

LYFTDATA PRODUCT DOCUMENTATION

DSL Outputs Reference

Output connector and destination reference pages from the generated DSL corpus.

Version 2.1.0. Generated from the docs.lyftdata.com source corpus on 2026-05-25.

Contents

1. Azure Blob
2. Azure Monitor Data Collector
3. BigQuery
4. ClickHouse
5. Discard
6. Elasticsearch
7. File Store
8. Log Files
9. FTP
10. Google Cloud Storage
11. HTTP Get
12. HTTP Post
13. Kafka
14. Message
15. MongoDB
16. MQTT
17. MySQL
18. OpenSearch
19. OpenTelemetry
20. PostgreSQL
21. Print
22. S3
23. Splunk HEC
24. ODBC SQL
25. SQL Server
26. Syslog
27. Web Dav Store
28. WebSocket
29. Worker Channel
30. Worker KV

Azure Blob

Reference for the Azure Blob component in LyftData's DSL

Source: <reference/dsl/outputs/azure-blob.md>

Azure Blob (`azure-blob`)

Send data to a Microsoft Azure Storage blob (Block Storage).

Block Store Cloud Enterprise edition binary json raw

Minimal example

```
output:
azure-blob: blob-destination: name: ~ container-name: ~ storage-account: ~ storage-master-
key: ~
```

JSON

```
{
"output": { "azure-blob": { "blob-destination": { "name": null }, "container-name": null,
"storage-account": null, "storage-master-key": null } } }
```

Contents

- [Minimal example](#)
- [Authentication](#)
- [Behavior](#)
- [Location](#)
- [Object Properties](#)
- [Processing](#)
- [Reliability](#)
- [Schema](#)

Authentication

Authentication

Field	Type	Required	Description
<code>storage-account</code>	<code>string</code>	✓	The Storage Account Name to be used (credential).
<code>storage-master-key</code>	<code>string</code>	✓	The Storage Master Key to be used (credential).

Behavior

Behavior

Field	Type	Required	Description
<code>mode</code>	<code>Mode</code>		Either writing or deleting a blob. Allowed values: <code>put</code> , <code>delete</code>

Location

Location

Field	Type	Required	Description
<code>container-name</code>	<code>string</code>	✓	The storage service container for created blobs.
<code>blob-destination</code>	<code>Blob Destination</code>	✓	Either a literal blob name or the name of the field to read it from. Allowed values: <code>name</code> , <code>field</code>

Object Properties

Object Properties

Field	Type	Required	Description
<code>disable-blob-name-guid</code> ✓	<code>boolean (bool)</code>		Do not append a GUID to the output blob name. Default: <code>false</code>
<code>guid-prefix</code>	<code>string</code>		GUID Prefix, will be prepended to the GUID, the default value is "/".
<code>guid-suffix</code>	<code>string</code>		GUID Suffix, will be appended to the GUID if specified.
<code>content-type</code>	<code>string</code>		Set the created blob to the following content-type. Default: <code>text/plain</code>

Processing

Processing

Field	Type	Required	Description
<code>batch</code>	<code>Batch</code>		Batching output events.
<code>input-field</code>	<code>field (string)</code>		Send only the content of the specified field to the blob. Examples: <code>data_field</code>
<code>preprocessors</code>	<code>[Output Preprocessor[]]</code> (#output-preprocessor-options)		Preprocessors (process data before making it available for upload) these processors will be run in the order

they are specified. Allowed values: `gzip`, `parquet`, `base64`

Reliability

Reliability

Field	Type	Required	Description
<code>retry</code>	<code>Retry</code>		How to retry operation if it fails.
<code>track-schema</code> ✓	<code>boolean</code> (<code>bool</code>)		Check the schema of the written data and update <code>__SCHEMA_NUMBER</code> (written data must be JSON). Default: <code>false</code>

Schema

- [Blob Destination Options](#)
- [Batch Fields](#)
- [Retry Fields](#)
- [Mode Options](#)
- [Batch - Mode Options](#)
- [Output Preprocessor Options](#)

Blob Destination Options

Option	Name	Type	Description
<code>name</code>	Name	<code>string</code>	The name for the output blob, this will be suffixed by a GUID by default.
<code>field</code>	Field	<code>string</code>	A field containing the value for <code>output_blob</code> .

Batch Fields

Field	Type	Required	Description
<code>fixed-size</code> ✓	<code>number</code> (<code>integer</code>)		maximum number of events in an output batch. Examples: <code>42</code> , <code>1.2e-10</code>
<code>mode</code>	<code>Mode</code>	✓	If 'document' send on end of document generated by input. If 'fixed', use <code>fixed_size</code> . Allowed values: <code>fixed</code> , <code>document</code>
<code>timeout</code>	<code>time-interval</code> (<code>string</code>)	✓	interval after which the batch is sent, to keep throughput going (default 100ms). Default: <code>100ms</code> Examples: <code>500ms</code> , <code>2h</code>
<code>header</code>	<code>multiline-text</code> (<code>string</code>)		put a header line before the batch.
<code>footer</code>	<code>multiline-text</code> (<code>string</code>)		put a header line after the last line of the batch.
<code>use-document-marker</code> ✓	<code>boolean</code> (<code>bool</code>)		Enrich the job metadata with a document marker (for document handling in batch mode). Default: <code>false</code>

<code>wrap-as-json</code> ✓	<code>boolean (bool)</code>	Format the output batch as a JSON array. Default: <code>false</code>
--------------------------------	-----------------------------	--

Retry Fields

Field	Type	Required	Description
<code>timeout</code>	<code>time-interval (string)</code>	✓	timeout (e.g. 500ms, 2s etc. - default is 30). Examples: <code>500ms</code> , <code>2h</code>
<code>retries</code>	<code>number (integer)</code>		number of retries. Examples: <code>42</code> , <code>1.2e-10</code>

Mode Options

Value	Name	Description
<code>put</code>	put	Put Objects
<code>delete</code>	delete	Delete Objects

Batch - Mode Options

Value	Name	Description
<code>fixed</code>	fixed	Fixed
<code>document</code>	document	Document

Output Preprocessor Options

Value	Name	Description
<code>gzip</code>	gzip	Gzip the output data
<code>parquet</code>	parquet	Extract the received data as JSON rows from a parquet file
<code>base64</code>	base64	Decode base64 as binary

Azure Monitor Data Collector

Reference for the Azure Monitor Data Collector component in LyftData's DSL

Source: `reference/dsl/outputs/azure-monitor.md`

Azure Monitor Data Collector (`azure-monitor`)

Specialized preset that wraps `azure-monitor` output parameters so deployments can ship datasets into Azure Log Analytics/Sentinel without hand-authoring `http-post` jobs.

Observability Enterprise edition json

Minimal example

```
output:
azure-monitor: shared-key: ~ workspace-id: ~
```

JSON

```
{
  "output": { "azure-monitor": { "shared-key": null, "workspace-id": null } } }
```

Contents

- [Minimal example](#)
- [Authentication](#)
- [Diagnostics](#)
- [Endpoint](#)
- [Payload](#)
- [Processing](#)
- [Reliability](#)
- [Schema](#)

Authentication

Authentication

Field	Type	Required	Description
<code>shared-key</code>	<code>string</code>	<input checked="" type="checkbox"/>	Base64-encoded workspace shared key.

Diagnostics

Diagnostics

Field	Type	Required	Description
<code>test-url</code>	<code>url</code> (<code>string</code>)		Override endpoint during dry-runs (bypasses workspace+api_version). Examples: <code>https://example.com/path</code>

Endpoint

Endpoint

Field	Type	Required	Description
<code>workspace-id</code>	<code>string</code>	<input checked="" type="checkbox"/>	Azure Log Analytics workspace ID (customer ID).
<code>ingestion-uri</code>	<code>string</code>		Full ingestion URI (defaults to <code>https://{workspace}.ods.opinsights.azure.com/api/logs?api-version=2016-04-01</code>).
<code>api-version</code>	<code>string</code>		API version appended when <code>ingestion_uri</code> is omitted.

Payload

Payload

Field	Type	Required	Description
<code>log-type</code>	<code>string</code>		Log-Type assigned inside Azure Monitor.
<code>body-field</code>	<code>field</code> (<code>string</code>)		Restrict the payload to this field (must contain JSON). Examples: <code>data_field</code>
<code>time-generated-field</code>	<code>field</code> (<code>string</code>)		ISO 8601 timestamp field mapped to <code>time-generated-field</code> . Examples: <code>data_field</code>
<code>resource-id</code>	<code>string</code>		Azure Resource ID applied via <code>x-ms-AzureResourceId</code> .

Processing

Processing

Field	Type	Required	Description
<code>batch</code>	<code>Batch</code>		Batching behavior for upstream events.

Reliability

Reliability

Field	Type	Required	Description
<code>retry</code>	<code>Retry</code>		Retry policy applied to Azure Monitor requests.

Schema

- [Batch Fields](#)
- [Retry Fields](#)
- [Batch - Mode Options](#)

Batch Fields

Field	Type	Required	Description
<code>fixed-size</code> ✓	<code>number</code> (<code>integer</code>)		maximum number of events in an output batch. Examples: <code>42</code> , <code>1.2e-10</code>
<code>mode</code>	<code>Mode</code>	✓	If 'document' send on end of document generated by input. If 'fixed', use <code>fixed_size</code> . Allowed values: <code>fixed</code> , <code>document</code>
<code>timeout</code>	<code>time-interval</code> (<code>string</code>)	✓	interval after which the batch is sent, to keep throughput going (default 100ms). Default: <code>100ms</code> Examples: <code>500ms</code> , <code>2h</code>
<code>header</code>	<code>multiline-text</code> (<code>string</code>)		put a header line before the batch.
<code>footer</code>	<code>multiline-text</code> (<code>string</code>)		put a header line after the last line of the batch.
<code>use-document-marker</code> ✓	<code>boolean</code> (<code>bool</code>)		Enrich the job metadata with a document marker (for document handling in batch mode). Default: <code>false</code>
<code>wrap-as-json</code> ✓	<code>boolean</code> (<code>bool</code>)		Format the output batch as a JSON array. Default: <code>false</code>

Retry Fields

Field	Type	Required	Description
<code>timeout</code>	<code>time-interval</code> (<code>string</code>)	✓	timeout (e.g. 500ms, 2s etc. - default is 30). Examples: <code>500ms</code> , <code>2h</code>
<code>retries</code>	<code>number</code> (<code>integer</code>)		number of retries. Examples: <code>42</code> , <code>1.2e-10</code>

Batch - Mode Options

Value	Name	Description
<code>fixed</code>	<code>fixed</code>	Fixed
<code>document</code>	<code>document</code>	Document

BigQuery

Reference for the BigQuery component in LyftData's DSL

Source: [reference/dsl/outputs/big-query.md](#)

BigQuery (`big-query`)

Stream data into Google BigQuery.

Data Warehouse Enterprise edition json

Minimal example

```
output:
big-query: connection: project: ~
```

JSON

```
{
"output": { "big-query": { "connection": { "project": null } } } }
```

Contents

- [Minimal example](#)
- [Authentication](#)
- [Connection](#)
- [Load Job](#)
- [Operation](#)
- [Parameters](#)
- [Processing](#)
- [Reliability](#)
- [Schema](#)
- [Streaming](#)
- [Schema](#)

Authentication

Authentication

Field	Type	Required	Description
<code>credentials</code>	<code>Credentials</code>		Authentication credentials. Allowed values: <code>service-account</code> , <code>service-account-file</code> , <code>workload-identity</code> , <code>application-</code>

default

Connection

Connection

Field	Type	Required	Description
connection	Connection	<input checked="" type="checkbox"/>	Connection details (project, dataset, table).

Load Job

Load Job

Field	Type	Required	Description
load-job	Load Job		Configuration for Load Jobs (GCS staging).

Operation

Operation

Field	Type	Required	Description
mode	Mode		Output mode (streaming or load-job). Default: streaming Allowed values: streaming , load-job

Parameters

Parameters

Field	Type	Required	Description
parameters	[Parameters[]](#parameters-fields)		Parameters for template expansion.

Processing

Processing

Field	Type	Required	Description
batch	Batch		Batching configuration (size/timeout).

Reliability

Reliability

Field	Type	Required	Description
<code>retry</code>	<code>Retry</code>		Retry configuration for transient failures.

Schema

Schema

Field	Type	Required	Description
<code>schema</code>	<code>Schema</code>		Schema handling and coercion settings.

Streaming

Streaming

Field	Type	Required	Description
<code>streaming</code>	<code>Streaming</code>		Configuration for Storage Write API.

Schema

- [Credentials Options](#)
- [Credentials - Service Account Fields](#)
- [Credentials - Service Account File Fields](#)
- [Credentials - Workload Identity Fields](#)
- [Connection - Impersonation Fields](#)
- [Connection Fields](#)
- [Load Job Fields](#)
- [Parameters Fields](#)
- [Batch Fields](#)
- [Retry Fields](#)
- [Schema - Overrides Fields](#)
- [Schema Fields](#)
- [Streaming Fields](#)
- [Load Job - Write Disposition Options](#)
- [Load Job - Create Disposition Options](#)
- [Mode Options](#)
- [Batch - Mode Options](#)
- [Streaming - Stream Type Options](#)

Credentials Options

Option	Name	Type	Description
--------	------	------	-------------

<code>service-account</code>	Service Account	object	
<code>service-account-file</code>	Service Account File	object	
<code>workload-identity</code>	Workload Identity	object	
<code>application-default</code>	Application Default	map	

Credentials - Service Account Fields

Field	Type	Required	Description
<code>secret-id</code> ✓	string	✓	

Credentials - Service Account File Fields

Field	Type	Required	Description
<code>path</code>	string	✓	

Credentials - Workload Identity Fields

Field	Type	Required	Description
<code>audience</code> ✓	string	✓	
<code>service-account-email</code> ✓	string		

Connection - Impersonation Fields

Field	Type	Required	Description
<code>service-account</code> ✓	string	✓	
<code>delegates</code> ✓	string[]		

Connection Fields

Field	Type	Required	Description
<code>project</code> ✓	string	✓	
<code>dataset</code> ✓	string		
<code>table</code> ✓	string		
<code>billing-project</code> ✓	string		
<code>location</code> ✓	string		
<code>impersonation</code>	Impersonation		Impersonate a service account (requires IAM Credentials API).

Load Job Fields

Field	Type	Required	Description
<code>staging-uri</code> ✓	<code>string</code>	✓	GCS URI for staging files (gs://bucket/path).
<code>write-disposition</code>	<code>Write Disposition</code>		Default: <code>append</code> Allowed values: <code>write-append</code> , <code>write-truncate</code> , <code>write-empty</code>
<code>create-disposition</code>	<code>Create Disposition</code>		Default: <code>create-if-needed</code> Allowed values: <code>create-if-needed</code> , <code>create-never</code>

Parameters Fields

Field	Type	Required	Description
<code>name</code>	<code>string</code>	✓	
<code>value</code> ✓	<code>string</code>		
<code>r-type</code>	<code>string</code>		

Batch Fields

Field	Type	Required	Description
<code>fixed-size</code> ✓	<code>number (integer)</code>		maximum number of events in an output batch. Examples: <code>42</code> , <code>1.2e-10</code>
<code>mode</code>	<code>Mode</code>	✓	If 'document' send on end of document generated by input. If 'fixed', use <code>fixed_size</code> . Allowed values: <code>fixed</code> , <code>document</code>
<code>timeout</code>	<code>time-interval (string)</code>	✓	interval after which the batch is sent, to keep throughput going (default 100ms). Default: <code>100ms</code> Examples: <code>500ms</code> , <code>2h</code>
<code>header</code>	<code>multiline-text (string)</code>		put a header line before the batch.
<code>footer</code>	<code>multiline-text (string)</code>		put a header line after the last line of the batch.
<code>use-document-marker</code> ✓	<code>boolean (bool)</code>		Enrich the job metadata with a document marker (for document handling in batch mode). Default: <code>false</code>
<code>wrap-as-json</code> ✓	<code>boolean (bool)</code>		Format the output batch as a JSON array. Default: <code>false</code>

Retry Fields

Field	Type	Required	Description
<code>timeout</code>	<code>time-interval (string)</code>	✓	timeout (e.g. 500ms, 2s etc. - default is 30). Examples: <code>500ms</code> , <code>2h</code>
<code>retries</code>	<code>number (integer)</code>		number of retries. Examples: <code>42</code> , <code>1.2e-10</code>

Schema - Overrides Fields

Field	Type	Required	Description
<code>field-path</code>	<code>string</code>	<input checked="" type="checkbox"/>	
<code>target-type</code>	<code>string</code>		

Schema Fields

Field	Type	Required	Description
<code>auto-fetch</code> ✓	<code>boolean (bool)</code>		Default: <code>true</code>
<code>overrides</code>	<code>[Overrides[]](#schema-overrides-fields)</code>		

Streaming Fields

Field	Type	Required	Description
<code>stream-type</code>	<code>Stream Type</code>		Default: <code>default</code> Allowed values: <code>default</code> , <code>committed</code> , <code>pending</code> , <code>buffered</code>
<code>insert-id-field</code>	<code>string</code>		Field to use for deduplication (insertId).

Load Job - Write Disposition Options

Value	Name	Description
<code>write-append</code>	<code>write-append</code>	Write Append
<code>write-truncate</code>	<code>write-truncate</code>	Write Truncate
<code>write-empty</code>	<code>write-empty</code>	Write Empty

Load Job - Create Disposition Options

Value	Name	Description
<code>create-if-needed</code>	<code>create-if-needed</code>	Create If Needed
<code>create-never</code>	<code>create-never</code>	Create Never

Mode Options

Value	Name	Description
<code>streaming</code>	<code>streaming</code>	Streaming
<code>load-job</code>	<code>load-job</code>	Load Job

Batch - Mode Options

Value	Name	Description
<code>fixed</code>	fixed	Fixed
<code>document</code>	document	Document

Streaming - Stream Type Options

Value	Name	Description
<code>default</code>	default	Default
<code>committed</code>	committed	Committed
<code>pending</code>	pending	Pending
<code>buffered</code>	buffered	Buffered

ClickHouse

Reference for the ClickHouse component in LyftData's DSL

Source: <reference/dsl/outputs/clickhouse.md>

ClickHouse (`clickhouse`)

Write events to a ClickHouse table using the HTTP interface or native endpoint.

Database Enterprise edition json

Minimal example

```
output:
clickhouse: endpoint: ~ table: ~
```

JSON

```
{
"output": { "clickhouse": { "endpoint": null, "table": null } } }
```

Contents

- [Minimal example](#)
- [Fields](#)
- [Authentication](#)
- [Connection](#)
- [Payload](#)
- [Schema](#)
- [Schema](#)

Fields

Field	Type	Required	Description
<code>retry</code>	<code>Retry</code>		How to retry failed inserts.
<code>batch</code>	<code>Batch</code>		Batch events before inserting.

Authentication

Authentication

Field	Type	Required	Description
<code>auth</code>	<code>Auth</code>		Optional authentication settings (basic auth, token, or TLS material).

Connection

Connection

Field	Type	Required	Description
<code>endpoint</code>	<code>string</code>	✓	Base endpoint (example: <code>https://localhost:8443</code> or <code>http://cluster-router:8123</code>).
<code>database</code>	<code>string</code>		Optional database override; defaults to server-side default database.
<code>table</code>	<code>string</code>	✓	Target table for writes.
<code>compression</code>	<code>Compression</code>		Optional compression codec negotiated with the server. Allowed values: <code>none</code> , <code>lz4</code>
<code>settings</code>	<code>map (string)</code>		Additional ClickHouse setting key/value pairs (forwarded as query params).
<code>insecure-skip-verify</code> ✓	<code>boolean (bool)</code>		Skip TLS certificate verification (development only). Default: <code>false</code>
<code>timezone</code>	<code>string</code>		Optional timezone override used when formatting temporal data.

Payload

Payload

Field	Type	Required	Description
<code>input-field</code>	<code>field (string)</code>		Optional event field to treat as the raw payload. When omitted the entire event is used. Examples: <code>data_field</code>
<code>format</code>	<code>Format</code>		Insert format to use when sending data to ClickHouse. Allowed values: <code>json-each-row</code> , <code>json-compact-each-row</code> , <code>csv</code> , <code>values</code>

Schema

Schema

Field	Type	Required	Description
<code>columns</code>	<code>[Columns[]](#columns-fields)</code>		Declarative column mapping; leave empty to stream <code>JSONEachRow</code> documents.

Schema

- [Retry Fields](#)
- [Batch Fields](#)
- [Auth Fields](#)
- [Columns Fields](#)
- [Settings Table](#)
- [Batch - Mode Options](#)
- [Compression Options](#)
- [Format Options](#)

Retry Fields

Field	Type	Required	Description
<code>timeout</code>	<code>time-interval</code> (<code>string</code>)	✓	timeout (e.g. 500ms, 2s etc. - default is 30). Examples: <code>500ms</code> , <code>2h</code>
<code>retries</code>	<code>number</code> (<code>integer</code>)		number of retries. Examples: <code>42</code> , <code>1.2e-10</code>

Batch Fields

Field	Type	Required	Description
<code>fixed-size</code> ✓	<code>number</code> (<code>integer</code>)		maximum number of events in an output batch. Examples: <code>42</code> , <code>1.2e-10</code>
<code>mode</code>	<code>Mode</code>	✓	If 'document' send on end of document generated by input. If 'fixed', use <code>fixed_size</code> . Allowed values: <code>fixed</code> , <code>document</code>
<code>timeout</code>	<code>time-interval</code> (<code>string</code>)	✓	interval after which the batch is sent, to keep throughput going (default 100ms). Default: <code>100ms</code> Examples: <code>500ms</code> , <code>2h</code>
<code>header</code>	<code>multiline-text</code> (<code>string</code>)		put a header line before the batch.
<code>footer</code>	<code>multiline-text</code> (<code>string</code>)		put a header line after the last line of the batch.
<code>use-document-marker</code> ✓	<code>boolean</code> (<code>bool</code>)		Enrich the job metadata with a document marker (for document handling in batch mode). Default: <code>false</code>
<code>wrap-as-json</code> ✓	<code>boolean</code> (<code>bool</code>)		Format the output batch as a JSON array. Default: <code>false</code>

Auth Fields

Field	Type	Required	Description
<code>username</code>	<code>string</code>		
<code>password</code>	<code>string</code>		
<code>access-token</code>	<code>string</code>		

<code>tls-certificate</code>	<code>string</code>		
<code>tls-key</code>	<code>string</code>		

Columns Fields

Field	Type	Required	Description
<code>name</code>	<code>string</code>	<input checked="" type="checkbox"/>	
<code>field</code>	<code>string</code>		Event field name (dot-notation) to pull data from.
<code>pointer</code>	<code>json-pointer</code> (<code>string</code>)		JSON Pointer to extract from the event (used when field is insufficient).
<code>expression</code>	<code>string</code>		Optional ClickHouse expression to compute the column.
<code>r-type</code>	<code>string</code>		Explicit ClickHouse type (e.g. DateTime64(3), Nullable(String)).
<code>default</code>	<code>string</code>		Default value used when the source field/pointer resolves to null.
<code>codec</code>	<code>string</code>		Optional codec hint (e.g. LZ4, ZSTD).

