time).
- ix) Click the 'Save Component in Storage' ≤ icon to save the component configuration.



heduler	0	4 @ ■ ÷ ÷
Basic Information Meta	Information	
Scheduler Name* Sample Scheduler	2 Payload(Ex: ('key''value')) ("key": "value")	3 Scheduler Time* 0 0/2 * 1/1 * ? * Cron Generator Minutes Hourly Daily Weekly Monthly Yearly Every 2  v minute(s) on second

x) A notification message appears to confirm the action.



- xi) Update and activate the pipeline (notification messages appear to confirm the same).
- xii) Open the Scheduler tab from the landing page screen.
- xiii) The scheduled pipeline gets listed displaying the triggered time and Next Run Time details.

Scheduler Name	Colorado Java Tirana	Next Due Time	Disalization	Comple Cohedular	
Scheduler Name	Scheduler Time	Next Run Time	Pipeline Name	Sample Scheduler	
sched	0 0/5 * 1/1 * ? *	16-11-2020 16:50:00	Email_demo	Description 个	Triggered Time
starter	0 0/1 * 1/1 * ? *	30-11-2020 13:01:00	car_trial	Successfully Executed	Apr 1, 2021, 7:30:00 PM
				Successfully Executed	Apr 1, 2021, 8:30:00 PM
New_sched	0 0 0/1 1/1 * ? *	05-03-2021 16:30:00	Covid_OtherAPIs	Successfully Executed	Apr 1, 2021, 9:30:00 PM
covid_state	0 5 0/2 1/1 * ? *	05-03-2021 17:35:00	Covid_OtherAPIs	Successfully Executed	Apr 1, 2021, 10:30:00 PM
				Successfully Executed	Apr 1, 2021, 11:30:00 PM
newSched	090/21/1*?*	05-03-2021 17:39:00	Covid_OtherAPIs	Successfully Executed	Apr 2, 2021, 12:30:00 AM
new	0 11 0/2 1/1 * ? *	05-03-2021 17:41:00	Covid_OtherAPIs	Successfully Executed	Apr 2, 2021, 1:30:00 AM
				Successfully Executed	Apr 2, 2021, 2:30:00 AM
CovidRawdata	0 0 0/1 1/1 * ? *	02-04-2021 19:30:00	CovidIndia_RawData	Successfully Executed	Apr 2, 2021, 3:30:00 AM
Sample Scheduler	0 0/2 * 1/1 * ? *	None	sample_pipeline	1 - 10 of 24	1< <

Note: The scheduler supports only reader components at present.

## **10.** Settings

The user gets settings information about various pipeline functionalities. The Settings page provides information about Kafka Configuration, Data Prep Scripts, Data Science Models, and Logger. The Kafka Configuration details open by default while selecting the '**Settings**' page.



<b>≣</b> Pi	peline Settings		
<b>f</b>	Kafka Configuration	Kafka Configuration	C 🖪
:=	Dataprep Scripts		
+	Data Science Models	Basic Information	~
=:	Logger	Configuration Details*	~
×	Default Configuration		
0			
0			
٢			

#### **10.1.** Kafka Configuration

The Kafka Configuration page displays editable Basic Information and Configuration Details for Kafka.

Basic Information contains Configuration Name and Configuration Type. The basic information details can be refreshed and saved by using the icons given for the same.

Kafka Co	onfiguration		C 🖬
	Basic Information	^	
	Configuration Name * Kafka		
	Configuration Type * Kafka		
	Configuration Details*	~	

The Configuration Details tab for Kafka displays the following information:

- a. Bootstrap Servers
- b. Group ID
- c. Auto Commit
- d. Auto Commit Interval(ms)
- e. Session Timeout(ms)
- f. Kafka Request Required Acknowledgment
- g. Key Serializer
- h. Key Deserializer
- i. Value Serializer
- j. Value Deserializer
- k. Max Block(ms)
- I. Max Inflight request per connection

The user can refresh the configuration and save the configuration by using the provided icons for the same.



onfiguration	
Configuration Details*	^
Bootstrap Servers *	Group ID *
kafka-0.kafka-headless.kafka.svc.cluster.loca	test-consumer-group
Auto Commit	Auto Commit Interval(ms)
true	100
Session Timeout(ms)	Kafka Request Required Acknowledgment
30000	all
Key Serializer *	Key Deserializer *
org.apache.kafka.common.serialization.Stri	org.apache.kafka.common.serialization.Stri
Value Serializer *	Value Deserializer *
org.apache.kafka.common.serialization.Stri	org.apache.kafka.common.serialization.Stri
Max Block(ms)	Max inflight request per connection
33554432	2147483647

#### 10.2. Dataprep Scripts

The user can see a list containing all the exported Dataprep Scripts from the Data Preparation module under the Dataprep Scripts tab provided on the Settings page. The user can also view the Transformation Details provided with the sequence of Steps, Action Type, and Column Names for the selected Dataprep Script.