Settings Table

Setting	Value
<code>setting.name</code>	<code>value</code>

Value format: `templated-text` .

Batch - Mode Options

Value	Name	Description
<code>fixed</code>	fixed	Fixed
<code>document</code>	document	Document

Compression Options

Value	Name	Description
<code>none</code>	none	None
<code>lz4</code>	lz4	Lz4

Format Options

Value	Name	Description
<code>json-each-row</code>	json-each-row	Json Each Row

<code>json-compact-each-row</code>	json-compact-each-row	Json Compact Each Row
<code>csv</code>	csv	Csv
<code>values</code>	values	Values

Discard

Reference for the Discard component in LyftData's DSL

Source: <reference/dsl/outputs/discard.md>

Discard (`discard`)

Discard any output from the job.

Debug binary json raw

Minimal example

```
output:  
discard: {}
```

JSON

```
{  
  "output": { "discard": {} } }
```

Elasticsearch

Reference for the Elasticsearch component in LyftData's DSL

Source: [reference/dsl/outputs/elasticsearch.md](#)

Elasticsearch (`elasticsearch`)

Bulk-index events into Elasticsearch clusters.

Search Enterprise edition binary json raw

Minimal example

```
output:
  elasticsearch: common: {}
```

JSON

```
{
  "output": { "elasticsearch": { "common": {} } } }
```

Contents

- [Minimal example](#)
- [Fields](#)
- [Schema](#)

Fields

Field	Type	Required	Description
<code>common</code>	Elastic Output Spec	<input checked="" type="checkbox"/>	Shared Elasticsearch connection and indexing options Shared configuration applied to all Elasticsearch outputs.

Schema

- [Elastic Output Spec - Retry Fields](#)
- [Elastic Output Spec - Batch Fields](#)
- [Elastic Output Spec - Elastic Auth - Basic Fields](#)
- [Elastic Output Spec - Elastic Auth - Api Key Fields](#)
- [Elastic Output Spec - Elastic Auth - Tls Fields](#)
- [Elastic Output Spec - Elastic Auth Fields](#)

- [Elastic Output Spec - Elastic AWS SigV4 Fields](#)
- [Elastic Output Spec - Elastic Data Stream Fields](#)
- [Elastic Output Spec Fields](#)
- [Elastic Output Spec - Headers Table](#)
- [Elastic Output Spec - Query Table](#)
- [Elastic Output Spec - Mode Options](#)
- [Elastic Output Spec - Op Type Options](#)

Elastic Output Spec - Retry Fields

Field	Type	Required	Description
<code>timeout</code>	<code>time-interval</code> (<code>string</code>)	✓	timeout (e.g. 500ms, 2s etc. - default is 30). Examples: <code>500ms</code> , <code>2h</code>
<code>retries</code>	<code>number</code> (<code>integer</code>)		number of retries. Examples: <code>42</code> , <code>1.2e-10</code>

Elastic Output Spec - Batch Fields

Field	Type	Required	Description
<code>uuid-field</code>	<code>field</code> (<code>string</code>)		Field where generated uuid, the unique marker for the group, will be stored. Examples: <code>data_field</code>
<code>invocation-time-field</code>	<code>field</code> (<code>string</code>)		Field where invocation time will be stored. Examples: <code>data_field</code>
<code>completion-time-field</code>	<code>field</code> (<code>string</code>)		Field where completion (end of execution) time will be stored. Examples: <code>data_field</code>
<code>begin-marker-field</code>	<code>field</code> (<code>string</code>)		Field used to mark first event in the group. Examples: <code>data_field</code>
<code>end-marker-field</code>	<code>field</code> (<code>string</code>)		Field used to mark last event in the group. Examples: <code>data_field</code>
<code>line-count-field</code>	<code>field</code> (<code>string</code>)		Field used to store the line count of the batch. Examples: <code>data_field</code>
<code>line-num-field</code>	<code>field</code> (<code>string</code>)		Field used to store the line number of the batch. Examples: <code>data_field</code>

Elastic Output Spec - Elastic Auth - Basic Fields

Field	Type	Required	Description
<code>username</code> ✓	<code>string</code>		
<code>password</code> ✓	<code>string</code>		

Elastic Output Spec - Elastic Auth - Api Key Fields

Field	Type	Required	Description
-------	------	----------	-------------

id ✓	string		
key ✓	string		
base64 ✓	string		

Elastic Output Spec - Elastic Auth - Tls Fields

Field	Type	Required	Description
ca-certificate ✓	string		PEM encoded CA certificate bundle.
client-certificate ✓	string		PEM encoded client certificate for mutual TLS.
client-key ✓	string		PEM encoded private key for mutual TLS.
insecure-skip-verify ✓	boolean (bool)		Skip TLS certificate verification (development only). Default: false

Elastic Output Spec - Elastic Auth Fields

Field	Type	Required	Description
basic	Basic		
api-key	Api Key		
bearer-token ✓	string		
tls	Tls		

Elastic Output Spec - Elastic AWS SigV4 Fields

Field	Type	Required	Description
region ✓	string	✓	AWS region used for request signing.
service ✓	string		Override AWS service name (default es).
profile ✓	string		Optional shared credentials profile.
role-arn ✓	string		AssumeRole ARN for federated access.
session-name ✓	string		Session name when assuming roles.
access-key-id ✓	string		Access key used for signing requests.
secret-access-key ✓	string		Secret key used for signing requests.
session-token ✓	string		Optional session token for temporary credentials.

Elastic Output Spec - Elastic Data Stream Fields

Field	Type	Required	Description
-------	------	----------	-------------

<code>r-type</code> ✓	<code>string</code>	✓	Data stream type (logs, metrics, traces).
<code>dataset</code> ✓	<code>string</code>	✓	Data stream dataset (e.g. service.application).
<code>namespace</code> ✓	<code>string</code>	✓	Data stream namespace (e.g. prod, staging).

Elastic Output Spec Fields

Field	Type	Required	Description
<code>retry</code>	Retry		Retry policy for bulk failures.
<code>batch</code>	Batch		Logical batching hints shared with other outputs.
<code>endpoint</code> ✓	<code>url (string)</code>		HTTPS endpoint for the cluster (required when cloud-id is unset). Examples: <code>https://example.com/path</code>
<code>cloud-id</code> ✓	<code>string</code>		Elastic Cloud / Serverless Cloud ID (mutually exclusive with endpoint).
<code>region</code> ✓	<code>string</code>		Optional default region used for auth helpers and logging.
<code>auth</code>	Elastic Auth		Authentication strategies (basic, API keys, bearer).
<code>aws</code>	Elastic AWS SigV4		AWS SigV4 credentials for managed OpenSearch/Elasticsearch.
<code>mode</code>	Mode		Indexing lifecycle mode (classic index vs data stream). Allowed values: <code>classic-index</code> , <code>data-stream</code>
<code>index</code> ✓	<code>string</code>		Static index/alias target when mode=classic-index.
<code>index-field</code>	<code>field (string)</code>		Event field that provides the target index/alias. Examples: <code>data_field</code>
<code>index-pointer</code>	<code>json-pointer (string)</code>		JSON pointer resolving to the index/alias.
<code>data-stream</code>	Elastic Data Stream		Data stream configuration when mode=data-stream.
<code>document-id-field</code>	<code>field (string)</code>		Event field containing the document_id. Examples: <code>data_field</code>
<code>document-id-pointer</code>	<code>json-pointer (string)</code>		JSON pointer resolving to the document_id.
<code>document-id-expression</code>	<code>string</code>		Expression used to compute the document_id.
<code>routing-field</code>	<code>field (string)</code>		Event field providing custom routing key. Examples: <code>data_field</code>
<code>routing-pointer</code>	<code>json-pointer (string)</code>		JSON pointer providing custom routing key.
<code>routing-expression</code>	<code>string</code>		Expression computing the routing key.

<code>pipeline</code> ✓	<code>string</code>		Optional ingest pipeline to apply server-side.
<code>op-type</code>	<code>Op Type</code>		Bulk operation type for each event. Allowed values: <code>index</code> , <code>create</code> , <code>update</code> , <code>delete</code>
<code>retry-on-conflict</code>	<code>number</code> (<code>integer</code>)		Retry on conflict (for update/upsert operations). Examples: <code>42</code> , <code>1.2e-10</code>
<code>refresh</code> ✓	<code>boolean</code> (<code>bool</code>)		Force refresh after each batch. Default: <code>false</code>
<code>gzip</code> ✓	<code>boolean</code> (<code>bool</code>)		Compress bulk payloads with gzip. Default: <code>false</code>
<code>input-field</code>	<code>field</code> (<code>string</code>)		Event field containing the JSON document payload. Examples: <code>data_field</code>
<code>input-pointer</code>	<code>json-pointer</code> (<code>string</code>)		JSON pointer for extracting the payload.
<code>payload-template</code>	<code>string</code>		Template expression to render the payload when not using event body.
<code>headers</code> ✓	<code>map</code> (<code>string</code>)		Additional HTTP headers forwarded with each request.
<code>query</code> ✓	<code>map</code> (<code>string</code>)		Static query parameters appended to bulk requests.
<code>request-timeout-secs</code>	<code>number</code> (<code>integer</code>)		Request timeout in seconds. Examples: <code>42</code> , <code>1.2e-10</code>
<code>connect-timeout-secs</code>	<code>number</code> (<code>integer</code>)		Connect timeout in seconds. Examples: <code>42</code> , <code>1.2e-10</code>

Elastic Output Spec - Headers Table

Header	Value
<code>Header-Name</code>	<code>value</code>

Value format: `templated-text` .

Elastic Output Spec - Query Table

Parameter	Value
<code>param</code>	<code>value</code>

Value format: `templated-text` .

Elastic Output Spec - Mode Options

Value	Name	Description
<code>classic-index</code>	<code>classic-index</code>	Standard index or alias writes

<code>data-stream</code>	data-stream	Data stream aware writes (auto-selects stream based on dataset/namespace)
--------------------------	-------------	---

Elastic Output Spec - Op Type Options

Value	Name	Description
<code>index</code>	index	Index (create or replace) each document
<code>create</code>	create	Only create new documents; fail if <code>_id</code> exists
<code>update</code>	update	Update existing documents (requires partial document or script)
<code>delete</code>	delete	Delete documents (requires <code>_id</code>)

File Store

Reference for the File Store component in LyftData's DSL

Source: <reference/dsl/outputs/file-store.md>

File Store (`file-store`)

Write to a local file system object store.

Block Store File binary json raw

Minimal example

```
output:
file-store: object-name: name: ~ path: ""
```

JSON

```
{
  "output": { "file-store": { "object-name": { "name": null }, "path": "" } } }
```

Contents

- [Minimal example](#)
- [Behavior](#)
- [Location](#)
- [Object Properties](#)
- [Processing](#)
- [Reliability](#)
- [Schema](#)

Behavior

Behavior

Field	Type	Required	Description
<code>mode</code>	<code>Mode</code>		Specify whether we're putting or deleting. Allowed values: <code>put</code> , <code>delete</code>

Location

Location

Field	Type	Required	Description
<code>path</code>	<code>path</code> (<code>string</code>)	✓	Select filesystem path as root of store. Examples: <code>/path/to/file</code> , <code>c:\users\joe\data\file.txt</code>
<code>object-name</code>	<code>ObjectName</code>	✓	File name (may contain slashes). Allowed values: <code>name</code> , <code>field</code>

Object Properties

Object Properties

Field	Type	Required	Description
<code>disable-object-name-guid</code> ✓	<code>boolean</code> (<code>bool</code>)		Disable the GUID prefix if you want object name to be treated literally (off for deletes). Default: <code>false</code>
<code>guid-prefix</code>	<code>string</code>		GUID Prefix, will be prepended to the GUID, the default value is <code>"/</code> .
<code>guid-suffix</code>	<code>string</code>		GUID Suffix, will be appended to the GUID if specified.

Processing

Processing

Field	Type	Required	Description
<code>batch</code>	<code>Batch</code>		Write events as batches.
<code>input-field</code>	<code>field</code> (<code>string</code>)		Use the specified field as the content for the file line. Examples: <code>data_field</code>
<code>preprocessors</code>	[<code>Output Preprocessor</code>]] (#output-preprocessor-options)		Preprocessors (process data before making it available for upload) these processors will be run in the order they are specified. Allowed values: <code>gzip</code> , <code>parquet</code> , <code>base64</code>

Reliability

Reliability

Field	Type	Required	Description
<code>retry</code>	<code>Retry</code>		Retry on failure.
<code>track-schema</code> ✓	<code>boolean</code> (<code>bool</code>)		Check the schema of the written data and update <code>__SCHEMA_NUMBER</code> (written data must be JSON). Default: <code>false</code>

Schema

- [Object Name Options](#)
- [Batch Fields](#)
- [Retry Fields](#)
- [Mode Options](#)
- [Batch - Mode Options](#)
- [Output Preprocessor Options](#)

Object Name Options

Option	Name	Type	Description
<code>name</code>	Name	<code>string</code>	Object Name.
<code>field</code>	Field	<code>string</code>	Field containing the Object Name.

Batch Fields

Field	Type	Required	Description
<code>fixed-size</code> ✓	<code>number</code> (<code>integer</code>)		maximum number of events in an output batch. Examples: <code>42</code> , <code>1.2e-10</code>
<code>mode</code>	<code>Mode</code>	✓	If 'document' send on end of document generated by input. If 'fixed', use <code>fixed_size</code> . Allowed values: <code>fixed</code> , <code>document</code>
<code>timeout</code>	<code>time-interval</code> (<code>string</code>)	✓	interval after which the batch is sent, to keep throughput going (default 100ms). Default: <code>100ms</code> Examples: <code>500ms</code> , <code>2h</code>
<code>header</code>	<code>multiline-text</code> (<code>string</code>)		put a header line before the batch.
<code>footer</code>	<code>multiline-text</code> (<code>string</code>)		put a header line after the last line of the batch.
<code>use-document-marker</code> ✓	<code>boolean</code> (<code>bool</code>)		Enrich the job metadata with a document marker (for document handling in batch mode). Default: <code>false</code>
<code>wrap-as-json</code> ✓	<code>boolean</code> (<code>bool</code>)		Format the output batch as a JSON array. Default: <code>false</code>

Retry Fields

Field	Type	Required	Description
<code>timeout</code>	<code>time-interval</code> (<code>string</code>)	✓	timeout (e.g. 500ms, 2s etc. - default is 30). Examples: <code>500ms</code> , <code>2h</code>
<code>retries</code>	<code>number</code> (<code>integer</code>)		number of retries. Examples: <code>42</code> , <code>1.2e-10</code>

Mode Options

Value	Name	Description
-------	------	-------------

<code>put</code>	put	Put Objects
<code>delete</code>	delete	Delete Objects

Batch - Mode Options

Value	Name	Description
<code>fixed</code>	fixed	Fixed
<code>document</code>	document	Document

Output Preprocessor Options

Value	Name	Description
<code>gzip</code>	gzip	Gzip the output data
<code>parquet</code>	parquet	Extract the received data as JSON rows from a parquet file
<code>base64</code>	base64	Decode base64 as binary

Log Files

Reference for the Log Files component in LyftData's DSL

Source: `reference/dsl/outputs/file.md`

Log Files (`file`)

Append events to one or more files.

See [File Handling](#) for guidance on choosing between log files and block store outputs, plus `${}` naming patterns.

> Note: As of LyftData `2.0.2`, the `file` output flushes after each event regardless of `flush-at-end`.

File binary json raw

Minimal example

```
output:
file: path: ~
```

JSON

```
{
"output": { "file": { "path": null } } }
```

Contents

- [Minimal example](#)
- [Behavior](#)
- [Location](#)
- [Processing](#)
- [Reliability](#)
- [Schema](#)

Behavior

Behavior

Field	Type	Required	Description
<code>compress-after</code> ✓	<code>boolean</code> (<code>bool</code>)		Compress the written file after the filename changes. Default: <code>false</code>

<code>create-directories</code> ✓	<code>boolean</code> (<code>bool</code>)		Create parent directories for the given path if they do not exist. Default: <code>false</code>
<code>file-per-event</code> ✓	<code>boolean</code> (<code>bool</code>)		Each event is written out to a new file, overwriting if needed. Default: <code>false</code>
<code>truncate</code> ✓	<code>boolean</code> (<code>bool</code>)		Existing files are truncated before writing. Default: <code>false</code>

Location

Location

Field	Type	Required	Description
<code>path</code>	<code>string</code>	✓	Path to the file to be written. Can have <code>\${}</code> expansions.

Processing

Processing

Field	Type	Required	Description
<code>batch</code>	<code>Batch</code>		Batching input events together.
<code>input-field</code>	<code>field</code> (<code>string</code>)		Use the specified field as the content for the file line. Examples: <code>data_field</code>
<code>flush-at-end</code> ✓	<code>boolean</code> (<code>bool</code>)		Don't flush after each event. Default: <code>false</code>

Reliability

Reliability

Field	Type	Required	Description
<code>retry</code>	<code>Retry</code>		Retry options.

Schema

- [Batch Fields](#)
- [Retry Fields](#)
- [Batch - Mode Options](#)

Batch Fields

Field	Type	Required	Description
-------	------	----------	-------------

<code>fixed-size</code> ✓	<code>number</code> (<code>integer</code>)		maximum number of events in an output batch. Examples: <code>42</code> , <code>1.2e-10</code>
<code>mode</code>	<code>Mode</code>	✓	If 'document' send on end of document generated by input. If 'fixed', use <code>fixed_size</code> . Allowed values: <code>fixed</code> , <code>document</code>
<code>timeout</code>	<code>time-interval</code> (<code>string</code>)	✓	interval after which the batch is sent, to keep throughput going (default 100ms). Default: <code>100ms</code> Examples: <code>500ms</code> , <code>2h</code>
<code>header</code>	<code>multiline-text</code> (<code>string</code>)		put a header line before the batch.
<code>footer</code>	<code>multiline-text</code> (<code>string</code>)		put a header line after the last line of the batch.
<code>use-document-marker</code> ✓	<code>boolean</code> (<code>bool</code>)		Enrich the job metadata with a document marker (for document handling in batch mode). Default: <code>false</code>
<code>wrap-as-json</code> ✓	<code>boolean</code> (<code>bool</code>)		Format the output batch as a JSON array. Default: <code>false</code>

Retry Fields

Field	Type	Required	Description
<code>timeout</code>	<code>time-interval</code> (<code>string</code>)	✓	timeout (e.g. 500ms, 2s etc. - default is 30). Examples: <code>500ms</code> , <code>2h</code>
<code>retries</code>	<code>number</code> (<code>integer</code>)		number of retries. Examples: <code>42</code> , <code>1.2e-10</code>

Batch - Mode Options

Value	Name	Description
<code>fixed</code>	<code>fixed</code>	Fixed
<code>document</code>	<code>document</code>	Document

FTP

Reference for the FTP component in LyftData's DSL

Source: `reference/dsl/outputs/ftp.md`

FTP (`ftp`)

Upload events to FTP/SFTP/TFTP servers.

File Utility Enterprise edition binary json raw

Minimal example

```
output:
ftp: endpoint: host: ~ path: ~
```

JSON

```
{
"output": { "ftp": { "endpoint": { "host": null }, "path": null } } }
```

Contents

- [Minimal example](#)
- [Connection](#)
- [Reliability](#)
- [Transfer](#)
- [Schema](#)

Connection

Connection

Field	Type	Required	Description
<code>protocol</code>	Protocol		Transfer protocol to use (ftp, sftp, ftps, or tftp). Allowed values: <code>ftp</code> , <code>ftps-explicit</code> , <code>ftps-implicit</code> , <code>sftp</code> , <code>tftp</code>
<code>endpoint</code>	Endpoint	✓	Remote endpoint settings (host, port, TLS requirements).
<code>credentials</code>	FTP Credentials		Credentials or key material needed to authenticate with the server. Allowed values: <code>username-password</code> , <code>tls-client-cert</code> , <code>ssh-key</code> , <code>anonymous</code> , <code>none</code>

<code>working-directory</code>	<code>string</code>		Optional working directory to <code>CWD</code> into before writing files.
<code>passive</code> ✓	<code>boolean</code> (<code>bool</code>)		Enable passive FTP data connections when required by firewalls.
<code>ftps</code>	<code>Ftps</code>		FTPS-specific options (TLS certificates, modes) when protocol is FTPS.
<code>sftp</code>	<code>Sftp</code>		SFTP-specific options (key paths, algorithms) when using SSH-based transfer.
<code>tftp</code>	<code>Tftp</code>		TFTP-specific options such as <code>blksize</code> tuning.

Reliability

Reliability

Field	Type	Required	Description
<code>retry</code>	<code>Retry</code>		Retry policy to apply when uploads fail or time out.
<code>batch</code>	<code>Batch</code>		Batch multiple events into a single transfer when supported by the target.
<code>resume-checkpoint-variable</code>	<code>string</code>		Name of the variable used to persist resume checkpoints between runs.

Transfer

Transfer

Field	Type	Required	Description
<code>path</code>	<code>string</code>	✓	Remote path or prefix (may include variables) where files are written.
<code>metadata</code>	<code>Metadata</code>		Extra metadata columns to include with each upload event.
<code>concurrency</code>	<code>number</code> (<code>integer</code>)		Maximum number of files to transfer concurrently. Examples: <code>42</code> , <code>1.2e-10</code>
<code>tmp-suffix</code>	<code>string</code>		Temporary suffix applied to files while uploading; removed after completion.
<code>preprocessors</code>	[<code>Output Preprocessor</code> []] (#output-preprocessor-options)		Output preprocessors to run before writing data (for example compression). Allowed values: <code>gzip</code> , <code>parquet</code> , <code>base64</code>
<code>payload-mode</code> ✓	<code>Payload Mode</code>		Payload mode describing how bytes are emitted (auto/json/raw/binary). Allowed values: <code>auto</code> , <code>json</code> , <code>raw</code> , <code>binary</code>

<code>append</code> ✓	<code>boolean</code> (<code>bool</code>)	Append to existing remote files instead of overwriting them. Default: <code>false</code>
<code>checksum</code>	<code>Checksum</code>	Checksum verification strategy applied after a transfer completes.
<code>stream-buffer-size</code>	<code>number</code> (<code>integer</code>)	Size of the in-memory buffer used during streaming uploads. Examples: <code>42</code> , <code>1.2e-10</code>

Schema

- [FTP Credentials Options](#)
- [Sftp - SFTP Host Key Verification Options](#)
- [Endpoint Fields](#)
- [FTP Credentials - Username Password Fields](#)
- [FTP Credentials - Tls Client Cert Fields](#)
- [FTP Credentials - Ssh Key Fields](#)
- [Ftps Fields](#)
- [Sftp - SFTP Host Key Verification - Known Hosts File Fields](#)
- [Sftp - SFTP Host Key Verification - Fingerprint Fields](#)
- [Sftp Fields](#)
- [Tftp Fields](#)
- [Retry Fields](#)
- [Batch Fields](#)
- [Metadata Fields](#)
- [Checksum Fields](#)
- [Protocol Options](#)
- [Batch - Mode Options](#)
- [Output Preprocessor Options](#)
- [Payload Mode Options](#)
- [Checksum - Algorithm Options](#)

FTP Credentials Options

Option	Name	Type	Description
<code>username-password</code>	Username Password	<code>object</code>	
<code>tls-client-cert</code>	Tls Client Cert	<code>object</code>	
<code>ssh-key</code>	Ssh Key	<code>object</code>	
<code>anonymous</code>	Anonymous	<code>map</code>	
<code>none</code>	None	<code>map</code>	

Sftp - SFTP Host Key Verification Options

Option	Name	Type	Description
<code>known-hosts-file</code>	Known Hosts File	<code>object</code>	

<code>fingerprint</code>	Fingerprint	<code>object</code>	
<code>accept-new</code>	Accept New	<code>map</code>	

Endpoint Fields

Field	Type	Required	Description
<code>host</code>	<code>string</code>	<input checked="" type="checkbox"/>	
<code>port</code>	<code>number (integer)</code>		Examples: <code>42</code> , <code>1.2e-10</code>
<code>proxy</code>	<code>string</code>		
<code>bastion-host</code>	<code>string</code>		

FTP Credentials - Username Password Fields

Field	Type	Required	Description
<code>username-variable</code>	<code>string</code>	<input checked="" type="checkbox"/>	
<code>password-variable</code>	<code>string</code>	<input checked="" type="checkbox"/>	

FTP Credentials - Tls Client Cert Fields

Field	Type	Required	Description
<code>username-variable</code>	<code>string</code>	<input checked="" type="checkbox"/>	
<code>password-variable</code>	<code>string</code>	<input checked="" type="checkbox"/>	
<code>client-certificate-variable</code>	<code>string</code>	<input checked="" type="checkbox"/>	
<code>client-key-variable</code>	<code>string</code>	<input checked="" type="checkbox"/>	

FTP Credentials - Ssh Key Fields

Field	Type	Required	Description
<code>username-variable</code>	<code>string</code>	<input checked="" type="checkbox"/>	
<code>private-key-variable</code>	<code>string</code>	<input checked="" type="checkbox"/>	
<code>passphrase-variable</code>	<code>string</code>		

Ftps Fields

Field	Type	Required	Description
<code>ca-certificate-variable</code>	<code>string</code>		
<code>client-certificate-variable</code>	<code>string</code>		

<code>client-key-variable</code>	<code>string</code>		
<code>cipher-list</code>	<code>string</code>		
<code>verify-peer</code> ✓	<code>boolean (bool)</code>		Default: <code>false</code>
<code>min-tls-version</code>	<code>string</code>		

Sftp - SFTP Host Key Verification - Known Hosts File Fields

Field	Type	Required	Description
<code>path</code>	<code>path (string)</code>	✓	Examples: <code>/path/to/file</code> , <code>c:\users\joe\data\file.txt</code>

Sftp - SFTP Host Key Verification - Fingerprint Fields

Field	Type	Required	Description
<code>fingerprint</code>	<code>string</code>	✓	

Sftp Fields

Field	Type	Required	Description
<code>agent-forwarding</code> ✓	<code>boolean (bool)</code>		Default: <code>false</code>
<code>known-hosts-path</code>	<code>string</code>		
<code>host-key-verification</code>	SFTP Host Key Verification		Allowed values: <code>known-hosts-file</code> , <code>fingerprint</code> , <code>accept-new</code>
<code>jump-host</code>	<code>string</code>		

Tftp Fields

Field	Type	Required	Description
<code>block-size</code>	<code>number (integer)</code>		Examples: <code>42</code> , <code>1.2e-10</code>
<code>timeout-ms</code>	<code>number (integer)</code>		Examples: <code>42</code> , <code>1.2e-10</code>
<code>max-retries</code>	<code>number</code>		
<code>file-name</code>	<code>string</code>		

Retry Fields

Field	Type	Required	Description
<code>timeout</code>	<code>time-interval (string)</code>	✓	timeout (e.g. 500ms, 2s etc. - default is 30). Examples: <code>500ms</code> , <code>2h</code>
<code>retries</code>	<code>number (integer)</code>		number of retries. Examples: <code>42</code> , <code>1.2e-10</code>

Batch Fields

Field	Type	Required	Description
<code>fixed-size</code> ✓	<code>number</code> (<code>integer</code>)		maximum number of events in an output batch. Examples: <code>42</code> , <code>1.2e-10</code>
<code>mode</code>	<code>Mode</code>	✓	If 'document' send on end of document generated by input. If 'fixed', use <code>fixed_size</code> . Allowed values: <code>fixed</code> , <code>document</code>
<code>timeout</code>	<code>time-interval</code> (<code>string</code>)	✓	interval after which the batch is sent, to keep throughput going (default 100ms). Default: <code>100ms</code> Examples: <code>500ms</code> , <code>2h</code>
<code>header</code>	<code>multiline-text</code> (<code>string</code>)		put a header line before the batch.
<code>footer</code>	<code>multiline-text</code> (<code>string</code>)		put a header line after the last line of the batch.
<code>use-document-marker</code> ✓	<code>boolean</code> (<code>bool</code>)		Enrich the job metadata with a document marker (for document handling in batch mode). Default: <code>false</code>
<code>wrap-as-json</code> ✓	<code>boolean</code> (<code>bool</code>)		Format the output batch as a JSON array. Default: <code>false</code>

Metadata Fields

Field	Type	Required	Description
<code>object-name-field</code>	<code>field</code> (<code>string</code>)		Examples: <code>data_field</code>
<code>content-length-field</code>	<code>field</code> (<code>string</code>)		Examples: <code>data_field</code>
<code>last-modified-field</code>	<code>field</code> (<code>string</code>)		Examples: <code>data_field</code>
<code>checksum-field</code>	<code>field</code> (<code>string</code>)		Examples: <code>data_field</code>
<code>content-type-field</code>	<code>field</code> (<code>string</code>)		Examples: <code>data_field</code>
<code>etag-field</code>	<code>field</code> (<code>string</code>)		Examples: <code>data_field</code>

Checksum Fields

Field	Type	Required	Description
<code>verify-remote</code> ✓	<code>boolean</code> (<code>bool</code>)		Default: <code>false</code>
<code>algorithm</code>	<code>Algorithm</code>	✓	Allowed values: <code>md5</code> , <code>sha256</code>

Protocol Options

Value	Name	Description
<code>ftp</code>	<code>ftp</code>	Ftp
<code>ftps-explicit</code>	<code>ftps-explicit</code>	Ftps Explicit

<code>ftps-implicit</code>	ftps-implicit	Ftps Implicit
<code>sftp</code>	sftp	Sftp
<code>tftp</code>	tftp	Tftp

Batch - Mode Options

Value	Name	Description
<code>fixed</code>	fixed	Fixed
<code>document</code>	document	Document

Output Preprocessor Options

Value	Name	Description
<code>gzip</code>	gzip	Gzip the output data
<code>parquet</code>	parquet	Extract the received data as JSON rows from a parquet file
<code>base64</code>	base64	Decode base64 as binary

Payload Mode Options

Value	Name	Description
<code>auto</code>	auto	Auto
<code>json</code>	json	Json
<code>raw</code>	raw	Raw
<code>binary</code>	binary	Binary

Checksum - Algorithm Options

Value	Name	Description
<code>md5</code>	md5	Md5
<code>sha256</code>	sha256	Sha256

Google Cloud Storage

Reference for the Google Cloud Storage component in LyftData's DSL

Source: <reference/dsl/outputs/gcs.md>

Google Cloud Storage (`gcs`)

Write events to a Google Cloud bucket.

Block Store Cloud Enterprise edition binary json raw

Minimal example

```
output:
gcs: bucket-name: ~ credentials: storage-url: ~ object-name: name: ~
```

JSON

```
{
  "output": { "gcs": { "bucket-name": null, "credentials": { "storage-url": null }, "object-name": { "name": null } } } }
```

Contents

- [Minimal example](#)
- [Authentication](#)
- [Behavior](#)
- [Location](#)
- [Object Properties](#)
- [Processing](#)
- [Reliability](#)
- [Schema](#)

Authentication

Authentication

Field	Type	Required	Description
<code>credentials</code>	<code>Credentials</code>	<input checked="" type="checkbox"/>	Credentials for GCP. Supports service account JSON, application-default credentials, or context-provided material via <code>PossibleContextString</code> . Allowed values: <code>storage-url</code> , <code>service-account</code> , <code>application-credentials</code>

Behavior

Behavior

Field	Type	Required	Description
<code>mode</code>	<code>Mode</code>		Put or delete the object? <code>Put</code> uploads content (default), while <code>Delete</code> removes the named object. Allowed values: <code>put</code> , <code>delete</code>

Location

Location

Field	Type	Required	Description
<code>bucket-name</code>	<code>string</code>	✓	Bucket Name.
<code>object-name</code>	<code>ObjectName</code>	✓	Remote object name or field reference (via <code>ObjectDestination</code>). Supports folder-style paths using <code>/</code> separators. Allowed values: <code>name</code> , <code>field</code>

Object Properties

Object Properties

Field	Type	Required	Description
<code>disable-object-name-guid</code> ✓	<code>boolean</code> (<code>bool</code>)		Disable the autogenerated GUID prefix so the provided object name is used verbatim. This is forced off for delete operations. Default: <code>false</code>
<code>guid-prefix</code>	<code>string</code>		Optional GUID prefix appended ahead of the generated identifier. Defaults to <code>/</code> .
<code>guid-suffix</code>	<code>string</code>		Optional GUID suffix appended after the generated identifier.

Processing

Processing

Field	Type	Required	Description
<code>batch</code>	<code>Batch</code>		Configure batching before uploads; batching is recommended when writing many small events to reduce API calls.
<code>input-field</code>	<code>field</code> (<code>string</code>)		Use the specified field as the object payload. When omitted the entire event body is serialized according to the selected preprocessors. Examples: <code>data_field</code>

preprocessors	[Output Preprocessor[]] (#output-preprocessor-options)	Preprocessors (process data before making it available for upload) these processors will be run in the order they are specified. Allowed values: <code>gzip</code> , <code>parquet</code> , <code>base64</code>
---------------	---	---

Reliability

Reliability

Field	Type	Required	Description
retry	Retry		How to retry operation if it fails.
track-schema ✓	boolean (bool)		Validate JSON payloads against the stored schema and maintain <code>__SCHEMA_NUMBER</code> metadata. Only available when the payload serializes to JSON. Default: <code>false</code>

Schema

- [Credentials - Service Account Options](#)
- [Credentials - Application Credentials Options](#)
- [Credentials Options](#)
- [Object Name Options](#)
- [Batch Fields](#)
- [Retry Fields](#)
- [Mode Options](#)
- [Batch - Mode Options](#)
- [Output Preprocessor Options](#)

Credentials - Service Account Options

Option	Name	Type	Description
path	Path	string	Service Account File location. Examples: <code>/path/to/file</code> , <code>c:\users\joe\data\file.txt</code>
key	Key	string	

Credentials - Application Credentials Options

Option	Name	Type	Description
path	Path	string	Application Credential File Location. Examples: <code>/path/to/file</code> , <code>c:\users\joe\data\file.txt</code>
key	Key	string	Application Credential in JSON format.

Credentials Options

Option	Name	Type	Description
<code>storage-url</code>	Storage Url	<code>string</code>	URL in the form <code>gs://<bucket>/path</code> .
<code>service-account</code>	Service Account	<code>one_of</code>	Service Account credentials. Allowed values: <code>path</code> , <code>key</code>
<code>application-credentials</code>	Application Credentials	<code>one_of</code>	Application credentials. Allowed values: <code>path</code> , <code>key</code>

Object Name Options

Option	Name	Type	Description
<code>name</code>	Name	<code>string</code>	Object Name.
<code>field</code>	Field	<code>string</code>	Field containing the Object Name.

Batch Fields

Field	Type	Required	Description
<code>fixed-size</code> ✓	<code>number</code> (<code>integer</code>)		maximum number of events in an output batch. Examples: <code>42</code> , <code>1.2e-10</code>
<code>mode</code>	<code>Mode</code>	✓	If 'document' send on end of document generated by input. If 'fixed', use <code>fixed_size</code> . Allowed values: <code>fixed</code> , <code>document</code>
<code>timeout</code>	<code>time-interval</code> (<code>string</code>)	✓	interval after which the batch is sent, to keep throughput going (default 100ms). Default: <code>100ms</code> Examples: <code>500ms</code> , <code>2h</code>
<code>header</code>	<code>multiline-text</code> (<code>string</code>)		put a header line before the batch.
<code>footer</code>	<code>multiline-text</code> (<code>string</code>)		put a header line after the last line of the batch.
<code>use-document-marker</code> ✓	<code>boolean</code> (<code>bool</code>)		Enrich the job metadata with a document marker (for document handling in batch mode). Default: <code>false</code>
<code>wrap-as-json</code> ✓	<code>boolean</code> (<code>bool</code>)		Format the output batch as a JSON array. Default: <code>false</code>

Retry Fields

Field	Type	Required	Description
<code>timeout</code>	<code>time-interval</code> (<code>string</code>)	✓	timeout (e.g. 500ms, 2s etc. - default is 30). Examples: <code>500ms</code> , <code>2h</code>
<code>retries</code>	<code>number</code> (<code>integer</code>)		number of retries. Examples: <code>42</code> , <code>1.2e-10</code>

Mode Options

Value	Name	Description
<code>put</code>	put	Put Objects
<code>delete</code>	delete	Delete Objects

Batch - Mode Options

Value	Name	Description
<code>fixed</code>	fixed	Fixed
<code>document</code>	document	Document

Output Preprocessor Options

Value	Name	Description
<code>gzip</code>	gzip	Gzip the output data
<code>parquet</code>	parquet	Extract the received data as JSON rows from a parquet file
<code>base64</code>	base64	Decode base64 as binary

HTTP Get

Reference for the HTTP Get component in LyftData's DSL

Source: <reference/dsl/outputs/http-get.md>

HTTP Get (`http-get`)

Send event data to a remote server using GET.

Http json

Minimal example

```
output:
http-get: url: ""
```

JSON

```
{
"output": { "http-get": { "url": "" } } }
```

Contents

- [Minimal example](#)
- [Payload](#)
- [Processing](#)
- [Reliability](#)
- [Request](#)
- [Security](#)
- [Schema](#)

Payload

Payload

Field	Type	Required	Description
<code>body</code>	<code>Body</code>		How to construct the request body (event, field, or literal). Allowed values: <code>event</code> , <code>field</code> , <code>literal</code>

Processing

Processing

Field	Type	Required	Description
<code>batch</code>	<code>Batch</code>		Collect input events together into batches.

Reliability

Reliability

Field	Type	Required	Description
<code>retry</code>	<code>Retry</code>		How to retry failed requests.

Request

Request

Field	Type	Required	Description
<code>url</code>	<code>url</code> (<code>string</code>)	<input checked="" type="checkbox"/>	The URL may contain <code>\${}</code> field expansions. Examples: <code>https://example.com/path</code>
<code>headers</code>	<code>map</code> (<code>string</code>)		HTTP headers of the request.
<code>idempotency-key-field</code>	<code>field</code> (<code>string</code>)		Field providing the value for the Idempotency-Key header when supported by the sink. Examples: <code>data_field</code>

Security

Security

Field	Type	Required	Description
<code>insecure</code> <input checked="" type="checkbox"/>	<code>boolean</code> (<code>bool</code>)		Ignore TLS certificate validation errors (This is unsafe to use). Default: <code>false</code>

Schema

- [Body Options](#)
- [Body - Field Fields](#)
- [Body - Literal Fields](#)
- [Batch Fields](#)
- [Retry Fields](#)
- [Headers Table](#)
- [Batch - Mode Options](#)

Body Options

Option	Name	Type	Description
<code>event</code>	Event	<code>map</code>	
<code>field</code>	Field	<code>object</code>	
<code>literal</code>	Literal	<code>object</code>	

Body - Field Fields

Field	Type	Required	Description
<code>field</code>	<code>string</code>	✓	

Body - Literal Fields

Field	Type	Required	Description
<code>value</code>	<code>string</code>	✓	Literal/template request body; supports <code>\${}</code> expansions and multiline YAML strings.

Batch Fields

Field	Type	Required	Description
<code>fixed-size</code> ✓	<code>number</code> (<code>integer</code>)		maximum number of events in an output batch. Examples: <code>42</code> , <code>1.2e-10</code>
<code>mode</code>	<code>Mode</code>	✓	If 'document' send on end of document generated by input. If 'fixed', use <code>fixed_size</code> . Allowed values: <code>fixed</code> , <code>document</code>
<code>timeout</code>	<code>time-interval</code> (<code>string</code>)	✓	interval after which the batch is sent, to keep throughput going (default 100ms). Default: <code>100ms</code> Examples: <code>500ms</code> , <code>2h</code>
<code>header</code>	<code>multiline-text</code> (<code>string</code>)		put a header line before the batch.
<code>footer</code>	<code>multiline-text</code> (<code>string</code>)		put a header line after the last line of the batch.
<code>use-document-marker</code> ✓	<code>boolean</code> (<code>bool</code>)		Enrich the job metadata with a document marker (for document handling in batch mode). Default: <code>false</code>
<code>wrap-as-json</code> ✓	<code>boolean</code> (<code>bool</code>)		Format the output batch as a JSON array. Default: <code>false</code>

Retry Fields

Field	Type	Required	Description
<code>timeout</code>	<code>time-interval</code> (<code>string</code>)	✓	timeout (e.g. 500ms, 2s etc. - default is 30). Examples: <code>500ms</code> , <code>2h</code>

<code>retries</code>	<code>number (integer)</code>		number of retries. Examples: <code>42</code> , <code>1.2e-10</code>
----------------------	-------------------------------	--	---

Headers Table

Header	Value
<code>Header-Name</code>	<code>value</code>

Value format: `templated-text`.

Batch - Mode Options

Value	Name	Description
<code>fixed</code>	fixed	Fixed
<code>document</code>	document	Document

HTTP Post

Reference for the HTTP Post component in LyftData's DSL

Source: <reference/dsl/outputs/http-post.md>

HTTP Post (`http-post`)

POST event data to an HTTPS server.

Http json

Minimal example

```
output:
http-post: url: ""
```

JSON

```
{
  "output": { "http-post": { "url": "" } } }
```

Contents

- [Minimal example](#)
- [Payload](#)
- [Processing](#)
- [Reliability](#)
- [Request](#)
- [Security](#)
- [Schema](#)

Payload

Payload

Field	Type	Required	Description
<code>body</code>	<code>Body</code>		How to construct the request body (event, field, or literal). Allowed values: <code>event</code> , <code>field</code> , <code>literal</code>

Processing

Processing

Field	Type	Required	Description
<code>batch</code>	<code>Batch</code>		Collect input events together into batches.

Reliability

Reliability

Field	Type	Required	Description
<code>retry</code>	<code>Retry</code>		How to retry failed requests.

Request

Request

Field	Type	Required	Description
<code>url</code>	<code>url</code> (<code>string</code>)	<input checked="" type="checkbox"/>	The URL may contain <code>\${}</code> field expansions. Examples: <code>https://example.com/path</code>
<code>headers</code>	<code>map</code> (<code>string</code>)		HTTP headers of the request.
<code>idempotency-key-field</code>	<code>field</code> (<code>string</code>)		Field providing the value for the Idempotency-Key header when supported by the sink. Examples: <code>data_field</code>
<code>method</code>	<code>Method</code>		HTTP method to use for the query (either 'get' or 'post'). Allowed values: <code>get</code> , <code>post</code> , <code>put</code> , <code>patch</code> , <code>delete</code> , <code>options</code> , <code>head</code>

Security

Security

Field	Type	Required	Description
<code>insecure</code> <input checked="" type="checkbox"/>	<code>boolean</code> (<code>bool</code>)		Ignore TLS certificate validation errors (This is unsafe to use). Default: <code>false</code>

Schema

- [Body Options](#)
- [Body - Field Fields](#)
- [Body - Literal Fields](#)
- [Batch Fields](#)
- [Retry Fields](#)
- [Headers Table](#)

- [Batch - Mode Options](#)
- [Method Options](#)

Body Options

Option	Name	Type	Description
<code>event</code>	Event	<code>map</code>	
<code>field</code>	Field	<code>object</code>	
<code>literal</code>	Literal	<code>object</code>	

Body - Field Fields

Field	Type	Required	Description
<code>field</code>	<code>string</code>	<input checked="" type="checkbox"/>	

Body - Literal Fields

Field	Type	Required	Description
<code>value</code>	<code>string</code>	<input checked="" type="checkbox"/>	Literal/template request body; supports <code>\${}</code> expansions and multiline YAML strings.