Image: Script Sumple Scripts       Script Name       Created Date       Actions       East453       Image: Script Name       East453       Image: Script Name       East453       Image: Script Name       East453       Image: Script Name       Image: Script Name       Script Name       East453       Image: Script Name       Image: Script Name       Image: Script Name       Image: Script Name       Script Name       East453       Image: Script Name       Image: Script Name       Image: Script Name       Script Name       Image: Script Name       Script Name       Script Name       Image: Script Name       Script Name       Image: Script Name       Scr	<b>A</b>	Kafka Configuration	Dataprep Scripts List			Search Script	Q
Image: Section of the section of t	=	Dataprep Scripts	Script Name	Created Date	Actions	test453	
Image: Logger     Imag	+	Data Science Models	test453		i	Linked pipelines	
Default Configuration     test45     pM     Transformation Details       Image: Configuration     sub_transform2     pc 7, 2021, 1:45:14     image: Configuration     image: Configuration       Image: Configuration     sub_transform2     pc 7, 2021, 1:36:17     image: Configuration     image: Configuration       Image: Configuration     sub_transform2     pc 7, 2021, 1:36:17     image: Configuration     image: Configuration       Image: Configuration     sub_transformation_Test1     pc 7, 2021, 1:33:40     image: Configuration     image: Configuration       Image: Configuration     sub_transformation_Test1     pc 7, 2021, 1:33:40     image: Configuration     image: Configuration       Image: Configuration     demo configuration     Nov 1, 2021, 5:47:01     image: Configuration     image: Configuration	=:	Logger	test452	Dec 7, 2021, 3:44:04	i	Splitting the workflow	Ø
Image: Sub_transform2     Dec 7, 2021, 1:36:17     Image: Dec 7, 2021, 1:33:40       Image: Sub_transformation_Test     Dec 7, 2021, 1:33:40     Image: Dec 7, 2021, 1:33:40       Image: Sub_transformation_Test     Dec 7, 2021, 1:33:40     Image: Dec 7, 2021, 1:33:40       Image: Sub_transformation_Test     Dec 7, 2021, 1:33:40     Image: Dec 7, 2021, 1:33:40       Image: Sub_transformation_Test     Dec 7, 2021, 1:33:40     Image: Dec 7, 2021, 1:33:40       Image: Sub_transformation_Test     Dec 7, 2021, 1:33:40     Image: Dec 7, 2021, 1:33:40       Image: Sub_transformation_Test     Dec 7, 2021, 1:33:40     Image: Dec 7, 2021, 1:33:40       Image: Sub_transformation_Test     Dec 7, 2021, 1:33:40     Image: Dec 7, 2021, 1:33:40       Image: Sub_transformation_Test     Dec 7, 2021, 1:33:40     Image: Dec 7, 2021, 1:33:40       Image: Sub_transformation_Test     Dec 7, 2021, 1:33:40     Image: Dec 7, 2021, 1:33:40       Image: Sub_transformation_Test     Dec 7, 2021, 1:33:40     Image: Dec 7, 2021, 1:33:40       Image: Sub_transformation_Test     Dec 7, 2021, 1:33:40     Image: Dec 7, 2021, 1:33:40       Image: Sub_transformation_Test     Dec 7, 2021, 1:33:40     Image: Dec 7, 2021, 1:33:40       Image: Sub_transformation_Test     Dec 7, 2021, 1:33:40     Image: Dec 7, 2021, 1:33:40       Image: Sub_transformation_Test     Dec 7, 2021, 1:33:40     Image: Dec 7, 2021, 1:33:40       Image: Sub_transformation_Te	×	Default Configuration	test45	Dec 7, 2021, 1:45:14	ĩ	Transformation Details	
Image: Sub_Transformation_Test1         Dec 7, 2021, 1:33:40 PM         Image: Dec 7,	0		sub_transform2		Î		Names
demo connected cars NOV 1, 2021, 5:47:01 3 DELETE COLUMN	-		sub_Transformation_Test1		Î		.current_statu
	\$		demo_connected_cars		Î	3 DELETE_COLUMN	

Note: The user can delete a Dataprep Script by clicking the '**Delete Script**' **I** icon provided next to it.