Batch Fields

Field	Type	Required	Description
<code>fixed-size</code> ✓	<code>number</code> (<code>integer</code>)		maximum number of events in an output batch. Examples: <code>42</code> , <code>1.2e-10</code>
<code>mode</code>	<code>Mode</code>	<input checked="" type="checkbox"/>	If 'document' send on end of document generated by input. If 'fixed', use <code>fixed_size</code> . Allowed values: <code>fixed</code> , <code>document</code>
<code>timeout</code>	<code>time-interval</code> (<code>string</code>)	<input checked="" type="checkbox"/>	interval after which the batch is sent, to keep throughput going (default 100ms). Default: <code>100ms</code> Examples: <code>500ms</code> , <code>2h</code>
<code>header</code>	<code>multiline-text</code> (<code>string</code>)		put a header line before the batch.
<code>footer</code>	<code>multiline-text</code> (<code>string</code>)		put a header line after the last line of the batch.
<code>use-document-marker</code> ✓	<code>boolean</code> (<code>bool</code>)		Enrich the job metadata with a document marker (for document handling in batch mode). Default: <code>false</code>
<code>wrap-as-json</code> ✓	<code>boolean</code> (<code>bool</code>)		Format the output batch as a JSON array. Default: <code>false</code>

Retry Fields

Field	Type	Required	Description
-------	------	----------	-------------

<code>timeout</code>	<code>time-interval</code> (string)	<input checked="" type="checkbox"/>	timeout (e.g. 500ms, 2s etc. - default is 30). Examples: <code>500ms</code> , <code>2h</code>
<code>retries</code>	<code>number</code> (integer)		number of retries. Examples: <code>42</code> , <code>1.2e-10</code>

Headers Table

Header	Value
<code>Header-Name</code>	<code>value</code>

Value format: `templated-text` .

Batch - Mode Options

Value	Name	Description
<code>fixed</code>	fixed	Fixed
<code>document</code>	document	Document

Method Options

Value	Name	Description
<code>get</code>	get	Get
<code>post</code>	post	Post
<code>put</code>	put	Put
<code>patch</code>	patch	Patch
<code>delete</code>	delete	Delete
<code>options</code>	options	Options
<code>head</code>	head	Head

Kafka

Reference for the Kafka component in LyftData's DSL

Source: <reference/dsl/outputs/kafka.md>

Kafka (`kafka`)

Publish events to Kafka/Redpanda clusters.

Messaging Enterprise edition binary json raw

Minimal example

```
output:  
kafka: bootstrap-servers: ""
```

JSON

```
{  
  "output": { "kafka": { "bootstrap-servers": "" } } }
```

Contents

- [Minimal example](#)
- [Advanced](#)
- [Batching](#)
- [Connection](#)
- [Partitioning](#)
- [Payload](#)
- [Performance](#)
- [Reliability](#)
- [Security](#)
- [Schema](#)

Advanced

Advanced

Field	Type	Required	Description
<code>global-options</code> ✓	<code>map</code> (<code>string</code>)		Additional librdkafka producer configuration.
<code>topic-options</code> ✓	<code>map</code> (<code>string</code>)		Topic level overrides forwarded to librdkafka.

Batching

Batching

Field	Type	Required	Description
<code>batch</code>	<code>Batch</code>		Logical batching hints shared with other outputs.

Connection

Connection

Field	Type	Required	Description
<code>bootstrap-servers</code>	<code>hostname (string)</code>	<input checked="" type="checkbox"/>	Comma separated bootstrap servers. Examples: <code>example.com</code> , <code>localhost</code>
<code>client-id</code>	<code>string</code>		Optional producer client identifier.

Partitioning

Partitioning

Field	Type	Required	Description
<code>partitioner</code>	<code>Kafka Partitioner</code>		Partitioning strategy for the producer. Allowed values: <code>round-robin</code> , <code>hash-key</code> , <code>explicit</code>
<code>key-field</code>	<code>field (string)</code>		Event field containing message key (used for partitioning). Examples: <code>data_field</code>
<code>key-pointer</code>	<code>json-pointer (string)</code>		JSON pointer to message key.
<code>key-expression</code>	<code>string</code>		Expression used to compute message key.
<code>partition-field</code>	<code>field (string)</code>		Event field providing explicit partition. Examples: <code>data_field</code>
<code>partition-pointer</code>	<code>json-pointer (string)</code>		JSON pointer providing explicit partition.
<code>partition-value</code> ✓	<code>string</code>		Static partition value when using explicit partitioner.

Payload

Payload

Field	Type	Required	Description
-------	------	----------	-------------

<code>topic</code>	<code>string</code>		Static topic name.
<code>topic-field</code>	<code>field (string)</code>		Event field containing the target topic. Examples: <code>data_field</code>
<code>topic-pointer</code>	<code>json-pointer (string)</code>		JSON pointer resolving to target topic.
<code>input-field</code>	<code>field (string)</code>		Optional event field to treat as payload instead of entire event. Examples: <code>data_field</code>
<code>headers</code> ✓	<code>map (string)</code>		Message headers to attach to each record.
<code>payload</code>	Kafka Payload Format		Encoding strategy for message payloads. Allowed values: <code>json</code> , <code>raw</code> , <code>avro-bridge</code>
<code>avro</code>	Schema Registry		Schema registry configuration required when <code>payload=avro-bridge</code> .

Performance

Performance

Field	Type	Required	Description
<code>linger-ms</code>	<code>number (integer)</code>		Linger time before flushing a batch (ms). Examples: <code>42</code> , <code>1.2e-10</code>
<code>message-timeout-ms</code>	<code>number (integer)</code>		Timeout waiting for delivery reports (ms). Examples: <code>42</code> , <code>1.2e-10</code>
<code>compression</code>	Kafka Compression		Compression codec used by the producer. Allowed values: <code>none</code> , <code>gzip</code> , <code>snappy</code> , <code>lz4</code> , <code>zstd</code>

Reliability

Reliability

Field	Type	Required	Description
<code>retry</code>	Retry		Retry policy for failed deliveries.
<code>acks</code>	Kafka Acknowledgements		Acknowledgement requirement. Allowed values: <code>none</code> , <code>leader</code> , <code>all</code>
<code>idempotent</code> ✓	<code>boolean (bool)</code>		Enable Kafka idempotent producer mode. Default: <code>false</code>
<code>max-in-flight</code>	<code>number (integer)</code>		Maximum in-flight requests per connection (relevant for idempotency). Examples: <code>42</code> , <code>1.2e-10</code>

Security

Security

Field	Type	Required	Description
<code>auth</code>	<code>Kafka Authentication</code>		Authentication and TLS configuration.

Schema

- [Batch Fields](#)
- [Schema Registry Fields](#)
- [Retry Fields](#)
- [Kafka Authentication - Kafka SASL Fields](#)
- [Kafka Authentication - Kafka TLS Fields](#)
- [Kafka Authentication Fields](#)
- [Global Options Table](#)
- [Topic Options Table](#)
- [Headers Table](#)
- [Kafka Authentication - Kafka SASL - Extensions Table](#)
- [Kafka Authentication - Config Table](#)
- [Batch - Mode Options](#)
- [Kafka Partitioner Options](#)
- [Kafka Payload Format Options](#)
- [Kafka Compression Options](#)
- [Kafka Acknowledgements Options](#)
- [Kafka Authentication - Kafka SASL - Kafka SASL Mechanism Options](#)

Batch Fields

Field	Type	Required	Description
<code>fixed-size</code> ✓	<code>number</code> (<code>integer</code>)		maximum number of events in an output batch. Examples: <code>42</code> , <code>1.2e-10</code>
<code>mode</code>	<code>Mode</code>	✓	If 'document' send on end of document generated by input. If 'fixed', use <code>fixed_size</code> . Allowed values: <code>fixed</code> , <code>document</code>
<code>timeout</code>	<code>time-interval</code> (<code>string</code>)	✓	interval after which the batch is sent, to keep throughput going (default 100ms). Default: <code>100ms</code> Examples: <code>500ms</code> , <code>2h</code>
<code>header</code>	<code>multiline-text</code> (<code>string</code>)		put a header line before the batch.
<code>footer</code>	<code>multiline-text</code> (<code>string</code>)		put a header line after the last line of the batch.
<code>use-document-marker</code> ✓	<code>boolean</code> (<code>bool</code>)		Enrich the job metadata with a document marker (for document handling in batch mode). Default: <code>false</code>
<code>wrap-as-json</code> ✓	<code>boolean</code> (<code>bool</code>)		Format the output batch as a JSON array. Default: <code>false</code>

Schema Registry Fields

Field	Type	Required	Description
<code>url</code>	<code>url</code> (<code>string</code>)	✓	Examples: <code>https://example.com/path</code>
<code>username</code>	<code>string</code>		
<code>password</code>	<code>string</code>		
<code>access-token</code>	<code>string</code>		
<code>subject-suffix</code>	<code>string</code>		Optional suffix appended to subject names (e.g. <code>-value</code>).
<code>use-latest</code> ✓	<code>boolean</code> (<code>bool</code>)		When true, fetch the latest compatible schema instead of an explicit version. Default: <code>false</code>

Retry Fields

Field	Type	Required	Description
<code>timeout</code>	<code>time-interval</code> (<code>string</code>)	✓	timeout (e.g. 500ms, 2s etc. - default is 30). Examples: <code>500ms</code> , <code>2h</code>
<code>retries</code>	<code>number</code> (<code>integer</code>)		number of retries. Examples: <code>42</code> , <code>1.2e-10</code>

Kafka Authentication - Kafka SASL Fields

Field	Type	Required	Description
<code>mechanism</code>	Kafka SASL Mechanism	✓	Allowed values: <code>plain</code> , <code>scram-sha256</code> , <code>scram-sha512</code> , <code>o-auth-bearer</code>
<code>username</code> ✓	<code>string</code>		
<code>password</code> ✓	<code>string</code>		
<code>token</code> ✓	<code>string</code>		
<code>token-command</code> ✓	<code>string</code>		
<code>extensions</code> ✓	<code>map</code> (<code>string</code>)		

Kafka Authentication - Kafka TLS Fields

Field	Type	Required	Description
<code>ca-certificate</code>	<code>string</code>		
<code>client-certificate</code>	<code>string</code>		
<code>client-key</code>	<code>string</code>		
<code>insecure-skip-verify</code> ✓	<code>boolean</code> (<code>bool</code>)		Default: <code>false</code>

Kafka Authentication Fields

Field	Type	Required	Description
<code>sasl</code>	<code>Kafka SASL</code>		
<code>tls</code>	<code>Kafka TLS</code>		
<code>config</code> ✓	<code>map (string)</code>		Additional librdkafka configuration forwarded verbatim.

Global Options Table

Option	Value
<code>option.name</code>	<code>value</code>

Value format: `templated-text` .

Topic Options Table

Topic option	Value
<code>option.name</code>	<code>value</code>

Value format: `templated-text` .

Headers Table

Header	Value
<code>Header-Name</code>	<code>value</code>

Value format: `templated-text` .

Kafka Authentication - Kafka SASL - Extensions Table

Extension	Value
<code>key</code>	<code>value</code>

Value format: `templated-text` .

Kafka Authentication - Config Table

Option	Value
<code>option.name</code>	<code>value</code>

Value format: `templated-text` .

Batch - Mode Options

Value	Name	Description
<code>fixed</code>	fixed	Fixed
<code>document</code>	document	Document

Kafka Partitioner Options

Value	Name	Description
<code>round-robin</code>	round-robin	Round Robin
<code>hash-key</code>	hash-key	Hash Key
<code>explicit</code>	explicit	Explicit

Kafka Payload Format Options

Value	Name	Description
<code>json</code>	json	Json
<code>raw</code>	raw	Raw
<code>avro-bridge</code>	avro-bridge	Avro Bridge

Kafka Compression Options

Value	Name	Description
<code>none</code>	none	None
<code>gzip</code>	gzip	Gzip
<code>snappy</code>	snappy	Snappy
<code>lz4</code>	lz4	Lz4
<code>zstd</code>	zstd	Zstd

Kafka Acknowledgements Options

Value	Name	Description
<code>none</code>	none	None
<code>leader</code>	leader	Leader
<code>all</code>	all	All

Kafka Authentication - Kafka SASL - Kafka SASL Mechanism Options

Value	Name	Description
-------	------	-------------

<code>plain</code>	plain	Plain
<code>scram-sha256</code>	scram-sha256	Scram Sha256
<code>scram-sha512</code>	scram-sha512	Scram Sha512
<code>o-auth-bearer</code>	o-auth-bearer	O Auth Bearer

Message

Reference for the Message component in LyftData's DSL

Source: [reference/dsl/outputs/message.md](#)

Message (`message`)

Create a message to the internal message subsystem.

Messaging json

Minimal example

```
output:
message: {}
```

JSON

```
{
"output": { "message": {} } }
```

Contents

- [Minimal example](#)
- [Behavior](#)
- [Payload](#)
- [Variables](#)

Behavior

Behavior

Field	Type	Required	Description
<code>tag</code>	<code>string</code>		A user-defined tag that's attached to the output message type for filtering.

Payload

Payload

Field	Type	Required	Description
-------	------	----------	-------------

<code>input-field</code>	<code>field</code> (<code>string</code>)		Send only the content of the specified field to the message subsystem. Examples: <code>data_field</code>
--------------------------	---	--	---

Variables

Variables

Field	Type	Required	Description
<code>set-variable-name</code>	<code>string</code>		Use the input to set the value of a variable.

MongoDB

Reference for the MongoDB component in LyftData's DSL

Source: `reference/dsl/outputs/mongodb.md`

MongoDB (`mongodb`)

Write events to MongoDB collections.

Database Enterprise edition json

Minimal example

```
output:
mongodb: collection: ~ database: ~ operation: insert-one: {} uri: ""
```

JSON

```
{
  "output": { "mongodb": { "collection": null, "database": null, "operation": { "insert-one":
    {} } }, "uri": "" } }
```

Contents

- [Minimal example](#)
- [Batching](#)
- [Connection](#)
- [Operation](#)
- [Reliability](#)
- [Schema](#)

Batching

Batching

Field	Type	Required	Description
<code>batch</code>	<code>Batch</code>		Optional output batching hints (used by <code>insert-many</code>).

Connection

Connection

Field	Type	Required	Description
<code>uri</code>	<code>url (string)</code>	✓	MongoDB connection URI. Examples: <code>https://example.com/path</code>
<code>database</code>	<code>string</code>	✓	Database name.
<code>collection</code>	<code>string</code>	✓	Collection name.

Operation

Operation

Field	Type	Required	Description
<code>operation</code>	<code>Operation</code>	✓	Write operation to execute. Allowed values: <code>insert-one</code> , <code>insert-many</code> , <code>replace-one</code> , <code>update-one</code>

Reliability

Reliability

Field	Type	Required	Description
<code>retry</code>	<code>Retry</code>		Retry policy for failed writes.

Schema

- [Operation Options](#)
- [Batch Fields](#)
- [Operation - Insert One Fields](#)
- [Operation - Insert Many Fields](#)
- [Operation - Replace One Fields](#)
- [Operation - Update One Fields](#)
- [Retry Fields](#)
- [Batch - Mode Options](#)

Operation Options

Option	Name	Type	Description
<code>insert-one</code>	Insert One	<code>object</code>	Insert a single document.
<code>insert-many</code>	Insert Many	<code>object</code>	Insert one or many documents. When the selected value is an array, each element is inserted.
<code>replace-one</code>	Replace One	<code>object</code>	Replace a single matched document.
<code>update-one</code>	Update One	<code>object</code>	Update a single matched document.

Batch Fields

Field	Type	Required	Description
<code>fixed-size</code> ✓	number (integer)		maximum number of events in an output batch. Examples: <code>42</code> , <code>1.2e-10</code>
<code>mode</code>	Mode	✓	If 'document' send on end of document generated by input. If 'fixed', use <code>fixed_size</code> . Allowed values: <code>fixed</code> , <code>document</code>
<code>timeout</code>	time-interval (string)	✓	interval after which the batch is sent, to keep throughput going (default 100ms). Default: <code>100ms</code> Examples: <code>500ms</code> , <code>2h</code>
<code>header</code>	multiline-text (string)		put a header line before the batch.
<code>footer</code>	multiline-text (string)		put a header line after the last line of the batch.
<code>use-document-marker</code> ✓	boolean (bool)		Enrich the job metadata with a document marker (for document handling in batch mode). Default: <code>false</code>
<code>wrap-as-json</code> ✓	boolean (bool)		Format the output batch as a JSON array. Default: <code>false</code>

Operation - Insert One Fields

Field	Type	Required	Description
<code>document-field</code>	field (string)		Optional event field containing the document to insert. Defaults to the entire event. Examples: <code>data_field</code>

Operation - Insert Many Fields

Field	Type	Required	Description
<code>document-field</code>	field (string)		Optional event field containing the document(s) to insert. Defaults to the entire event. Examples: <code>data_field</code>

Operation - Replace One Fields

Field	Type	Required	Description
<code>filter-field</code>	field (string)	✓	Event field that resolves to the filter document. Examples: <code>data_field</code>
<code>replacement-field</code>	field (string)		Event field containing replacement document. Defaults to the entire event. Examples: <code>data_field</code>
<code>upsert</code> ✓	boolean (bool)		Create the document when no match exists. Default: <code>false</code>

Operation - Update One Fields

Field	Type	Required	Description
<code>filter-field</code>	<code>field</code> (<code>string</code>)	<input checked="" type="checkbox"/>	Event field that resolves to the filter document. Examples: <code>data_field</code>
<code>update-field</code>	<code>field</code> (<code>string</code>)		Event field containing update document. Defaults to the entire event. Examples: <code>data_field</code>
<code>upsert</code> ✓	<code>boolean</code> (<code>bool</code>)		Create the document when no match exists. Default: <code>false</code>

Retry Fields

Field	Type	Required	Description
<code>timeout</code>	<code>time-interval</code> (<code>string</code>)	<input checked="" type="checkbox"/>	timeout (e.g. 500ms, 2s etc. - default is 30). Examples: <code>500ms</code> , <code>2h</code>
<code>retries</code>	<code>number</code> (<code>integer</code>)		number of retries. Examples: <code>42</code> , <code>1.2e-10</code>

Batch - Mode Options

Value	Name	Description
<code>fixed</code>	<code>fixed</code>	Fixed
<code>document</code>	<code>document</code>	Document

MQTT

Reference for the MQTT component in LyftData's DSL

Source: <reference/dsl/outputs/mqtt.md>

MQTT (`mqtt`)

Publish events to external MQTT brokers or an embedded broker instance.

Messaging Enterprise edition binary json raw

Minimal example

```
output:
mqtt: broker: {} client: endpoint: ""
```

JSON

```
{
  "output": { "mqtt": { "broker": {}, "client": { "endpoint": "" } } } }
```

Contents

- [Minimal example](#)
- [Fields](#)
- [Schema](#)

Fields

Field	Type	Required	Description
<code>client</code>	MQTT Client Publisher	✓	Publish to an external MQTT broker.
<code>broker</code>	MQTT Embedded Broker Publisher	✓	Publish into an embedded broker that serves downstream subscribers.

Schema

- [MQTT Client Publisher - Retry Fields](#)
- [MQTT Client Publisher - MQTT TLS Fields](#)
- [MQTT Client Publisher - MQTT Dead Letter Fields](#)
- [MQTT Client Publisher Fields](#)
- [MQTT Embedded Broker Publisher - Retry Fields](#)

- [MQTT Embedded Broker Publisher - Listener Fields](#)
- [MQTT Embedded Broker Publisher Fields](#)
- [MQTT Client Publisher - MQTT Protocol Options](#)
- [MQTT Client Publisher - Payload Mode Options](#)
- [MQTT Embedded Broker Publisher - Payload Mode Options](#)

MQTT Client Publisher - Retry Fields

Field	Type	Required	Description
<code>timeout</code>	<code>time-interval</code> (<code>string</code>)	<input checked="" type="checkbox"/>	timeout (e.g. 500ms, 2s etc. - default is 30). Examples: <code>500ms</code> , <code>2h</code>
<code>retries</code>	<code>number</code> (<code>integer</code>)		number of retries. Examples: <code>42</code> , <code>1.2e-10</code>

MQTT Client Publisher - MQTT TLS Fields

Field	Type	Required	Description
<code>ca-certificate</code>	<code>string</code>		Optional CA certificate (PEM encoded).
<code>client-certificate</code>	<code>string</code>		Optional client certificate (PEM encoded).
<code>client-key</code>	<code>string</code>		Optional client private key (PEM encoded).

MQTT Client Publisher - MQTT Dead Letter Fields

Field	Type	Required	Description
<code>topic</code>	<code>string</code>	<input checked="" type="checkbox"/>	Topic to publish failed messages to once retries are exhausted.
<code>qos</code>	<code>number</code>		QoS level for dead-letter publishes (0 or 1). Default: <code>0</code>
<code>retain</code> ✓	<code>boolean</code> (<code>bool</code>)		Whether the dead-letter publish should retain the message. Default: <code>false</code>

MQTT Client Publisher Fields

Field	Type	Required	Description
<code>retry</code>	<code>Retry</code>		Retry policy for publish failures.
<code>endpoint</code>	<code>hostname</code> (<code>string</code>)	<input checked="" type="checkbox"/>	Broker endpoint URI (e.g. <code>mqtt://broker:1883</code> or <code>mqtt://broker:8883</code>). Examples: <code>example.com</code> , <code>localhost</code>
<code>protocol</code>	<code>MQTT Protocol</code>		MQTT protocol version used for publishing. Allowed values: <code>v3</code> , <code>v5</code>
<code>client-id</code>	<code>string</code>		Optional client identifier override.
<code>keep-alive-secs</code>	<code>number</code> (<code>integer</code>)		Keep-alive interval in seconds. Default: <code>30</code> Examples: <code>42</code> , <code>1.2e-10</code>

<code>clean-session</code> ✓	<code>boolean</code> (<code>bool</code>)	Whether to request a clean session on connect. Default: <code>true</code>
<code>username</code>	<code>string</code>	Optional username for broker authentication.
<code>password</code>	<code>string</code>	Optional password for broker authentication.
<code>tls</code>	MQTT TLS	TLS material for secure connections.
<code>request-channel-capacity</code>	<code>number</code> (<code>integer</code>)	Request channel capacity for the client worker. Default: <code>10</code> Examples: <code>42</code> , <code>1.2e-10</code>
<code>inflight</code>	<code>number</code> (<code>integer</code>)	Maximum inflight window for QoS1 publishes. Default: <code>100</code> Examples: <code>42</code> , <code>1.2e-10</code>
<code>qos</code>	<code>number</code>	Publish QoS level (0 or 1). Default: <code>0</code>
<code>retain</code> ✓	<code>boolean</code> (<code>bool</code>)	Whether publishes should set the retain flag. Default: <code>false</code>
<code>topic</code>	<code>string</code>	Static topic to publish to when dynamic selectors are absent.
<code>topic-field</code>	<code>field</code> (<code>string</code>)	Event field to resolve the topic from at runtime. Examples: <code>data_field</code>
<code>topic-pointer</code>	<code>json-pointer</code> (<code>string</code>)	JSON pointer to resolve the topic from at runtime.
<code>payload-mode</code>	Payload Mode	Controls how event payloads are encoded prior to publish. Default: <code>auto</code> Allowed values: <code>auto</code> , <code>json</code> , <code>raw</code> , <code>binary</code>
<code>pool-size</code>	<code>number</code> (<code>integer</code>)	Number of MQTT client connections maintained for parallel publishing. Default: <code>1</code> Examples: <code>42</code> , <code>1.2e-10</code>
<code>response-topic</code>	<code>string</code>	Optional response topic for MQTT v5 publishes.
<code>correlation-data</code>	<code>string</code>	Optional correlation data (base64 encoded) for MQTT v5 publishes.
<code>message-expiry-interval-secs</code>	<code>number</code> (<code>integer</code>)	Optional message expiry interval (seconds) for MQTT v5 publishes. Examples: <code>42</code> , <code>1.2e-10</code>
<code>dead-letter</code>	MQTT Dead Letter	Dead-letter behaviour when retries are exhausted.