#### **10.3.** Data Science Models

The exported models from Data Science Workbench gets listed on this page. The linked pipeline to the selected model is also mentioned on the right side of the page. The user can navigate to the related pipeline by clicking the '**Open Pipeline**' <sup>C</sup> icon from this page.



Kafka Configuration	Data Science Models	5		Search Script	Q
Dataprep Scripts	Model Name	Model Type	Created On	dmpymodel	
Data Science Models	dmpymodel	Python Model	Tuesday, August 24, 2021	Linked pipelines	
Logger	dmrmdoel	R Model	Thursday, September 2, 2021	PythonModel	
Default Configuration	DMRmodel	R Model	Thursday, September 2, 2021	PythonModel	Z
	Model_mongoql	Python Model	Monday, September 6, 2021		

#### **10.3.1.** Using a Deployed Model in Data Pipeline

- i) Navigate to the Pipeline Settings page.
- ii) Select the 'Data Science Models' option.
- iii) The deployed Data Science Models display in a list.
- iv) Select a Data Science Model from the list.

Kafka Configuration	2 Data Science Mode	els		Search Script	Q
Dataprep Scripts	3 Model Name	Model Type	Created On	dmpymodel	
1 Data Science Models	4 dmpymodel	Python Model	Tuesday, August 24, 2021	Linked pipelines	
Logger	dmrmdoel	R Model	Thursday, September 2, 2021	PythonModel	
Default Configuration	DMRmodel	R Model	Thursday, September 2, 2021	PythonModel	
	Model_mongoqi	Python Model	Monday, September 6, 2021		
				1	

- v) Navigate to the ML section on the Pipeline Editor page.
- vi) Drag the Python Model component to the workspace.
- vii) Click on the dragged Python Model component to display the configuration fields.
- viii) Select the 'Meta Information' tab.
- ix) Select the 'Python Model' using the Model Type drop-down menu.
- x) Select a model using the 'Model Name' drop-down menu.

Search Component	<				
System Custom					
Reader				۹ 1009	୍ବ
Writer	Python Mo				
Transformation					
ML					
R 🛃 🔍	Python Model		9		<u>+</u> - 7
	Basic Information 8 Meta Information				
	9 Model Type		_		
	Python Model 🕤 dmpymodel	(	<b></b>		

xi) The user can drag and connect the Data Science Model with other relevant components to create a pipeline workflow (as displayed in the following image).



Mongod.	PythomMod_	Python	Pythor 8			Q	100% Q	
Python Model	0				e		<u>+</u>	Ť
Basic Information	Meta Information	Configuration	Logs					
Model Type Python Model	Model Nam			Ŧ				

#### 10.4. Logger

This page displays the Logger configuration details for the UI Logs that appear on the log panel of the workflow editor.

<b>≣</b> P	ipeline Settings			
<b>A</b>	Kafka Configuration	Logger Details		8
:=	Dataprep Scripts			
+	Data Science Models	Information	^	
≡:	Logger	Custom Log custom_log		
×	Default Configuration	Developer Logs Loggerdp_dev		
0		UI Logs		
0		Loggerdp_ui 		
۵		30000		

#### **10.5.** Default Configuration

The Default Configuration displays pipeline default configuration information categorized as High, Medium, and Low level. The users can edit the information and click the 'Save Configuration' icon.

■ P	ipeline Settings		
÷	Kafka Configuration	Default Configuration	8
=	Dataprep Scripts		
+	Data Science Models	high	^
=	Logger	batch	~
B	Default Configuration	realtime	~
S			
0		medium	^
٥		batch	~
		realtime	~
		low	^
		batch	~
		realtime	~



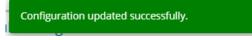
Each category displays further division as batch and real-time:

high	
batch	^
Spark	~
Non Spark	~
realtime	^
Spark	~
Non Spark	~

Each batch and real-time category further display Spark and Non-Spark categories:

tch			^
Spark			^
Driver			
Core*	Core Limit * 4096	Memory * 2048	
Executor			
Core *	Instances *	Memory *	
1	2	4096	

A message appears to confirm the updates.



The configuration gets saved.