MQTT Embedded Broker Publisher - Retry Fields

Field	Type	Required	Description
<code>timeout</code>	<code>time-interval</code> (<code>string</code>)	✓	timeout (e.g. 500ms, 2s etc. - default is 30). Examples: <code>500ms</code> , <code>2h</code>
<code>retries</code>	<code>number</code> (<code>integer</code>)		number of retries. Examples: <code>42</code> , <code>1.2e-10</code>

MQTT Embedded Broker Publisher - Listener Fields

Field	Type	Required	Description
<code>bind</code>	<code>hostname</code> (<code>string</code>)	<input checked="" type="checkbox"/>	Socket address in host:port form for the embedded broker listener. Examples: <code>example.com</code> , <code>localhost</code>
<code>tls</code> ✓	<code>boolean</code> (<code>bool</code>)		Whether TLS should be enabled for this listener (placeholder). Default: <code>false</code>

MQTT Embedded Broker Publisher Fields

Field	Type	Required	Description
<code>retry</code>	<code>Retry</code>		Retry policy applied before surfacing broker publish failures.
<code>qos</code>	<code>number</code>		Publish QoS level (0 or 1). Default: <code>0</code>
<code>retain</code> ✓	<code>boolean</code> (<code>bool</code>)		Whether embedded broker publishes should set the retain flag. Default: <code>false</code>
<code>topic</code>	<code>string</code>		Static topic used when dynamic selectors are absent.
<code>topic-field</code>	<code>field</code> (<code>string</code>)		Event field to resolve the topic from. Examples: <code>data_field</code>
<code>topic-pointer</code>	<code>json-pointer</code> (<code>string</code>)		JSON pointer to resolve the topic from.
<code>payload-mode</code>	<code>Payload Mode</code>		Controls how payloads are encoded prior to publish. Default: <code>auto</code> Allowed values: <code>auto</code> , <code>json</code> , <code>raw</code> , <code>binary</code>
<code>listeners</code>	[<code>Listener[]</code>](#mqtt-embedded-broker-publisher-listener-fields)		Listener surfaces to bind for the embedded broker.
<code>persistence-path</code>	<code>string</code>		Optional on-disk directory for broker persistence (defaults to job data dir).
<code>topic-prefix</code>	<code>string</code>		Optional prefix applied to topics before publishing into the embedded broker.
<code>worker-threads</code>	<code>number</code> (<code>integer</code>)		Number of worker threads dedicated to the embedded broker runtime. Default: <code>1</code> Examples: <code>42</code> , <code>1.2e-10</code>

MQTT Client Publisher - MQTT Protocol Options

Value	Name	Description
<code>v3</code>	<code>v3</code>	V3
<code>v5</code>	<code>v5</code>	V5

MQTT Client Publisher - Payload Mode Options

Value	Name	Description
<code>auto</code>	auto	Auto
<code>json</code>	json	Json
<code>raw</code>	raw	Raw
<code>binary</code>	binary	Binary

MQTT Embedded Broker Publisher - Payload Mode Options

Value	Name	Description
<code>auto</code>	auto	Auto
<code>json</code>	json	Json
<code>raw</code>	raw	Raw
<code>binary</code>	binary	Binary

MySQL

Reference for the MySQL component in LyftData's DSL

Source: <reference/dsl/outputs/mysql.md>

MySQL (mysql)

Execute parameterised statements against MySQL databases.

Database Enterprise edition json

Minimal example

```
output:
mysql: connection: url: url: ""
```

JSON

```
{
  "output": { "mysql": { "connection": { "url": { "url": "" } } } } }
```

Contents

- [Minimal example](#)
- [Connection](#)
- [Processing](#)
- [Reliability](#)
- [Result](#)
- [Statement](#)
- [Schema](#)

Connection

Connection

Field	Type	Required	Description
connection	Connection	✓	Connection details for the target database. Allowed values: url, parts

Processing

Processing

Field	Type	Required	Description
<code>batch</code>	<code>Batch</code>		Logical batching hints shared with other outputs.

Reliability

Reliability

Field	Type	Required	Description
<code>retry</code>	<code>Retry</code>		Retry policy for failed executions.

Result

Result

Field	Type	Required	Description
<code>affected-rows-field</code>	<code>field (string)</code>		Event field populated with affected row counts. Examples: <code>data_field</code>

Statement

Statement

Field	Type	Required	Description
<code>statement</code>	<code>multiline-text (string)</code>		Inline SQL statement executed when <code>statement-file</code> is not provided.
<code>statement-file</code>	<code>path (string)</code>		Path to a SQL file executed when <code>statement</code> is not provided. Examples: <code>/path/to/file</code> , <code>c:\users\joe\data\file.txt</code>
<code>parameters</code>	<code>[Parameters[]] (#parameters-fields)</code>		Bindings that map event fields into parameter placeholders.

Schema

- [Connection Options](#)
- [Connection - Url Fields](#)
- [Connection - Parts - Tuning Fields](#)
- [Connection - Parts Fields](#)
- [Batch Fields](#)
- [Retry Fields](#)
- [Parameters Fields](#)
- [Connection - Parts - Parameters Table](#)
- [Batch - Mode Options](#)

Connection Options

Option	Name	Type	Description
<code>url</code>	<code>Url</code>	<code>object</code>	
<code>parts</code>	<code>Parts</code>	<code>object</code>	

Connection - Url Fields

Field	Type	Required	Description
<code>url</code>	<code>url (string)</code>	<input checked="" type="checkbox"/>	Database connection URL (sqlx/ODBC compatible). Examples: <code>https://example.com/path</code>
<code>max-connections</code>	<code>number (integer)</code>		Maximum size of the async connection pool. Examples: <code>42</code> , <code>1.2e-10</code>
<code>min-connections</code>	<code>number (integer)</code>		Minimum number of connections retained in the pool. Examples: <code>42</code> , <code>1.2e-10</code>
<code>connect-timeout</code>	<code>string</code>		Optional connection timeout (humantime duration string).
<code>idle-timeout</code>	<code>string</code>		Optional idle timeout (humantime duration string).
<code>schema</code>	<code>string</code>		Optional schema override applied after connect.
<code>options</code>	<code>string[]</code>		Statements executed after connect (e.g. SET search_path TO ...).
<code>disable-tls-verification</code> ✓	<code>boolean (bool)</code>		Skip TLS verification (development only). Default: <code>false</code>

Connection - Parts - Tuning Fields

Field	Type	Required	Description
<code>max-connections</code>	<code>number (integer)</code>		Maximum size of the async connection pool. Examples: <code>42</code> , <code>1.2e-10</code>
<code>min-connections</code>	<code>number (integer)</code>		Minimum number of connections retained in the pool. Examples: <code>42</code> , <code>1.2e-10</code>
<code>connect-timeout</code>	<code>string</code>		Optional connection timeout (humantime duration string).
<code>idle-timeout</code>	<code>string</code>		Optional idle timeout (humantime duration string).
<code>schema</code>	<code>string</code>		Optional schema override applied after connect.
<code>options</code>	<code>string[]</code>		Statements executed after connect (e.g. SET search_path TO ...).
<code>disable-tls-verification</code> ✓	<code>boolean (bool)</code>		Skip TLS verification (development only). Default: <code>false</code>

Connection - Parts Fields

Field	Type	Required	Description
tuning	Tuning	✓	
host	string	✓	Hostname or IP address of the database server.
port	number (integer)		Optional TCP port (defaults to the backend default). Examples: 42 , 1.2e-10
database	string		Optional database name (mapped to the URL path segment).
username	string		Optional username (mapped to the URL userinfo).
password	string		Optional password (mapped to the URL userinfo).
parameters	map (string)		Additional query parameters appended to the generated URL.

Batch Fields

Field	Type	Required	Description
fixed-size ✓	number (integer)		maximum number of events in an output batch. Examples: 42 , 1.2e-10
mode	Mode	✓	If 'document' send on end of document generated by input. If 'fixed', use fixed_size . Allowed values: fixed , document
timeout	time-interval (string)	✓	interval after which the batch is sent, to keep throughput going (default 100ms). Default: 100ms Examples: 500ms , 2h
header	multiline-text (string)		put a header line before the batch.
footer	multiline-text (string)		put a header line after the last line of the batch.
use-document-marker ✓	boolean (bool)		Enrich the job metadata with a document marker (for document handling in batch mode). Default: false
wrap-as-json ✓	boolean (bool)		Format the output batch as a JSON array. Default: false

Retry Fields

Field	Type	Required	Description
timeout	time-interval (string)	✓	timeout (e.g. 500ms, 2s etc. - default is 30). Examples: 500ms , 2h
retries	number (integer)		number of retries. Examples: 42 , 1.2e-10

Parameters Fields

Field	Type	Required	Description
name	string	✓	

field	string	<input checked="" type="checkbox"/>	
default	multiline-text (string)		

Connection - Parts - Parameters Table

Parameter	Value
name	value

Value format: `templated-text`.

Batch - Mode Options

Value	Name	Description
fixed	fixed	Fixed
document	document	Document

OpenSearch

Reference for the OpenSearch component in LyftData's DSL

Source: [reference/dsl/outputs/open-search.md](#)

OpenSearch (`open-search`)

Bulk-index events into OpenSearch clusters, including AWS-managed variants.

Search Enterprise edition json

Minimal example

```
output:
open-search: common: {}
```

JSON

```
{
"output": { "open-search": { "common": {} } } }
```

Contents

- [Minimal example](#)
- [Fields](#)
- [Schema](#)

Fields

Field	Type	Required	Description
<code>common</code>	Elastic Output Spec	<input checked="" type="checkbox"/>	Shared OpenSearch connection and indexing options Shared configuration applied to all OpenSearch outputs.

Schema

- [Elastic Output Spec - Retry Fields](#)
- [Elastic Output Spec - Batch Fields](#)
- [Elastic Output Spec - Elastic Auth - Basic Fields](#)
- [Elastic Output Spec - Elastic Auth - Api Key Fields](#)
- [Elastic Output Spec - Elastic Auth - Tls Fields](#)
- [Elastic Output Spec - Elastic Auth Fields](#)

- [Elastic Output Spec - Elastic AWS SigV4 Fields](#)
- [Elastic Output Spec - Elastic Data Stream Fields](#)
- [Elastic Output Spec Fields](#)
- [Elastic Output Spec - Headers Table](#)
- [Elastic Output Spec - Query Table](#)
- [Elastic Output Spec - Mode Options](#)
- [Elastic Output Spec - Op Type Options](#)

Elastic Output Spec - Retry Fields

Field	Type	Required	Description
<code>timeout</code>	<code>time-interval</code> (<code>string</code>)	✓	timeout (e.g. 500ms, 2s etc. - default is 30). Examples: <code>500ms</code> , <code>2h</code>
<code>retries</code>	<code>number</code> (<code>integer</code>)		number of retries. Examples: <code>42</code> , <code>1.2e-10</code>

Elastic Output Spec - Batch Fields

Field	Type	Required	Description
<code>uuid-field</code>	<code>field</code> (<code>string</code>)		Field where generated uuid, the unique marker for the group, will be stored. Examples: <code>data_field</code>
<code>invocation-time-field</code>	<code>field</code> (<code>string</code>)		Field where invocation time will be stored. Examples: <code>data_field</code>
<code>completion-time-field</code>	<code>field</code> (<code>string</code>)		Field where completion (end of execution) time will be stored. Examples: <code>data_field</code>
<code>begin-marker-field</code>	<code>field</code> (<code>string</code>)		Field used to mark first event in the group. Examples: <code>data_field</code>
<code>end-marker-field</code>	<code>field</code> (<code>string</code>)		Field used to mark last event in the group. Examples: <code>data_field</code>
<code>line-count-field</code>	<code>field</code> (<code>string</code>)		Field used to store the line count of the batch. Examples: <code>data_field</code>
<code>line-num-field</code>	<code>field</code> (<code>string</code>)		Field used to store the line number of the batch. Examples: <code>data_field</code>

Elastic Output Spec - Elastic Auth - Basic Fields

Field	Type	Required	Description
<code>username</code> ✓	<code>string</code>		
<code>password</code> ✓	<code>string</code>		

Elastic Output Spec - Elastic Auth - Api Key Fields

Field	Type	Required	Description
-------	------	----------	-------------

id ✓	string		
key ✓	string		
base64 ✓	string		

Elastic Output Spec - Elastic Auth - Tls Fields

Field	Type	Required	Description
ca-certificate ✓	string		PEM encoded CA certificate bundle.
client-certificate ✓	string		PEM encoded client certificate for mutual TLS.
client-key ✓	string		PEM encoded private key for mutual TLS.
insecure-skip-verify ✓	boolean (bool)		Skip TLS certificate verification (development only). Default: false

Elastic Output Spec - Elastic Auth Fields

Field	Type	Required	Description
basic	Basic		
api-key	Api Key		
bearer-token ✓	string		
tls	Tls		

Elastic Output Spec - Elastic AWS SigV4 Fields

Field	Type	Required	Description
region ✓	string	✓	AWS region used for request signing.
service ✓	string		Override AWS service name (default es).
profile ✓	string		Optional shared credentials profile.
role-arn ✓	string		AssumeRole ARN for federated access.
session-name ✓	string		Session name when assuming roles.
access-key-id ✓	string		Access key used for signing requests.
secret-access-key ✓	string		Secret key used for signing requests.
session-token ✓	string		Optional session token for temporary credentials.

Elastic Output Spec - Elastic Data Stream Fields

Field	Type	Required	Description
-------	------	----------	-------------

<code>r-type</code> ✓	<code>string</code>	✓	Data stream type (logs, metrics, traces).
<code>dataset</code> ✓	<code>string</code>	✓	Data stream dataset (e.g. service.application).
<code>namespace</code> ✓	<code>string</code>	✓	Data stream namespace (e.g. prod, staging).

Elastic Output Spec Fields

Field	Type	Required	Description
<code>retry</code>	Retry		Retry policy for bulk failures.
<code>batch</code>	Batch		Logical batching hints shared with other outputs.
<code>endpoint</code> ✓	<code>url (string)</code>		HTTPS endpoint for the cluster (required when cloud-id is unset). Examples: <code>https://example.com/path</code>
<code>cloud-id</code> ✓	<code>string</code>		Elastic Cloud / Serverless Cloud ID (mutually exclusive with endpoint).
<code>region</code> ✓	<code>string</code>		Optional default region used for auth helpers and logging.
<code>auth</code>	Elastic Auth		Authentication strategies (basic, API keys, bearer).
<code>aws</code>	Elastic AWS SigV4		AWS SigV4 credentials for managed OpenSearch/Elasticsearch.
<code>mode</code>	Mode		Indexing lifecycle mode (classic index vs data stream). Allowed values: <code>classic-index</code> , <code>data-stream</code>
<code>index</code> ✓	<code>string</code>		Static index/alias target when mode=classic-index.
<code>index-field</code>	<code>field (string)</code>		Event field that provides the target index/alias. Examples: <code>data_field</code>
<code>index-pointer</code>	<code>json-pointer (string)</code>		JSON pointer resolving to the index/alias.
<code>data-stream</code>	Elastic Data Stream		Data stream configuration when mode=data-stream.
<code>document-id-field</code>	<code>field (string)</code>		Event field containing the document_id. Examples: <code>data_field</code>
<code>document-id-pointer</code>	<code>json-pointer (string)</code>		JSON pointer resolving to the document_id.
<code>document-id-expression</code>	<code>string</code>		Expression used to compute the document_id.
<code>routing-field</code>	<code>field (string)</code>		Event field providing custom routing key. Examples: <code>data_field</code>
<code>routing-pointer</code>	<code>json-pointer (string)</code>		JSON pointer providing custom routing key.
<code>routing-expression</code>	<code>string</code>		Expression computing the routing key.

<code>pipeline</code> ✓	<code>string</code>		Optional ingest pipeline to apply server-side.
<code>op-type</code>	<code>Op Type</code>		Bulk operation type for each event. Allowed values: <code>index</code> , <code>create</code> , <code>update</code> , <code>delete</code>
<code>retry-on-conflict</code>	<code>number</code> (<code>integer</code>)		Retry on conflict (for update/upsert operations). Examples: <code>42</code> , <code>1.2e-10</code>
<code>refresh</code> ✓	<code>boolean</code> (<code>bool</code>)		Force refresh after each batch. Default: <code>false</code>
<code>gzip</code> ✓	<code>boolean</code> (<code>bool</code>)		Compress bulk payloads with gzip. Default: <code>false</code>
<code>input-field</code>	<code>field</code> (<code>string</code>)		Event field containing the JSON document payload. Examples: <code>data_field</code>
<code>input-pointer</code>	<code>json-pointer</code> (<code>string</code>)		JSON pointer for extracting the payload.
<code>payload-template</code>	<code>string</code>		Template expression to render the payload when not using event body.
<code>headers</code> ✓	<code>map</code> (<code>string</code>)		Additional HTTP headers forwarded with each request.
<code>query</code> ✓	<code>map</code> (<code>string</code>)		Static query parameters appended to bulk requests.
<code>request-timeout-secs</code>	<code>number</code> (<code>integer</code>)		Request timeout in seconds. Examples: <code>42</code> , <code>1.2e-10</code>
<code>connect-timeout-secs</code>	<code>number</code> (<code>integer</code>)		Connect timeout in seconds. Examples: <code>42</code> , <code>1.2e-10</code>

Elastic Output Spec - Headers Table

Header	Value
<code>Header-Name</code>	<code>value</code>

Value format: `templated-text` .

Elastic Output Spec - Query Table

Parameter	Value
<code>param</code>	<code>value</code>

Value format: `templated-text` .

Elastic Output Spec - Mode Options

Value	Name	Description
<code>classic-index</code>	<code>classic-index</code>	Standard index or alias writes

<code>data-stream</code>	data-stream	Data stream aware writes (auto-selects stream based on dataset/namespace)
--------------------------	-------------	---

Elastic Output Spec - Op Type Options

Value	Name	Description
<code>index</code>	index	Index (create or replace) each document
<code>create</code>	create	Only create new documents; fail if <code>_id</code> exists
<code>update</code>	update	Update existing documents (requires partial document or script)
<code>delete</code>	delete	Delete documents (requires <code>_id</code>)

OpenTelemetry

Reference for the OpenTelemetry component in LyftData's DSL

Source: [reference/dsl/outputs/open-telemetry.md](#)

OpenTelemetry (`open-telemetry`)

Export events using the OpenTelemetry OTLP protocol.

Observability Enterprise edition json

Minimal example

```
output:
open-telemetry: connection: endpoint: ""
```

JSON

```
{
  "output": { "open-telemetry": { "connection": { "endpoint": "" } } } }
```

Contents

- [Minimal example](#)
- [Attributes](#)
- [Batching](#)
- [Connection](#)
- [Diagnostics](#)
- [Reliability](#)
- [Signals](#)
- [Schema](#)

Attributes

Attributes

Field	Type	Required	Description
<code>resource-attributes</code> ✓	<code>map (string)</code>		Static resource attributes to attach to exported signals.

Batching

Batching

Field	Type	Required	Description
<code>batch</code>	<code>Batch</code>		Logical batching hints shared with other outputs.

Connection

Connection

Field	Type	Required	Description
<code>connection</code>	<code>Connection</code>	<input checked="" type="checkbox"/>	gRPC client connection settings.

Diagnostics

Diagnostics

Field	Type	Required	Description
<code>exporter-tracing</code> ✓	<code>boolean</code> (<code>bool</code>)		Emit exporter-internal spans/logs to aid debugging collector connectivity. Only enable during troubleshooting; defaults to false. Default: <code>false</code>

Reliability

Reliability

Field	Type	Required	Description
<code>retry</code>	<code>Retry</code>		Retry policy applied to exporter failures.

Signals

Signals

Field	Type	Required	Description
<code>signals</code>	<code>[Signals[]]</code> (#signals-options)		Signals to render from events (default: logs, traces). Allowed values: <code>logs</code> , <code>traces</code> , <code>metrics</code>

Schema

- [Connection - Auth Options](#)
- [Batch Fields](#)
- [Connection - Auth - Basic Fields](#)
- [Connection - Auth - Bearer Fields](#)

- [Connection - OTEL Client TLS Fields](#)
- [Connection Fields](#)
- [Retry Fields](#)
- [Resource Attributes Table](#)
- [Connection - Headers Table](#)
- [Batch - Mode Options](#)
- [Signals Options](#)

Connection - Auth Options

Option	Name	Type	Description
<code>basic</code>	Basic	object	
<code>bearer</code>	Bearer	object	

Batch Fields

Field	Type	Required	Description
<code>fixed-size</code> ✓	number (integer)		maximum number of events in an output batch. Examples: <code>42</code> , <code>1.2e-10</code>
<code>mode</code>	Mode	✓	If 'document' send on end of document generated by input. If 'fixed', use <code>fixed_size</code> . Allowed values: <code>fixed</code> , <code>document</code>
<code>timeout</code>	time-interval (string)	✓	interval after which the batch is sent, to keep throughput going (default 100ms). Default: <code>100ms</code> Examples: <code>500ms</code> , <code>2h</code>
<code>header</code>	multiline-text (string)		put a header line before the batch.
<code>footer</code>	multiline-text (string)		put a header line after the last line of the batch.
<code>use-document-marker</code> ✓	boolean (bool)		Enrich the job metadata with a document marker (for document handling in batch mode). Default: <code>false</code>
<code>wrap-as-json</code> ✓	boolean (bool)		Format the output batch as a JSON array. Default: <code>false</code>

Connection - Auth - Basic Fields

Field	Type	Required	Description
<code>username</code> ✓	string	✓	
<code>password</code> ✓	string	✓	

Connection - Auth - Bearer Fields

Field	Type	Required	Description
-------	------	----------	-------------

token ✓	string	✓	
header-name	string		Optional custom header name, defaults to <code>authorization</code> .
scheme	string		Optional scheme override, defaults to <code>Bearer</code> .