## **11.** Third-Party Data Ingestion using Data Pipeline

The Python Script component in Data Pipeline can be used in a very flexible way as a reader, ingestion, transform component, and data writer component.

All the user needs to do is to write a custom script according to the use case.

www.bdb.ai



Below is an example of listening data from an MQTT topic and calling a machine learning model API. this API return data after applying the model to it.

#### 11.1. Data Ingestion from an MQTT Consumer

#### Sample Code:

from threading import Thread import paho.mqtt.client as mqtt import os,json,requests from confluent\_kafka import Producer

def get\_out\_event():

```
componentInsId = os.getenv('COMPONENT_INST_ID')
url = str(os.getenv('MDS_BASE_URL')) + "/api/v1/componentInst/" + str(componentInsId)
uuid = os.getenv('UUID')
if uuid is not None:
    url = url + "?uuid=" + str(uuid)
resp = requests.get(url)
component_json = json.loads(resp.content)
data = component_json['data']
meta_data = data['metaData']
out_event = meta_data['outEvent'][0]
return out_event
```

def kafka\_config():

```
url = str(os.getenv('MDS_BASE_URL')) + "/api/v1/configuration/kafka"
resp = requests.get(url)
component_json = json.loads(resp.content)
config = component_json['data']['confDetails']['bootstrap.servers']
return config
```

config = kafka\_config()
# config = "192.168.1.32:9092"
out\_event = get\_out\_event()
# out\_event = "adnoc\_wesocket\_test"
p = Producer{{'bootstrap.servers': config}}

```
def sent_mqtt_data(broker_address,username, password,topic):
    client = mqtt.Client("P1") # create new instance
    #client_client_id="lens_2q4a SVgeMiSkFw6fpTUbElx7G0B"
    client.username_pw_set(username, pass word)
    client.connect(broker_address, port=1883,keepalive=120) # connect to broker
    def on_message(client, userdata,message):
        p.produce(out_event,message.payload.decode("utf-8"))
    print("connected")
    client.subscribe(topic=topic,qos=0)
    client.on_message = on_message
```

client.loop\_forever() # return client

Corresponding File:



### 11.2. Configuring a Data Pipeline According to the Script

i) Navigate to the Transformation section of the component pallet.



#### ii) Select and drag the Python Script component to the canvas (Workflow Editor).

<ul> <li>Pipeline</li> </ul>	Editor > Data produ	ucer for toyota	2	
Search Cor	nponent	<		
System	Custom			Data_produ
Reader	-			ke ke
Writer	-			
Transformat				

- iii) Configure the Python Script component.
- iv) The **Basic Information** tab opens by default.
- v) Select the '**Real-Time**' as invocation type so it can keep listening to the MQTT topic in real-time.

Python Script	0		
Basic Information	Meta Information	Configuration	
Invocation Type* Real-Time		■ Deployment Type docker	Container Image Version 4.6.0-rel01

- vi) Open the Meta Information tab and configure as displayed in the below image:
  - 1. Component Name: Provide a name for the component.
  - 2. Script: Insert the custom python script in this space.
  - 3. Start Function: Activate the function to run the Python script.
  - 4. In Event Data Type: Select a data frame or list.
  - 5. External Libraries: Provide names of the libraries separated by a comma
  - 6. Input Date: Fill in the values of the input parameters of the start function.
  - 7. Click the 'Save' option for the component.



Python Script		000000	7 ❷∎ : ÷ ±
Basic Information Meta Information	Configuration		
Component Name* MQTT-Water	2 Script* import paho.mqtt.client as mo import os,json,requests from confluent_kafka import R	tt roducer	
Start Function* sent_mqtt_data	• 4 In Event DataType for broker_	address	perated by comma (,)]
Input Data			
broker_address	Value ×		1
Key username	Value		
Key password	Value X		
Key topic	Value		
Key topic	Value X		

Note : List of already installed libraries are as given below:

```
certifi==2019.6.16
chardet==3.0.4
confluent-kafka==1.1.0
idna==2.8
numpy==1.17.1
pandas==0.25.1
PyMySQL==0.9.3
python-dateutil==2.8.0
pytz==2019.2
requests==2.22.0
six==1.12.0
urllib3==1.25.3
```

# **11.2.1.** Making a POST request along with the data Sample code:

```
# In[]: arguments for OCR
import requests
import time

def main(df):
    try:
        print("Process Started")
        # In[]: Creating data to post
        url = 'API URL'
        response = requests.post(url,data=data_to send)
        # In[]: Reading the returned information
        res = response.json()
        # results_text = res['IttsText']
        print(res)
```

return [res]



Corresponding File:



## 12. Signing Out

The users can Sign-out from the Data Pipeline tab at any given stage, but preferable is that the users should complete all the tasks they wish to perform and save them before closing the tab or signing out from the Platform.