Connection - OTel Client TLS Fields

Field	Type	Required	Description
ca-certificate ✓	path (string)		Examples: <code>/path/to/file</code> , <code>c:\users\joe\data\file.txt</code>
domain ✓	string		
client-certificate ✓	path (string)		Examples: <code>/path/to/file</code> , <code>c:\users\joe\data\file.txt</code>
client-key ✓	path (string)		Examples: <code>/path/to/file</code> , <code>c:\users\joe\data\file.txt</code>
insecure-skip-verify ✓	boolean (bool)		Default: <code>false</code>

Connection Fields

Field	Type	Required	Description
endpoint ✓	url (string)	✓	Examples: <code>https://example.com/path</code>
timeout	string		Timeout for export requests (e.g., <code>30s</code>).
headers ✓	map (string)		
auth	Auth		Allowed values: <code>basic</code> , <code>bearer</code>
tls	OTel Client TLS		

Retry Fields

Field	Type	Required	Description
timeout	time-interval (string)	✓	timeout (e.g. 500ms, 2s etc. - default is 30). Examples: <code>500ms</code> , <code>2h</code>
retries	number (integer)		number of retries. Examples: <code>42</code> , <code>1.2e-10</code>

Resource Attributes Table

Attribute	Value
service.name	value

Value format: `templated-text`.

Connection - Headers Table

Header	Value
<code>Header-Name</code>	<code>value</code>

Value format: `templated-text`.

Batch - Mode Options

Value	Name	Description
<code>fixed</code>	fixed	Fixed
<code>document</code>	document	Document

Signals Options

Value	Name	Description
<code>logs</code>	logs	Logs
<code>traces</code>	traces	Traces
<code>metrics</code>	metrics	Metrics

PostgreSQL

Reference for the PostgreSQL component in LyftData's DSL

Source: `reference/dsl/outputs/postgres.md`

PostgreSQL (`postgres`)

Execute parameterised statements against PostgreSQL databases.

Database Enterprise edition json

Minimal example

```
output:
postgres: connection: url: url: ""
```

JSON

```
{
"output": { "postgres": { "connection": { "url": { "url": "" } } } } }
```

Contents

- [Minimal example](#)
- [Connection](#)
- [Processing](#)
- [Reliability](#)
- [Result](#)
- [Statement](#)
- [Transaction](#)
- [Schema](#)

Connection

Connection

Field	Type	Required	Description
<code>connection</code>	<code>Connection</code>	✓	Connection details for the target database. Allowed values: <code>url</code> , <code>parts</code>

Processing

Processing

Field	Type	Required	Description
<code>batch</code>	<code>Batch</code>		Logical batching hints shared with other outputs.

Reliability

Reliability

Field	Type	Required	Description
<code>retry</code>	<code>Retry</code>		Retry policy for failed executions.

Result

Result

Field	Type	Required	Description
<code>result-field</code>	<code>field</code> (string)		Event field updated with statement scalar results. Examples: <code>data_field</code>
<code>affected-rows-field</code>	<code>field</code> (string)		Event field populated with affected row counts. Examples: <code>data_field</code>

Statement

Statement

Field	Type	Required	Description
<code>statement</code>	<code>multiline-text</code> (string)		Inline SQL statement executed when <code>statement-file</code> is not provided.
<code>statement-file</code>	<code>path</code> (string)		Path to a SQL file executed when <code>statement</code> is not provided. Examples: <code>/path/to/file</code> , <code>c:\users\joe\data\file.txt</code>
<code>parameters</code>	[Parameters[]] (#parameters-fields)		Bindings that map event fields into parameter placeholders.

Transaction

Transaction

Field	Type	Required	Description
<code>transaction</code>	<code>Transaction</code>		Optional transaction scope applied per execution.

Schema

- [Connection Options](#)
- [Connection - Url Fields](#)
- [Connection - Parts - Tuning Fields](#)
- [Connection - Parts Fields](#)
- [Batch Fields](#)
- [Retry Fields](#)
- [Parameters Fields](#)
- [Transaction Fields](#)
- [Connection - Parts - Parameters Table](#)
- [Batch - Mode Options](#)
- [Transaction - Mode Options](#)

Connection Options

Option	Name	Type	Description
<code>url</code>	Url	object	
<code>parts</code>	Parts	object	

Connection - Url Fields

Field	Type	Required	Description
<code>url</code>	<code>url (string)</code>	<input checked="" type="checkbox"/>	Database connection URL (sqlx/ODBC compatible). Examples: <code>https://example.com/path</code>
<code>max-connections</code>	<code>number (integer)</code>		Maximum size of the async connection pool. Examples: <code>42</code> , <code>1.2e-10</code>
<code>min-connections</code>	<code>number (integer)</code>		Minimum number of connections retained in the pool. Examples: <code>42</code> , <code>1.2e-10</code>
<code>connect-timeout</code>	<code>string</code>		Optional connection timeout (humantime duration string).
<code>idle-timeout</code>	<code>string</code>		Optional idle timeout (humantime duration string).
<code>schema</code>	<code>string</code>		Optional schema override applied after connect.
<code>options</code>	<code>string[]</code>		Statements executed after connect (e.g. SET search_path TO ...).
<code>disable-tls-verification</code> ✓	<code>boolean (bool)</code>		Skip TLS verification (development only). Default: <code>false</code>

Connection - Parts - Tuning Fields

Field	Type	Required	Description
<code>max-connections</code>	<code>number (integer)</code>		Maximum size of the async connection pool. Examples: <code>42</code> , <code>1.2e-10</code>

<code>min-connections</code>	<code>number</code> (<code>integer</code>)		Minimum number of connections retained in the pool. Examples: <code>42</code> , <code>1.2e-10</code>
<code>connect-timeout</code>	<code>string</code>		Optional connection timeout (humantime duration string).
<code>idle-timeout</code>	<code>string</code>		Optional idle timeout (humantime duration string).
<code>schema</code>	<code>string</code>		Optional schema override applied after connect.
<code>options</code>	<code>string[]</code>		Statements executed after connect (e.g. SET search_path TO ...).
<code>disable-tls-verification</code> ✓	<code>boolean</code> (<code>bool</code>)		Skip TLS verification (development only). Default: <code>false</code>

Connection - Parts Fields

Field	Type	Required	Description
<code>tuning</code>	<code>Tuning</code>	✓	
<code>host</code>	<code>string</code>	✓	Hostname or IP address of the database server.
<code>port</code>	<code>number</code> (<code>integer</code>)		Optional TCP port (defaults to the backend default). Examples: <code>42</code> , <code>1.2e-10</code>
<code>database</code>	<code>string</code>		Optional database name (mapped to the URL path segment).
<code>username</code>	<code>string</code>		Optional username (mapped to the URL userinfo).
<code>password</code>	<code>string</code>		Optional password (mapped to the URL userinfo).
<code>parameters</code>	<code>map</code> (<code>string</code>)		Additional query parameters appended to the generated URL.

Batch Fields

Field	Type	Required	Description
<code>fixed-size</code> ✓	<code>number</code> (<code>integer</code>)		maximum number of events in an output batch. Examples: <code>42</code> , <code>1.2e-10</code>
<code>mode</code>	<code>Mode</code>	✓	If 'document' send on end of document generated by input. If 'fixed', use <code>fixed_size</code> . Allowed values: <code>fixed</code> , <code>document</code>
<code>timeout</code>	<code>time-interval</code> (<code>string</code>)	✓	interval after which the batch is sent, to keep throughput going (default 100ms). Default: <code>100ms</code> Examples: <code>500ms</code> , <code>2h</code>
<code>header</code>	<code>multiline-text</code> (<code>string</code>)		put a header line before the batch.
<code>footer</code>	<code>multiline-text</code> (<code>string</code>)		put a header line after the last line of the batch.
<code>use-document-marker</code> ✓	<code>boolean</code> (<code>bool</code>)		Enrich the job metadata with a document marker (for document handling in batch mode). Default: <code>false</code>

<code>wrap-as-json</code> ✓	<code>boolean (bool)</code>	Format the output batch as a JSON array. Default: <code>false</code>
--------------------------------	-----------------------------	--

Retry Fields

Field	Type	Required	Description
<code>timeout</code>	<code>time-interval (string)</code>	✓	timeout (e.g. 500ms, 2s etc. - default is 30). Examples: <code>500ms</code> , <code>2h</code>
<code>retries</code>	<code>number (integer)</code>		number of retries. Examples: <code>42</code> , <code>1.2e-10</code>

Parameters Fields

Field	Type	Required	Description
<code>name</code>	<code>string</code>	✓	
<code>field</code>	<code>string</code>	✓	
<code>default</code>	<code>multiline-text (string)</code>		

Transaction Fields

Field	Type	Required	Description
<code>mode</code>	<code>Mode</code>		Allowed values: <code>none</code> , <code>per-event</code> , <code>per-batch</code> , <code>document</code>
<code>isolation-level</code>	<code>string</code>		
<code>access-mode</code>	<code>string</code>		
<code>deferrable</code>	<code>boolean (bool)</code>		

Connection - Parts - Parameters Table

Parameter	Value
<code>name</code>	<code>value</code>

Value format: `templated-text`.

Batch - Mode Options

Value	Name	Description
<code>fixed</code>	<code>fixed</code>	Fixed
<code>document</code>	<code>document</code>	Document

Transaction - Mode Options

Value	Name	Description
<code>none</code>	none	None
<code>per-event</code>	per-event	Per Event
<code>per-batch</code>	per-batch	Per Batch
<code>document</code>	document	Document

Print

Reference for the Print component in LyftData's DSL

Source: [reference/dsl/outputs/print.md](#)

Print (`print`)

Print event payloads to the terminal.

Debug json

Minimal example

```
output:
print: output: stdout
```

JSON

```
{
  "output": { "print": { "output": "stdout" } } }
```

Contents

- [Minimal example](#)
- [Behavior](#)
- [Schema](#)

Behavior

Behavior

Field	Type	Required	Description
<code>output</code> ✓	<code>Output</code>	✓	the kind of output: standard output, standard error, pretty output. Allowed values: <code>stdout</code> , <code>stderr</code> , <code>pretty-stdout</code> , <code>pretty-stderr</code>

Schema

- [Output Options](#)

Output Options

Value	Name	Description
<code>stdout</code>	stdout	Standard output of job
<code>stderr</code>	stderr	Standard error of job
<code>pretty-stdout</code>	pretty-stdout	Nice JSON to standard output
<code>pretty-stderr</code>	pretty-stderr	Nice JSON to standard error

S3

Reference for the S3 component in LyftData's DSL

Source: <reference/dsl/outputs/s3.md>

S3 (`s3`)

Write events to an S3 bucket.

Block Store Cloud Enterprise edition binary json raw

Minimal example

```
output:
s3: bucket-name: ~ object-name: name: ~
```

JSON

```
{
  "output": { "s3": { "bucket-name": null, "object-name": { "name": null } } } }
```

Contents

- [Minimal example](#)
- [Authentication](#)
- [Behavior](#)
- [Location](#)
- [Object Properties](#)
- [Processing](#)
- [Reliability](#)
- [Schema](#)

Authentication

Authentication

Field	Type	Required	Description
<code>access-key</code>	<code>string</code>		Access Key ID.
<code>secret-key</code>	<code>string</code>		Secret Key ID.
<code>security-token</code>	<code>string</code>		Security Token.

<code>session-token</code>	<code>string</code>		Session Token.
<code>role-arn</code>	<code>string</code>		A Role ARN for assuming role using above credentials.

Behavior

Behavior

Field	Type	Required	Description
<code>mode</code>	<code>Mode</code>		Put or delete the object? Default is Put. Allowed values: <code>put</code> , <code>delete</code>

Location

Location

Field	Type	Required	Description
<code>bucket-name</code>	<code>string</code>	<input checked="" type="checkbox"/>	Bucket Name.
<code>object-name</code>	<code>ObjectName</code>	<input checked="" type="checkbox"/>	Name of object, either as text or from a field. Allowed values: <code>name</code> , <code>field</code>
<code>region</code>	<code>string</code>		S3 Region.
<code>endpoint</code>	<code>string</code>		S3 Endpoint, if needed.

Object Properties

Object Properties

Field	Type	Required	Description
<code>disable-object-name-guid</code> ✓	<code>boolean</code> (<code>bool</code>)		Disable the GUID prefix if you want the object name to be treated literally (this is disabled for deletes). Default: <code>false</code>
<code>guid-prefix</code>	<code>string</code>		GUID Prefix, will be prepended to the GUID, the default value is "/".
<code>guid-suffix</code>	<code>string</code>		GUID Suffix, will be appended to the GUID if specified.

Processing

Processing

Field	Type	Required	Description
<code>batch</code>	<code>Batch</code>		Batching input events together.

<code>input-field</code>	<code>field (string)</code>		Use the specified field as the content for the object. Examples: <code>data_field</code>
<code>preprocessors</code>	<code>[Output Preprocessor[]]</code> (#output-preprocessor-options)		Preprocessors (process data before making it available for upload) these processors will be run in the order they are specified. Allowed values: <code>gzip</code> , <code>parquet</code> , <code>base64</code>

Reliability

Reliability

Field	Type	Required	Description
<code>retry</code>	<code>Retry</code>		How to retry failed requests.
<code>track-schema</code> ✓	<code>boolean (bool)</code>		Check the schema of the written data and update <code>__SCHEMA_NUMBER</code> (written data must be JSON). Default: <code>false</code>

Schema

- [Object Name Options](#)
- [Batch Fields](#)
- [Retry Fields](#)
- [Mode Options](#)
- [Batch - Mode Options](#)
- [Output Preprocessor Options](#)

Object Name Options

Option	Name	Type	Description
<code>name</code>	Name	<code>string</code>	Object Name.
<code>field</code>	Field	<code>string</code>	Field containing the Object Name.

Batch Fields

Field	Type	Required	Description
<code>fixed-size</code> ✓	<code>number (integer)</code>		maximum number of events in an output batch. Examples: <code>42</code> , <code>1.2e-10</code>
<code>mode</code>	<code>Mode</code>	✓	If 'document' send on end of document generated by input. If 'fixed', use <code>fixed_size</code> . Allowed values: <code>fixed</code> , <code>document</code>
<code>timeout</code>	<code>time-interval (string)</code>	✓	interval after which the batch is sent, to keep throughput going (default 100ms). Default: <code>100ms</code> Examples: <code>500ms</code> , <code>2h</code>
<code>header</code>	<code>multiline-text (string)</code>		put a header line before the batch.

<code>footer</code>	<code>multiline-text</code> (<code>string</code>)		put a header line after the last line of the batch.
<code>use-document-marker</code> ✓	<code>boolean</code> (<code>bool</code>)		Enrich the job metadata with a document marker (for document handling in batch mode). Default: <code>false</code>
<code>wrap-as-json</code> ✓	<code>boolean</code> (<code>bool</code>)		Format the output batch as a JSON array. Default: <code>false</code>

Retry Fields

Field	Type	Required	Description
<code>timeout</code>	<code>time-interval</code> (<code>string</code>)	✓	timeout (e.g. 500ms, 2s etc. - default is 30). Examples: <code>500ms</code> , <code>2h</code>
<code>retries</code>	<code>number</code> (<code>integer</code>)		number of retries. Examples: <code>42</code> , <code>1.2e-10</code>

Mode Options

Value	Name	Description
<code>put</code>	put	Put Objects
<code>delete</code>	delete	Delete Objects

Batch - Mode Options

Value	Name	Description
<code>fixed</code>	fixed	Fixed
<code>document</code>	document	Document

Output Preprocessor Options

Value	Name	Description
<code>gzip</code>	gzip	Gzip the output data
<code>parquet</code>	parquet	Extract the received data as JSON rows from a parquet file
<code>base64</code>	base64	Decode base64 as binary

Splunk HEC

Reference for the Splunk HEC component in LyftData's DSL

Source: `reference/dsl/outputs/splunk-hec.md`

Splunk HEC (`splunk-hec`)

Output events to a Splunk HTTP Event Collector endpoint (Splunk HEC).

Observability Enterprise edition json

Minimal example

```
output:
splunk-hec: hec-token: hec-token-value: ~ url: ""
```

JSON

```
{
  "output": { "splunk-hec": { "hec-token": { "hec-token-value": null }, "url": "" } } }
```

Contents

- [Minimal example](#)
- [Collector Options](#)
- [Endpoint](#)
- [Processing](#)
- [Reliability](#)
- [Security](#)
- [Schema](#)

Collector Options

Collector Options

Field	Type	Required	Description
<code>hec-token</code>	<code>Hec Token</code>	<input checked="" type="checkbox"/>	Specify a value to use for the HEC Token or set it using an event field. Allowed values: <code>hec-token-value</code> , <code>hec-token-field</code>
<code>metrics</code> ✓	<code>boolean</code> (<code>bool</code>)	<input type="checkbox"/>	Send a metrics formatted payload to the HEC endpoint. Default: <code>false</code>

<code>event-field</code>	<code>field</code> (<code>string</code>)		If specified, the field's contents will be submitted as the event payload to the endpoint. Examples: <code>data_field</code>
<code>time-field</code>	<code>field</code> (<code>string</code>)		Use the specified field for the timestamp of the endpoint. This should be in Unix epoch format. Examples: <code>data_field</code>
<code>index</code>	<code>Index</code>		Specify a value to use for the Splunk <code>index</code> or set it using an event field. Allowed values: <code>index-value</code> , <code>index-field</code>
<code>host</code>	<code>Host</code>		Specify a value to use for the Splunk <code>host</code> or set it using an event field. Allowed values: <code>host-value</code> , <code>host-field</code>
<code>source</code>	<code>Source</code>		Specify a value to use for the Splunk <code>source</code> or set it using an event field. Allowed values: <code>source-value</code> , <code>source-field</code>
<code>sourcetype</code>	<code>Sourcetype</code>		Specify a value to use for the Splunk <code>sourcetype</code> or set it using an event field. Allowed values: <code>source-type-value</code> , <code>source-type-field</code>
<code>remove</code> ✓	<code>boolean</code> (<code>bool</code>)		Consume (remove) fields from the event payload before submitting to the endpoint. Applicable to <code>time-field</code> , <code>host-field</code> , <code>source-field</code> , <code>sourcetype-field</code> , <code>index-field</code> and <code>hec-token-field</code> . Default: <code>false</code>

Endpoint

Endpoint

Field	Type	Required	Description
<code>url</code>	<code>url</code> (<code>string</code>)	✓	The URL of the Splunk HEC instance (example: <code>https://127.0.0.1:8088/services/collector/event</code>). Examples: <code>https://example.com/path</code>

Processing

Processing

Field	Type	Required	Description
<code>batch</code>	<code>Batch</code>		Batching input events together.

Reliability

Reliability

Field	Type	Required	Description
<code>retry</code>	<code>Retry</code>		How to retry this operation.

Security

Security

Field	Type	Required	Description
<code>insecure</code> ✓	boolean (bool)		Ignore TLS certificate validation errors (This is not recommended). Default: <code>false</code>
<code>disable-preflight</code> ✓	boolean (bool)		Disable the Splunk HEC "preflight" verification request (not recommended). When enabled (default), LyftData performs a lightweight request to a derived HEC health URL before sending any event payloads, to reduce the risk of accidental misconfiguration sending sensitive data to a non-HEC endpoint. Default: <code>false</code>

Schema

- [Hec Token Options](#)
- [Index Options](#)
- [Host Options](#)
- [Source Options](#)
- [Sourcetype Options](#)
- [Batch Fields](#)
- [Retry Fields](#)
- [Batch - Mode Options](#)

Hec Token Options

Option	Name	Type	Description
<code>hec-token-value</code>	Hec Token Value	string	
<code>hec-token-field</code>	Hec Token Field	string	Examples: <code>data_field</code>

Index Options

Option	Name	Type	Description
<code>index-value</code>	Index Value	string	
<code>index-field</code>	Index Field	string	Examples: <code>data_field</code>

Host Options

Option	Name	Type	Description
<code>host-value</code>	Host Value	string	
<code>host-field</code>	Host Field	string	Examples: <code>data_field</code>

Source Options

Option	Name	Type	Description
--------	------	------	-------------

<code>source-value</code>	Source Value	<code>string</code>	
<code>source-field</code>	Source Field	<code>string</code>	Examples: <code>data_field</code>

Sourcetype Options

Option	Name	Type	Description
<code>source-type-value</code>	Source Type Value	<code>string</code>	
<code>source-type-field</code>	Source Type Field	<code>string</code>	Examples: <code>data_field</code>

Batch Fields

Field	Type	Required	Description
<code>fixed-size</code> ✓	<code>number</code> (<code>integer</code>)		maximum number of events in an output batch. Examples: <code>42</code> , <code>1.2e-10</code>
<code>mode</code>	<code>Mode</code>	✓	If 'document' send on end of document generated by input. If 'fixed', use <code>fixed_size</code> . Allowed values: <code>fixed</code> , <code>document</code>
<code>timeout</code>	<code>time-interval</code> (<code>string</code>)	✓	interval after which the batch is sent, to keep throughput going (default 100ms). Default: <code>100ms</code> Examples: <code>500ms</code> , <code>2h</code>
<code>header</code>	<code>multiline-text</code> (<code>string</code>)		put a header line before the batch.
<code>footer</code>	<code>multiline-text</code> (<code>string</code>)		put a header line after the last line of the batch.
<code>use-document-marker</code> ✓	<code>boolean</code> (<code>bool</code>)		Enrich the job metadata with a document marker (for document handling in batch mode). Default: <code>false</code>
<code>wrap-as-json</code> ✓	<code>boolean</code> (<code>bool</code>)		Format the output batch as a JSON array. Default: <code>false</code>

Retry Fields

Field	Type	Required	Description
<code>timeout</code>	<code>time-interval</code> (<code>string</code>)	✓	timeout (e.g. 500ms, 2s etc. - default is 30). Examples: <code>500ms</code> , <code>2h</code>
<code>retries</code>	<code>number</code> (<code>integer</code>)		number of retries. Examples: <code>42</code> , <code>1.2e-10</code>

Batch - Mode Options

Value	Name	Description
<code>fixed</code>	<code>fixed</code>	Fixed
<code>document</code>	<code>document</code>	Document

ODBC SQL

Reference for the ODBC SQL component in LyftData's DSL

Source: <reference/dsl/outputs/sql-odbc.md>

ODBC SQL (`sql-odbc`)

Execute parameterised statements against relational databases over ODBC.

Database Enterprise edition json

Minimal example

```
output:
sql-odbc: connection: url: url: ""
```

JSON

```
{
  "output": { "sql-odbc": { "connection": { "url": { "url": "" } } } } }
```

Contents

- [Minimal example](#)
- [Connection](#)
- [Processing](#)
- [Reliability](#)
- [Result](#)
- [Statement](#)
- [Transaction](#)
- [Schema](#)

Connection

Connection

Field	Type	Required	Description
<code>connection</code>	<code>Connection</code>	✓	Connection details for the target database. Allowed values: <code>url</code> , <code>parts</code>

Processing

Processing

Field	Type	Required	Description
<code>batch</code>	<code>Batch</code>		Logical batching hints shared with other outputs.

Reliability

Reliability

Field	Type	Required	Description
<code>retry</code>	<code>Retry</code>		Retry policy for failed executions.

Result

Result

Field	Type	Required	Description
<code>result-field</code>	<code>field</code> (string)		Event field updated with statement scalar results. Examples: <code>data_field</code>
<code>affected-rows-field</code>	<code>field</code> (string)		Event field populated with affected row counts. Examples: <code>data_field</code>

Statement

Statement

Field	Type	Required	Description
<code>statement</code>	<code>multiline-text</code> (string)		Inline SQL statement executed when <code>statement-file</code> is not provided.
<code>statement-file</code>	<code>path</code> (string)		Path to a SQL file executed when <code>statement</code> is not provided. Examples: <code>/path/to/file</code> , <code>c:\users\joe\data\file.txt</code>
<code>parameters</code>	[Parameters[]] (#parameters-fields)		Bindings that map event fields into parameter placeholders.

Transaction

Transaction

Field	Type	Required	Description
<code>transaction</code>	<code>Transaction</code>		Optional transaction scope applied per execution.

Schema

- [Connection Options](#)
- [Connection - Url Fields](#)
- [Connection - Parts - Tuning Fields](#)
- [Connection - Parts Fields](#)
- [Batch Fields](#)
- [Retry Fields](#)
- [Parameters Fields](#)
- [Transaction Fields](#)
- [Batch - Mode Options](#)
- [Transaction - Mode Options](#)

Connection Options

Option	Name	Type	Description
<code>url</code>	Url	object	
<code>parts</code>	Parts	object	

Connection - Url Fields

Field	Type	Required	Description
<code>url</code>	<code>url (string)</code>	<input checked="" type="checkbox"/>	Database connection URL (sqlx/ODBC compatible). Examples: <code>https://example.com/path</code>
<code>max-connections</code>	<code>number (integer)</code>		Maximum size of the async connection pool. Examples: <code>42</code> , <code>1.2e-10</code>
<code>min-connections</code>	<code>number (integer)</code>		Minimum number of connections retained in the pool. Examples: <code>42</code> , <code>1.2e-10</code>
<code>connect-timeout</code>	<code>string</code>		Optional connection timeout (humantime duration string).
<code>idle-timeout</code>	<code>string</code>		Optional idle timeout (humantime duration string).
<code>schema</code>	<code>string</code>		Optional schema override applied after connect.
<code>options</code>	<code>string[]</code>		Statements executed after connect (e.g. SET search_path TO ...).
<code>disable-tls-verification</code> ✓	<code>boolean (bool)</code>		Skip TLS verification (development only). Default: <code>false</code>

Connection - Parts - Tuning Fields

Field	Type	Required	Description
<code>max-connections</code>	<code>number (integer)</code>		Maximum size of the async connection pool. Examples: <code>42</code> , <code>1.2e-10</code>

<code>min-connections</code>	<code>number</code> (<code>integer</code>)		Minimum number of connections retained in the pool. Examples: <code>42</code> , <code>1.2e-10</code>
<code>connect-timeout</code>	<code>string</code>		Optional connection timeout (humantime duration string).
<code>idle-timeout</code>	<code>string</code>		Optional idle timeout (humantime duration string).
<code>schema</code>	<code>string</code>		Optional schema override applied after connect.
<code>options</code>	<code>string[]</code>		Statements executed after connect (e.g. SET search_path TO ...).
<code>disable-tls-verification</code> ✓	<code>boolean</code> (<code>bool</code>)		Skip TLS verification (development only). Default: <code>false</code>

Connection - Parts Fields

Field	Type	Required	Description
<code>tuning</code>	<code>Tuning</code>	✓	
<code>connection-string</code>	<code>string</code>	✓	ODBC connection string (translated to <code>odbc:<connection-string></code>).

Batch Fields

Field	Type	Required	Description
<code>fixed-size</code> ✓	<code>number</code> (<code>integer</code>)		maximum number of events in an output batch. Examples: <code>42</code> , <code>1.2e-10</code>
<code>mode</code>	<code>Mode</code>	✓	If 'document' send on end of document generated by input. If 'fixed', use <code>fixed_size</code> . Allowed values: <code>fixed</code> , <code>document</code>
<code>timeout</code>	<code>time-interval</code> (<code>string</code>)	✓	interval after which the batch is sent, to keep throughput going (default 100ms). Default: <code>100ms</code> Examples: <code>500ms</code> , <code>2h</code>
<code>header</code>	<code>multiline-text</code> (<code>string</code>)		put a header line before the batch.
<code>footer</code>	<code>multiline-text</code> (<code>string</code>)		put a header line after the last line of the batch.
<code>use-document-marker</code> ✓	<code>boolean</code> (<code>bool</code>)		Enrich the job metadata with a document marker (for document handling in batch mode). Default: <code>false</code>
<code>wrap-as-json</code> ✓	<code>boolean</code> (<code>bool</code>)		Format the output batch as a JSON array. Default: <code>false</code>

Retry Fields

Field	Type	Required	Description
-------	------	----------	-------------

<code>timeout</code>	<code>time-interval</code> (<code>string</code>)	<input checked="" type="checkbox"/>	timeout (e.g. 500ms, 2s etc. - default is 30). Examples: <code>500ms</code> , <code>2h</code>
<code>retries</code>	<code>number</code> (<code>integer</code>)		number of retries. Examples: <code>42</code> , <code>1.2e-10</code>

Parameters Fields

Field	Type	Required	Description
<code>name</code>	<code>string</code>	<input checked="" type="checkbox"/>	
<code>field</code>	<code>string</code>	<input checked="" type="checkbox"/>	
<code>default</code>	<code>multiline-text</code> (<code>string</code>)		

Transaction Fields

Field	Type	Required	Description
<code>mode</code>	<code>Mode</code>		Allowed values: <code>none</code> , <code>per-event</code> , <code>per-batch</code> , <code>document</code>
<code>isolation-level</code>	<code>string</code>		
<code>access-mode</code>	<code>string</code>		
<code>deferrable</code>	<code>boolean</code> (<code>bool</code>)		

Batch - Mode Options

Value	Name	Description
<code>fixed</code>	fixed	Fixed
<code>document</code>	document	Document

Transaction - Mode Options

Value	Name	Description
<code>none</code>	none	None
<code>per-event</code>	per-event	Per Event
<code>per-batch</code>	per-batch	Per Batch
<code>document</code>	document	Document

SQL Server

Reference for the SQL Server component in LyftData's DSL

Source: [reference/dsl/outputs/sql-server.md](#)

SQL Server (`sql-server`)

Execute parameterised statements against Microsoft SQL Server databases.

Database Enterprise edition json

Minimal example

```
output:
sql-server: connection: url: url: ""
```

JSON

```
{
  "output": { "sql-server": { "connection": { "url": { "url": "" } } } } }
```

Contents

- [Minimal example](#)
- [Connection](#)
- [Processing](#)
- [Reliability](#)
- [Result](#)
- [Statement](#)
- [Transaction](#)
- [Schema](#)

Connection

Connection

Field	Type	Required	Description
<code>connection</code>	<code>Connection</code>	✓	Connection details for the target database. Allowed values: <code>url</code> , <code>parts</code>

Processing

Processing

Field	Type	Required	Description
<code>batch</code>	<code>Batch</code>		Logical batching hints shared with other outputs.

Reliability

Reliability

Field	Type	Required	Description
<code>retry</code>	<code>Retry</code>		Retry policy for failed executions.

Result

Result

Field	Type	Required	Description
<code>result-field</code>	<code>field</code> (string)		Event field updated with statement scalar results. Examples: <code>data_field</code>
<code>affected-rows-field</code>	<code>field</code> (string)		Event field populated with affected row counts. Examples: <code>data_field</code>

Statement

Statement

Field	Type	Required	Description
<code>statement</code>	<code>multiline-text</code> (string)		Inline SQL statement executed when <code>statement-file</code> is not provided.
<code>statement-file</code>	<code>path</code> (string)		Path to a SQL file executed when <code>statement</code> is not provided. Examples: <code>/path/to/file</code> , <code>c:\users\joe\data\file.txt</code>
<code>parameters</code>	[Parameters[]] (#parameters-fields)		Bindings that map event fields into parameter placeholders.

Transaction

Transaction

Field	Type	Required	Description
-------	------	----------	-------------

<code>transaction</code>	<code>Transaction</code>	Optional transaction scope applied per execution (supports <code>none</code> and <code>per-event</code>).
--------------------------	--------------------------	--

Schema

- [Connection Options](#)
- [Connection - Url Fields](#)
- [Connection - Parts - Tuning Fields](#)
- [Connection - Parts Fields](#)
- [Batch Fields](#)
- [Retry Fields](#)
- [Parameters Fields](#)
- [Transaction Fields](#)
- [Connection - Parts - Parameters Table](#)
- [Batch - Mode Options](#)
- [Transaction - Mode Options](#)

Connection Options

Option	Name	Type	Description
<code>url</code>	<code>Url</code>	<code>object</code>	
<code>parts</code>	<code>Parts</code>	<code>object</code>	

Connection - Url Fields

Field	Type	Required	Description
<code>url</code>	<code>url (string)</code>	<input checked="" type="checkbox"/>	Database connection URL (sqlx/ODBC compatible). Examples: <code>https://example.com/path</code>
<code>max-connections</code>	<code>number (integer)</code>		Maximum size of the async connection pool. Examples: <code>42</code> , <code>1.2e-10</code>
<code>min-connections</code>	<code>number (integer)</code>		Minimum number of connections retained in the pool. Examples: <code>42</code> , <code>1.2e-10</code>
<code>connect-timeout</code>	<code>string</code>		Optional connection timeout (humantime duration string).
<code>idle-timeout</code>	<code>string</code>		Optional idle timeout (humantime duration string).
<code>schema</code>	<code>string</code>		Optional schema override applied after connect.
<code>options</code>	<code>string[]</code>		Statements executed after connect (e.g. SET search_path TO ...).
<code>disable-tls-verification</code> ✓	<code>boolean (bool)</code>		Skip TLS verification (development only). Default: <code>false</code>

Connection - Parts - Tuning Fields

Field	Type	Required	Description
<code>max-connections</code>	<code>number</code> (<code>integer</code>)		Maximum size of the async connection pool. Examples: <code>42</code> , <code>1.2e-10</code>
<code>min-connections</code>	<code>number</code> (<code>integer</code>)		Minimum number of connections retained in the pool. Examples: <code>42</code> , <code>1.2e-10</code>
<code>connect-timeout</code>	<code>string</code>		Optional connection timeout (humantime duration string).
<code>idle-timeout</code>	<code>string</code>		Optional idle timeout (humantime duration string).
<code>schema</code>	<code>string</code>		Optional schema override applied after connect.
<code>options</code>	<code>string[]</code>		Statements executed after connect (e.g. SET search_path TO ...).
<code>disable-tls-verification</code> ✓	<code>boolean</code> (<code>bool</code>)		Skip TLS verification (development only). Default: <code>false</code>

Connection - Parts Fields

Field	Type	Required	Description
<code>tuning</code>	<code>Tuning</code>	✓	
<code>host</code>	<code>string</code>	✓	Hostname or IP address of the database server.
<code>port</code>	<code>number</code> (<code>integer</code>)		Optional TCP port (defaults to the backend default). Examples: <code>42</code> , <code>1.2e-10</code>
<code>database</code>	<code>string</code>		Optional database name (mapped to the URL path segment).
<code>username</code>	<code>string</code>		Optional username (mapped to the URL userinfo).
<code>password</code>	<code>string</code>		Optional password (mapped to the URL userinfo).
<code>parameters</code>	<code>map</code> (<code>string</code>)		Additional query parameters appended to the generated URL.

Batch Fields

Field	Type	Required	Description
<code>fixed-size</code> ✓	<code>number</code> (<code>integer</code>)		maximum number of events in an output batch. Examples: <code>42</code> , <code>1.2e-10</code>
<code>mode</code>	<code>Mode</code>	✓	If 'document' send on end of document generated by input. If 'fixed', use <code>fixed_size</code> . Allowed values: <code>fixed</code> , <code>document</code>
<code>timeout</code>	<code>time-interval</code> (<code>string</code>)	✓	interval after which the batch is sent, to keep throughput going (default 100ms). Default: <code>100ms</code> Examples: <code>500ms</code> , <code>2h</code>
<code>header</code>	<code>multiline-text</code> (<code>string</code>)		put a header line before the batch.

<code>footer</code>	<code>multiline-text</code> (<code>string</code>)		put a header line after the last line of the batch.
<code>use-document-marker</code> ✓	<code>boolean</code> (<code>bool</code>)		Enrich the job metadata with a document marker (for document handling in batch mode). Default: <code>false</code>
<code>wrap-as-json</code> ✓	<code>boolean</code> (<code>bool</code>)		Format the output batch as a JSON array. Default: <code>false</code>

Retry Fields

Field	Type	Required	Description
<code>timeout</code>	<code>time-interval</code> (<code>string</code>)	✓	timeout (e.g. 500ms, 2s etc. - default is 30). Examples: <code>500ms</code> , <code>2h</code>
<code>retries</code>	<code>number</code> (<code>integer</code>)		number of retries. Examples: <code>42</code> , <code>1.2e-10</code>

Parameters Fields

Field	Type	Required	Description
<code>name</code>	<code>string</code>	✓	
<code>field</code>	<code>string</code>	✓	
<code>default</code>	<code>multiline-text</code> (<code>string</code>)		

Transaction Fields

Field	Type	Required	Description
<code>mode</code>	<code>Mode</code>		Allowed values: <code>none</code> , <code>per-event</code> , <code>per-batch</code> , <code>document</code>
<code>isolation-level</code>	<code>string</code>		
<code>access-mode</code>	<code>string</code>		
<code>deferrable</code>	<code>boolean</code> (<code>bool</code>)		

Connection - Parts - Parameters Table

Parameter	Value
<code>name</code>	<code>value</code>

Value format: `templated-text` .

Batch - Mode Options

Value	Name	Description
<code>fixed</code>	<code>fixed</code>	Fixed

<code>document</code>	document	Document
-----------------------	----------	----------

Transaction - Mode Options

Value	Name	Description
<code>none</code>	none	None
<code>per-event</code>	per-event	Per Event
<code>per-batch</code>	per-batch	Per Batch
<code>document</code>	document	Document

Syslog

Reference for the Syslog component in LyftData's DSL

Source: [reference/dsl/outputs/syslog.md](#)

Syslog (`syslog`)

Emit events as syslog messages to remote or local syslog receivers. Supported endpoint schemes: - `udp://host:port` - `tcp://host:port` - `tls://host:port` - `unix:///path` (unix stream) - `unixgram:///path` (unix datagram).

Observability Enterprise edition json

Minimal example

```
output:
  syslog: endpoint: ""
```

JSON

```
{
  "output": { "syslog": { "endpoint": "" } } }
```

Contents

- [Minimal example](#)
- [Batching](#)
- [Endpoint](#)
- [Format](#)
- [Framing](#)
- [Headers](#)
- [Payload](#)
- [Priority](#)
- [Reliability](#)
- [Security](#)
- [Structured Data](#)
- [Schema](#)

Batching

Batching

Field	Type	Required	Description
<code>batch</code>	<code>Batch</code>		Optional batching hints (still emits one syslog message per event).

Endpoint

Endpoint

Field	Type	Required	Description
<code>endpoint</code>	<code>hostname</code> (<code>string</code>)	<input checked="" type="checkbox"/>	Syslog destination endpoint (selects transport by scheme). Examples: <code>example.com</code> , <code>localhost</code>

Format

Format

Field	Type	Required	Description
<code>format</code>	<code>Format</code>		Syslog message format to emit. Default: <code>rfc5424</code> Allowed values: <code>rfc5424</code> , <code>rfc3164</code>

Framing

Framing

Field	Type	Required	Description
<code>framing</code>	<code>Framing</code>		TCP/TLS/Unix-stream framing mode. Default: <code>octet-counting</code> Allowed values: <code>octet-counting</code> , <code>newline</code>

Headers

Headers

Field	Type	Required	Description
<code>hostname</code>	<code>Hostname</code>		RFC 5424 hostname (static value or event field). Defaults to <code>-</code> (NILVALUE). Allowed values: <code>hostname-value</code> , <code>hostname-field</code>
<code>app-name</code>	<code>App Name</code>		RFC 5424 app-name (static value or event field). Defaults to <code>-</code> (NILVALUE). Allowed values: <code>app-name-value</code> , <code>app-name-field</code>
<code>procid</code>	<code>Procid</code>		RFC 5424 procid (static value or event field). Defaults to <code>-</code> (NILVALUE). Allowed values: <code>proc-id-value</code> , <code>proc-id-field</code>
<code>msgid</code>	<code>Msgid</code>		RFC 5424 msgid (static value or event field). Defaults to <code>-</code> (NILVALUE). Allowed values: <code>msg-id-value</code> , <code>msg-id-field</code>

Payload

Payload

Field	Type	Required	Description
<code>input-field</code>	<code>field</code> (<code>string</code>)		If set, this field's value becomes the syslog payload (and is also used for field-derived syslog properties like facility/severity). Examples: <code>data_field</code>

Priority

Priority

Field	Type	Required	Description
<code>facility</code>	<code>Facility</code>		Syslog facility (static value or event field). Allowed values: <code>facility-value</code> , <code>facility-field</code>
<code>severity</code>	<code>Severity</code>		Syslog severity (static value or event field). Allowed values: <code>severity-value</code> , <code>severity-field</code>

Reliability

Reliability

Field	Type	Required	Description
<code>retry</code>	<code>Retry</code>		Retry policy for connection and send failures (TCP/TLS/Unix stream).

Security

Security

Field	Type	Required	Description
<code>tls</code>	<code>Tls</code>		TLS configuration (applies to <code>tls://</code> endpoints only).

Structured Data

Structured Data

Field	Type	Required	Description
<code>structured-data</code>	<code>Structured Data</code>		Optional RFC 5424 structured-data encoding. When enabled, the (possibly <code>input_field</code> selected) payload must be a JSON object, or a JSON string that can be parsed into an object when <code>parse_json_string=true</code> .

Schema

- [Hostname Options](#)
- [App Name Options](#)
- [Procid Options](#)
- [Msgid Options](#)
- [Facility Options](#)
- [Severity Options](#)
- [Batch Fields](#)
- [Retry Fields](#)
- [Tls Fields](#)
- [Structured Data Fields](#)
- [Batch - Mode Options](#)
- [Format Options](#)
- [Framing Options](#)

Hostname Options

Option	Name	Type	Description
<code>hostname-value</code>	Hostname Value	<code>string</code>	
<code>hostname-field</code>	Hostname Field	<code>string</code>	Examples: <code>data_field</code>

App Name Options

Option	Name	Type	Description
<code>app-name-value</code>	App Name Value	<code>string</code>	
<code>app-name-field</code>	App Name Field	<code>string</code>	Examples: <code>data_field</code>

Procid Options

Option	Name	Type	Description
<code>proc-id-value</code>	Proc Id Value	<code>string</code>	
<code>proc-id-field</code>	Proc Id Field	<code>string</code>	Examples: <code>data_field</code>

Msgid Options

Option	Name	Type	Description
<code>msg-id-value</code>	Msg Id Value	<code>string</code>	
<code>msg-id-field</code>	Msg Id Field	<code>string</code>	Examples: <code>data_field</code>

Facility Options

Option	Name	Type	Description
--------	------	------	-------------

<code>facility-value</code>	Facility Value	<code>string</code>	
<code>facility-field</code>	Facility Field	<code>string</code>	Examples: <code>data_field</code>

Severity Options

Option	Name	Type	Description
<code>severity-value</code>	Severity Value	<code>string</code>	
<code>severity-field</code>	Severity Field	<code>string</code>	Examples: <code>data_field</code>

Batch Fields

Field	Type	Required	Description
<code>fixed-size</code> ✓	<code>number</code> (<code>integer</code>)		maximum number of events in an output batch. Examples: <code>42</code> , <code>1.2e-10</code>
<code>mode</code>	<code>Mode</code>	✓	If 'document' send on end of document generated by input. If 'fixed', use <code>fixed_size</code> . Allowed values: <code>fixed</code> , <code>document</code>
<code>timeout</code>	<code>time-interval</code> (<code>string</code>)	✓	interval after which the batch is sent, to keep throughput going (default 100ms). Default: <code>100ms</code> Examples: <code>500ms</code> , <code>2h</code>
<code>header</code>	<code>multiline-text</code> (<code>string</code>)		put a header line before the batch.
<code>footer</code>	<code>multiline-text</code> (<code>string</code>)		put a header line after the last line of the batch.
<code>use-document-marker</code> ✓	<code>boolean</code> (<code>bool</code>)		Enrich the job metadata with a document marker (for document handling in batch mode). Default: <code>false</code>
<code>wrap-as-json</code> ✓	<code>boolean</code> (<code>bool</code>)		Format the output batch as a JSON array. Default: <code>false</code>

Retry Fields

Field	Type	Required	Description
<code>timeout</code>	<code>time-interval</code> (<code>string</code>)	✓	timeout (e.g. 500ms, 2s etc. - default is 30). Examples: <code>500ms</code> , <code>2h</code>
<code>retries</code>	<code>number</code> (<code>integer</code>)		number of retries. Examples: <code>42</code> , <code>1.2e-10</code>

Tls Fields

Field	Type	Required	Description
<code>ca-certificate</code>	<code>string</code>		Custom CA certificate bundle (PEM or path).
<code>client-certificate</code>	<code>string</code>		Client certificate for mutual TLS (PEM or path).