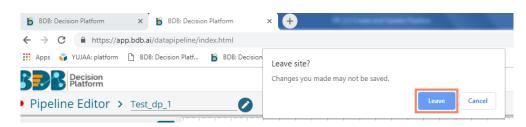
The Signing Out process for the Data Pipeline has two steps:

#### 1. Closing the BDB Data Pipeline

Once you have completed the Data Pipeline tasks, save your work, and close the Data Pipeline tab. i) Click the **'Close'** button (the 'X' on the right edge) from the Data Pipeline tab.

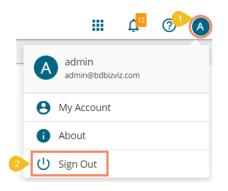


- ii) A window appears to confirm the action of closing the BDB Pipeline.
- iii) Click the 'Leave' option to close the selected pipeline.



#### 2. Sign Out from the BDB Platform

- i) After closing the Data Pipeline plugin, the users get redirected to the Platform homepage.
- ii) Click the 'User Profile' icon on the Platform homepage.
- iii) Click the 'Sign Out' option.



iv) The user successfully signs off from the BDB Platform.



Note: Clicking on the 'Sign Out' option redirects the user back to the login page of the BDB platform.

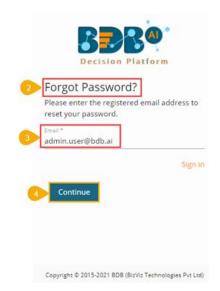
#### **12.1. Forgot Password Option**

The users are provided with a choice to change the password on the Login page of the platform.

i) Click the 'Forgot Password?' option from the Sign In page.

De	<b>BB</b> cision Platf	AD orm
🖴 Email / Use	r Id *	
• Password		
Auth Type		
Enterprise		•
	1 Forgot	Password ?
Converter @ 2019	-2021 RDR /RI+V/+	echnologies Pvt Ltd)

- ii) The 'Forgot Password?' page opens.
- iii) Provide the email id that is registered with BDB to send the reset password link.
- iv) Click the '**Continue**' option.



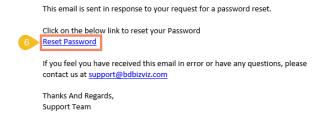
v) The user may be redirected to select a space in case of multiple spaces under one server link (The user needs to select a space and click the '**Continue**' option once again). If a user does not have multiple spaces, a message appears to notify the user that the password reset link (The users receive the reset link via their registered email.)

Password reset link has been sent to your email

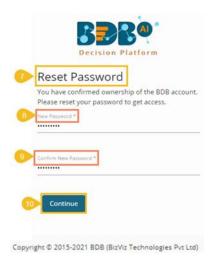
www.bdb.ai



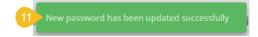
vi) Click the link from your registered email.



- vii) The user gets redirected to the 'Reset Password' page to set a new password.
- viii) Set a new password.
- ix) Confirm the newly set password.
- x) Click the '**Continue**' option.



xi) The password for the selected BDB account gets reset and a message appears to inform the user.



Note: The user gets redirected back to the Sign In page after successfully resetting the password.

#### 12.2. Force Login

The '**Force Login**' functionality has been introduced to control the number of active sessions up to three. The users can access only 3 sessions at a time when they try to access the 4<sup>th</sup> session, a warning message displays to inform that the user has consumed the permitted sessions and, a click on the '**Force Login**' would kill all those active sessions.

- i) Navigate to the BDB Platform Login page.
- ii) Enter the valid credentials to log in.
- iii) Click the 'Sign In' option.





- iv) The user gets the following message if the permitted active sessions (3 sessions at a time) are consumed.
- v) Click the '**Force Login**' option.

Permitted already co		
Force login will sessions.	terminate all	the active
Cancel	6	Force Log

- vi) A warning message appears the currently active sessions get killed, and the user gets redirected to the BDB Platform Sign In page.
- vii) The user needs to provide valid credentials once again and click the '**Continue**' option to access the platform.