<code>client-key</code>	<code>string</code>		Client private key for mutual TLS (PEM or path).
<code>server-name</code>	<code>string</code>		Override the TLS server name used for certificate validation (SNI).
<code>insecure-skip-verify</code> ✓	<code>boolean</code> (<code>bool</code>)		Skip certificate validation (development only). Default: <code>false</code>

Structured Data Fields

Field	Type	Required	Description
<code>sd-id</code>	<code>string</code>	✓	SD-ID used for the RFC 5424 structured data element (example: <code>lyftdata@32473</code>).
<code>parse-json-string</code> ✓	<code>boolean</code> (<code>bool</code>)		If the payload is a JSON string, parse it as JSON (must decode to an object) before encoding into structured data. Default: <code>false</code>

Batch - Mode Options

Value	Name	Description
<code>fixed</code>	fixed	Fixed
<code>document</code>	document	Document

Format Options

Value	Name	Description
<code>rfc5424</code>	rfc5424	Rfc5424
<code>rfc3164</code>	rfc3164	Rfc3164

Framing Options

Value	Name	Description
<code>octet-counting</code>	octet-counting	RFC 6587 octet-counting framing: <code><len> <message></code> .
<code>newline</code>	newline	Newline-delimited framing (LF).

Web Dav Store

Reference for the Web Dav Store component in LyftData's DSL

Source: <reference/dsl/outputs/web-dav-store.md>

Web Dav Store (`web-dav-store`)

Write to a WebDAV-compatible object store.

Block Store WebDAV Enterprise edition binary json raw

Minimal example

```
output:
web-dav-store: base-url: ~ object-name: name: ~
```

JSON

```
{
"output": { "web-dav-store": { "base-url": null, "object-name": { "name": null } } } }
```

Contents

- [Minimal example](#)
- [Advanced](#)
- [Behavior](#)
- [Location](#)
- [Object Properties](#)
- [Processing](#)
- [Reliability](#)
- [Schema](#)

Advanced

Advanced

Field	Type	Required	Description
<code>config</code>	<code>map</code> (<code>string</code>)		Additional configuration passed directly to the HTTP client (see <code>object_store</code> http configs).

Behavior

Behavior

Field	Type	Required	Description
<code>mode</code>	<code>Mode</code>		Specify whether we're putting or deleting. Allowed values: <code>put</code> , <code>delete</code>

Location

Location

Field	Type	Required	Description
<code>base-url</code>	<code>string</code>	✓	Base URL for the WebDAV endpoint (e.g., <code>https://host/webdav</code>).
<code>bucket-name</code>	<code>string</code>		Optional bucket or root path that will be prepended to object names.
<code>object-name</code>	<code>Object Name</code>	✓	File name (may contain slashes). Allowed values: <code>name</code> , <code>field</code>

Object Properties

Object Properties

Field	Type	Required	Description
<code>disable-object-name-guid</code> ✓	<code>boolean</code> (<code>bool</code>)		Disable the GUID prefix if you want object name to be treated literally (off for deletes). Default: <code>false</code>
<code>guid-prefix</code>	<code>string</code>		GUID Prefix, will be prepended to the GUID, the default value is <code>"/</code> .
<code>guid-suffix</code>	<code>string</code>		GUID Suffix, will be appended to the GUID if specified.

Processing

Processing

Field	Type	Required	Description
<code>batch</code>	<code>Batch</code>		Write events as batches.
<code>input-field</code>	<code>field</code> (<code>string</code>)		Use the specified field as the content for the file line. Examples: <code>data_field</code>
<code>preprocessors</code>	[<code>Output Preprocessor</code> []] (#output-preprocessor-options)		Preprocessors (process data before making it available for upload) these processors will be run in the order they are specified. Allowed values: <code>gzip</code> , <code>parquet</code> , <code>base64</code>

Reliability

Reliability

Field	Type	Required	Description
<code>retry</code>	<code>Retry</code>		Retry on failure.
<code>track-schema</code> ✓	<code>boolean</code> (<code>bool</code>)		Check the schema of the written data and update <code>__SCHEMA_NUMBER</code> (written data must be JSON). Default: <code>false</code>

Schema

- [Object Name Options](#)
- [Batch Fields](#)
- [Retry Fields](#)
- [Config Table](#)
- [Mode Options](#)
- [Batch - Mode Options](#)
- [Output Preprocessor Options](#)

Object Name Options

Option	Name	Type	Description
<code>name</code>	Name	<code>string</code>	Object Name.
<code>field</code>	Field	<code>string</code>	Field containing the Object Name.

Batch Fields

Field	Type	Required	Description
<code>fixed-size</code> ✓	<code>number</code> (<code>integer</code>)		maximum number of events in an output batch. Examples: <code>42</code> , <code>1.2e-10</code>
<code>mode</code>	<code>Mode</code>	✓	If 'document' send on end of document generated by input. If 'fixed', use <code>fixed_size</code> . Allowed values: <code>fixed</code> , <code>document</code>
<code>timeout</code>	<code>time-interval</code> (<code>string</code>)	✓	interval after which the batch is sent, to keep throughput going (default 100ms). Default: <code>100ms</code> Examples: <code>500ms</code> , <code>2h</code>
<code>header</code>	<code>multiline-text</code> (<code>string</code>)		put a header line before the batch.
<code>footer</code>	<code>multiline-text</code> (<code>string</code>)		put a header line after the last line of the batch.
<code>use-document-marker</code> ✓	<code>boolean</code> (<code>bool</code>)		Enrich the job metadata with a document marker (for document handling in batch mode). Default: <code>false</code>
<code>wrap-as-json</code> ✓	<code>boolean</code> (<code>bool</code>)		Format the output batch as a JSON array. Default: <code>false</code>

Retry Fields

Field	Type	Required	Description
timeout	time-interval (string)	<input checked="" type="checkbox"/>	timeout (e.g. 500ms, 2s etc. - default is 30). Examples: 500ms, 2h
retries	number (integer)		number of retries. Examples: 42, 1.2e-10

Config Table

Setting	Value
setting.name	value

Value format: templated-text.

Mode Options

Value	Name	Description
put	put	Put Objects
delete	delete	Delete Objects

Batch - Mode Options

Value	Name	Description
fixed	fixed	Fixed
document	document	Document

Output Preprocessor Options

Value	Name	Description
gzip	gzip	Gzip the output data
parquet	parquet	Extract the received data as JSON rows from a parquet file
base64	base64	Decode base64 as binary

WebSocket

Reference for the WebSocket component in LyftData's DSL

Source: `reference/dsl/outputs/web-socket.md`

WebSocket (`web-socket`)

Send events to remote WebSocket peers or expose an embedded server.

Messaging Enterprise edition json

Minimal example

```
output:  
web-socket: {}
```

JSON

```
{  
  "output": { "web-socket": {} } }
```

Contents

- [Minimal example](#)
- [Authentication](#)
- [Batching](#)
- [Connection](#)
- [Cursors](#)
- [Framing](#)
- [Headers](#)
- [Health](#)
- [Reliability](#)
- [Security](#)
- [Server Output](#)
- [Schema](#)

Authentication

Authentication

Field	Type	Required	Description
-------	------	----------	-------------

<code>auth</code>	<code>Auth</code>		Authentication material inserted into the handshake.
-------------------	-------------------	--	--

Batching

Batching

Field	Type	Required	Description
<code>batch</code>	<code>Batch</code>		Logical batching hints.

Connection

Connection

Field	Type	Required	Description
<code>connect</code>	<code>Connect</code>		Connection strategy for establishing outbound sockets.
<code>url</code> ✓	<code>url</code> (<code>string</code>)		Remote endpoint to publish to (required for client mode). Examples: <code>https://example.com/path</code>
<code>mode</code>	<code>Mode</code>		Operating mode for this output. Allowed values: <code>client</code> , <code>server-output</code> , <code>server-input</code> , <code>relay</code>

Cursors

Cursors

Field	Type	Required	Description
<code>cursor</code>	<code>Cursor</code>		Cursor configuration for resume tokens.

Framing

Framing

Field	Type	Required	Description
<code>payload</code>	<code>Payload</code>		Encoding strategy for emitted frames. Default: <code>text</code> Allowed values: <code>text</code> , <code>json</code> , <code>binary</code> , <code>binary-base64</code>

Headers

Headers

Field	Type	Required	Description
-------	------	----------	-------------

headers ✓	map (string)		Headers to include during the WebSocket handshake.
query ✓	map (string)		Query parameters appended to the connection URL.
subprotocols	string[]		Preferred subprotocols.

Health

Health

Field	Type	Required	Description
heartbeat	Heartbeat		Optional heartbeat configuration for outgoing connections.

Reliability

Reliability

Field	Type	Required	Description
retry	Retry		Retry policy for failed sends.
reconnect	Reconnect		Reconnection policy for client mode.

Security

Security

Field	Type	Required	Description
tls	Tls		TLS configuration used when acting as a client.

Server Output

Server Output

Field	Type	Required	Description
server	Server		Embedded server configuration when exposing a WebSocket endpoint.

Schema

- [Auth - Basic Fields](#)
- [Auth Fields](#)
- [Batch Fields](#)
- [Connect Fields](#)
- [Cursor Fields](#)

- [Heartbeat Fields](#)
- [Retry Fields](#)
- [Reconnect Fields](#)
- [Tls Fields](#)
- [Server - Tls Fields](#)
- [Server Fields](#)
- [Auth - Headers Table](#)
- [Auth - Query Table](#)
- [Headers Table](#)
- [Query Table](#)
- [Batch - Mode Options](#)
- [Connect - Mode Options](#)
- [Mode Options](#)
- [Payload Options](#)
- [Heartbeat - Action Options](#)

Auth - Basic Fields

Field	Type	Required	Description
<code>username</code> ✓	<code>string</code>	✓	Username credential (supports templating).
<code>password</code> ✓	<code>string</code>	✓	Password credential (supports templating).

Auth Fields

Field	Type	Required	Description
<code>headers</code> ✓	<code>map (string)</code>		Static or templated headers added during the handshake.
<code>query</code> ✓	<code>map (string)</code>		Query parameters added to the connection URL.
<code>bearer-token</code> ✓	<code>string</code>		Bearer token inserted as <code>Authorization: Bearer ...</code> .
<code>basic</code>	<code>Basic</code>		Basic auth credentials sent via header.

Batch Fields

Field	Type	Required	Description
<code>fixed-size</code> ✓	<code>number (integer)</code>		maximum number of events in an output batch. Examples: <code>42</code> , <code>1.2e-10</code>
<code>mode</code>	<code>Mode</code>	✓	If 'document' send on end of document generated by input. If 'fixed', use <code>fixed_size</code> . Allowed values: <code>fixed</code> , <code>document</code>
<code>timeout</code>	<code>time-interval (string)</code>	✓	interval after which the batch is sent, to keep throughput going (default 100ms). Default: <code>100ms</code> Examples: <code>500ms</code> , <code>2h</code>
<code>header</code>	<code>multiline-text (string)</code>		put a header line before the batch.

<code>footer</code>	<code>multiline-text</code> (<code>string</code>)		put a header line after the last line of the batch.
<code>use-document-marker</code> ✓	<code>boolean</code> (<code>bool</code>)		Enrich the job metadata with a document marker (for document handling in batch mode). Default: <code>false</code>
<code>wrap-as-json</code> ✓	<code>boolean</code> (<code>bool</code>)		Format the output batch as a JSON array. Default: <code>false</code>

Connect Fields

Field	Type	Required	Description
<code>mode</code>	<code>Mode</code>		Specifies whether to connect eagerly during startup or lazily on demand. Allowed values: <code>lazy</code> , <code>eager</code>
<code>eager-timeout</code> ✓	<code>string</code>		Optional timeout to wait for an eager connection before failing start-up.

Cursor Fields

Field	Type	Required	Description
<code>header-name</code>	<code>string</code>		Header name used to propagate the cursor on reconnect (e.g. <code>Last-Event-ID</code>).
<code>query-parameter</code>	<code>string</code>		Query parameter name used to propagate the cursor value.
<code>persist-to</code>	<code>string</code>		Storage location for cursor state (defaults to job-scoped state).
<code>initial-value</code> ✓	<code>string</code>		Initial cursor value to send on the first connection.
<code>state-id</code>	<code>string</code>		Optional explicit state identifier when sharing cursors across jobs.
<code>resume-on-reconnect</code> ✓	<code>boolean</code> (<code>bool</code>)		Whether to inject the cursor on every reconnect attempt. Default: <code>false</code>
<code>value-field</code>	<code>field</code> (<code>string</code>)		Event field used to derive the cursor value after successful delivery. Examples: <code>data_field</code>
<code>value-pointer</code>	<code>json-pointer</code> (<code>string</code>)		JSON pointer used to derive the cursor value from emitted payloads.

Heartbeat Fields

Field	Type	Required	Description
<code>ping-interval</code> ✓	<code>string</code>		Interval between ping frames.
<code>timeout</code> ✓	<code>string</code>		Grace period for receiving a pong before the connection is considered unhealthy.

<code>action</code>	<code>Action</code>		What to do when a heartbeat misses the timeout. Allowed values: <code>reconnect</code> , <code>log-only</code>
<code>payload</code> ✓	<code>string</code>		Optional ping payload to send with heartbeats.

Retry Fields

Field	Type	Required	Description
<code>timeout</code>	<code>time-interval</code> (<code>string</code>)	✓	timeout (e.g. 500ms, 2s etc. - default is 30). Examples: <code>500ms</code> , <code>2h</code>
<code>retries</code>	<code>number</code> (<code>integer</code>)		number of retries. Examples: <code>42</code> , <code>1.2e-10</code>

Reconnect Fields

Field	Type	Required	Description
<code>initial-backoff</code> ✓	<code>string</code>		Backoff applied after the first failure (human-friendly duration).
<code>max-backoff</code> ✓	<code>string</code>		Maximum backoff duration before clamping (human-friendly duration).
<code>multiplier</code>	<code>number</code> (<code>integer</code>)		Exponential multiplier applied after each failure. Examples: <code>42</code> , <code>1.2e-10</code>
<code>jitter</code>	<code>number</code> (<code>integer</code>)		Random jitter factor applied to backoff windows (0.0-1.0). Examples: <code>42</code> , <code>1.2e-10</code>
<code>max-attempts</code>	<code>number</code> (<code>integer</code>)		Maximum number of consecutive attempts before giving up (None=infinite). Examples: <code>42</code> , <code>1.2e-10</code>
<code>reset-after</code> ✓	<code>string</code>		Time after a successful connection before resetting the attempt counter.

Tls Fields

Field	Type	Required	Description
<code>ca-certificate</code> ✓	<code>string</code>		Custom CA certificate to trust.
<code>client-certificate</code> ✓	<code>string</code>		Client certificate for mutual TLS.
<code>client-key</code> ✓	<code>string</code>		Client private key for mutual TLS.
<code>insecure-skip-verify</code> ✓	<code>boolean</code> (<code>bool</code>)		Skip certificate validation (unsafe). Default: <code>false</code>

Server - Tls Fields

Field	Type	Required	Description
<code>cert</code>	<code>path</code> (<code>string</code>)	✓	Path to the certificate file. Examples: <code>/path/to/file</code> , <code>c:\users\joe\data\file.txt</code>

key	path (string)	✓	Path to the private key file. Examples: /path/to/file , c:\users\joe\data\file.txt
-----	--------------------	---	---

Server Fields

Field	Type	Required	Description
bind	string	✓	Address to bind (e.g. 0.0.0.0:8080).
path	string		Optional path/route restriction.
tls	Tls		Optional TLS material for the listener.
max-connections	number (integer)		Maximum simultaneously connected clients (None = unlimited). Examples: 42 , 1.2e-10
idle-timeout ✓	string		Disconnect clients after this idle duration.
allowed-origins	string[]		Restrict accepted requests to these Origin / Sec-WebSocket- Origin values.
send-buffer- capacity	number (integer)		Per-connection send buffer (messages queued when clients are slow). Examples: 42 , 1.2e-10

Auth - Headers Table

Header	Value
Header-Name	value

Value format: templated-text .

Auth - Query Table

Parameter	Value
param	value

Value format: templated-text .

Headers Table

Header	Value
Header-Name	value

Value format: templated-text .

Query Table

Parameter	Value
-----------	-------

param	value
-------	-------

Value format: `templated-text`.

Batch - Mode Options

Value	Name	Description
<code>fixed</code>	fixed	Fixed
<code>document</code>	document	Document

Connect - Mode Options

Value	Name	Description
<code>lazy</code>	lazy	Defer connection until the first event needs to be sent.
<code>eager</code>	eager	Establish the connection during initialisation and keep it open.

Mode Options

Value	Name	Description
<code>client</code>	client	Client
<code>server-output</code>	server-output	Server Output
<code>server-input</code>	server-input	Server Input
<code>relay</code>	relay	Relay

Payload Options

Value	Name	Description
<code>text</code>	text	Text
<code>json</code>	json	Json
<code>binary</code>	binary	Binary
<code>binary-base64</code>	binary-base64	Binary Base64

Heartbeat - Action Options

Value	Name	Description
<code>reconnect</code>	reconnect	Reconnect
<code>log-only</code>	log-only	Log Only

Worker Channel

Reference for the Worker Channel component in LyftData's DSL

Source: `reference/dsl/outputs/worker-channel.md`

Worker Channel (`worker-channel`)

Receive events from a worker managed channel.

Messaging binary json raw

Minimal example

```
output:
worker-channel: worker-channel-name: ~
```

JSON

```
{
  "output": { "worker-channel": { "worker-channel-name": null } } }
```

Contents

- [Minimal example](#)
- [Channel](#)
- [Payload](#)

Channel

Channel

Field	Type	Required	Description
<code>worker-channel-name</code>	<code>string</code>	<input checked="" type="checkbox"/>	The unique name for this worker channel (e.g. "my-channel").

Payload

Payload

Field	Type	Required	Description
<code>input-field</code>	<code>field</code> (<code>string</code>)		The field to use as the input for this worker channel. Examples: <code>data_field</code>

Worker KV

Reference for the Worker KV component in LyftData's DSL

Source: `reference/dsl/outputs/worker-kv.md`

Worker KV (`worker-kv`)

Mutate the deployments-managed Worker KV store.

Storage json raw

Minimal example

```
output:
worker-kv: namespace: ~ operation: set: key: ~
```

JSON

```
{
"output": { "worker-kv": { "namespace": null, "operation": { "set": { "key": null } } } } }
```

Contents

- [Minimal example](#)
- [Connection](#)
- [Operation](#)
- [Schema](#)

Connection

Connection

Field	Type	Required	Description
<code>namespace</code>	<code>string</code>	<input checked="" type="checkbox"/>	Fully-qualified Worker KV namespace (format: <code>tenant:namespace</code> , e.g. <code>acme:deployments/my-deployment</code>).

Operation

Operation

Field	Type	Required	Description
operation	Worker KV Output Operation	✓	Operation to perform when emitting an event. Allowed values: <code>set</code> , <code>set-if-absent</code> , <code>delete</code> , <code>add-to-set</code> , <code>remove-from-set</code> , <code>ack-batch</code>

Schema

- [Worker KV Output Operation Options](#)
- [Worker KV Output Operation - Set Fields](#)
- [Worker KV Output Operation - Set If Absent Fields](#)
- [Worker KV Output Operation - Delete Fields](#)
- [Worker KV Output Operation - Add To Set Fields](#)
- [Worker KV Output Operation - Remove From Set Fields](#)
- [Worker KV Output Operation - Ack Batch Fields](#)

Worker KV Output Operation Options

Option	Name	Type	Description
<code>set</code>	Set	object	
<code>set-if-absent</code>	Set If Absent	object	
<code>delete</code>	Delete	object	
<code>add-to-set</code>	Add To Set	object	
<code>remove-from-set</code>	Remove From Set	object	
<code>ack-batch</code>	Ack Batch	object	

Worker KV Output Operation - Set Fields

Field	Type	Required	Description
key	string	✓	Key to set within the namespace.
value-field	field (string)		JSON pointer (or dotted path) to extract value from event; defaults to entire event. Examples: <code>data_field</code>
ttl-ms	number (integer)		Optional TTL in milliseconds. Examples: <code>42</code> , <code>1.2e-10</code>

Worker KV Output Operation - Set If Absent Fields

Field	Type	Required	Description
key	string	✓	Key to set when currently missing.
value-field	field (string)		JSON pointer (or dotted path) to extract value from event; defaults to entire event. Examples: <code>data_field</code>

<code>ttl-ms</code>	<code>number</code> (<code>integer</code>)		Optional TTL in milliseconds. Examples: <code>42</code> , <code>1.2e-10</code>
---------------------	---	--	--

Worker KV Output Operation - Delete Fields

Field	Type	Required	Description
<code>key</code>	<code>string</code>	✓	Key to delete within the namespace.

Worker KV Output Operation - Add To Set Fields

Field	Type	Required	Description
<code>set</code>	<code>string</code>	✓	Workset name to add members to.
<code>member-field</code>	<code>field</code> (<code>string</code>)		Pointer selecting the member identifier (defaults to full event). Examples: <code>data_field</code>
<code>value-field</code>	<code>field</code> (<code>string</code>)		Pointer selecting the payload to store with the member (defaults to full event). Examples: <code>data_field</code>
<code>lease-ttl-ms</code>	<code>number</code> (<code>integer</code>)		Optional lease TTL in milliseconds (used when immediately leasing member). Examples: <code>42</code> , <code>1.2e-10</code>

Worker KV Output Operation - Remove From Set Fields

Field	Type	Required	Description
<code>set</code>	<code>string</code>	✓	Workset name to delete from.
<code>member-field</code>	<code>field</code> (<code>string</code>)		Pointer selecting the member identifier (defaults to full event). Examples: <code>data_field</code>

Worker KV Output Operation - Ack Batch Fields

Field	Type	Required	Description
<code>set</code>	<code>string</code>	✓	Workset name associated with the batch handle.
<code>batch-handle-field</code>	<code>field</code> (<code>string</code>)	✓	Pointer selecting the batch handle emitted by the corresponding input. Examples: <code>data_field</code>