

Snowpipe Streaming API REST endpoints

PREVIEW FEATURE — OPEN

Support for this feature is available to all accounts on AWS only.

Note

We recommend that you use the `snowpipe-streaming` SDK as the primary and default choice. The REST API isn't optimized for high-throughput scenarios.

The following request headers apply to all the endpoints for the Snowpipe Streaming REST API:

Header	Description
<code>Authorization</code>	auth token
<code>X-Snowflake-Authorization-Token-Type</code> (optional)	JWT/OAuth

Note

The maximum allowed size for a single request payload is 16 MB. If your data is larger, you must split it into multiple requests.

Get Hostname

The `Get Hostname` returns the hostname used to interact with the Snowpipe Streaming REST API. Each account has a unique hostname.

```
GET /v2/streaming/hostname
```

Response:

Field	Type	Description
hostname	string	The hostname of the account.

Exchange Scoped Token

The `Exchange Scoped Token` returns a security token that can be used to access only the Snowpipe Streaming API-related service. This provides security protection for the customer.

```
POST /oauth/token
```

Request:

Attribute	Required	Component	Description
content_type	Yes	Header	"application/x-www-form-urlencoded"
grant_type	Yes	payload	"urn:ietf:params:oauth:grant-type:jwt-bearer"
scope	Yes	payload	The hostname of the account

Response:

Field	Type	Description
token	string	The scoped token.

Open Channel

The `Open Channel` operation creates or opens a new channel against a pipe or table. If the channel already exists, Snowflake bumps the client sequencer of the channel and returns the last committed offset token.

```
PUT /v2/streaming/databases/{databaseName}/schemas/{schemaName}/pipes/{pipeName}/channels/{channelName}
```

Request:

Attribute	Required	Component	Description
databaseName	Yes	URI	Database name, case-insensitive
schemaName	Yes	URI	Schema name, case-insensitive
pipeName	Yes	URI	Pipe name, case-insensitive
channelName	Yes	URI	The name of the channel that you create or re-open, case-insensitive
offset_token	No	payload	String used to set an offset token when opening a channel.
requestId	No	Query parameter	UUID used to track requests through the system

Response:

Field	Type	Description
next_continuation_token	String	An API-managed token that must be used in the subsequent Append Rows request. The token links a series of calls, ensuring a contiguous, in-order stream of data and maintaining the session state for exactly-once delivery.
channel_status	Object	<p>A nested object with the following detailed information about the channel:</p> <ul style="list-style-type: none"> • database_name (String): The name of the database where the pipe is located • schema_name (String): The name of the schema where the pipe is located • pipe_name (String): The name of the specific pipe being used. • channel_name (String): The name of the streaming channel. • channel_status_code (String): A code that indicates the current status of the channel; for example, "ACTIVE". • last_committed_offset_token (String): The token that represents the last successfully committed offset. • created_on_ms (Long): The timestamp, in milliseconds, when the channel was created. • rows_inserted (Int): The total number of rows successfully inserted. • rows_parsed (Int): The total number of rows parsed. • rows_error_count (Int): The total number of rows that encountered an error. • last_error_offset_upper_bound (String): A token that indicates the upper bound of the offset where the last error occurred. • last_error_message (String): The message of the last error that occurred. • last_error_timestamp (Long): The timestamp, in

milliseconds, of the last error.

- `snowflake_avg_processing_latency_ms` (Int): The average processing latency of Snowflake in milliseconds.

Append Row(s)

The `Append Rows` operation inserts a batch of rows to the given channel.

```
POST /v2/streaming/data/databases/{databaseName}/schemas/{schemaName}/pipes/{pipeName}
```

Request:

Attribute	Required	Component	Description
<code>databaseName</code>	Yes	URI	Database name, case-insensitive
<code>schemaName</code>	Yes	URI	Schema name, case-insensitive
<code>pipeName</code>	Yes	URI	Pipe, case-insensitive
<code>channelName</code>	Yes	URI	Channel name, case-insensitive
<code>continuationToken</code>	Yes	Query parameter	Continuation token from Snowflake, encapsulates both client and row sequencers
<code>offsetToken</code>	No	Query parameter	String used to set an offset token per batch.
<code>rows</code>	Yes	payload	The actual data payload to be ingested in NDJSON format.
<code>requestId</code>	No	Query	A UUID used to track requests through the

parameter system.

Note

The JSON text within the NDJSON payload must strictly conform to the `RFC 8259` standard. Each JSON text must be followed by a newline character `\n` (`0x0A`). You can also insert a carriage return `\r` (`0x0D`) before the newline character.

Response:

Field	Type	Description
next_continuation_token	string	The next continuation token from Snowflake, which encapsulates both client and row sequencers. It should be used for inserting the next batch.

Drop Channel

The `Drop Channel` operation drops a channel at server side along with its metadata.

```
DELETE /v2/streaming/databases/{databaseName}/schemas/{schemaName}/pipes
```

Request:

Attribute	Required	Component	Description
databaseName	Yes	URI	Database name, case-insensitive
schemaName	Yes	URI	Schema name, case-insensitive
pipeOrTableName	Yes	URI	Pipe or table name, case-insensitive

channelName	Yes	URI	Channel name, case-insensitive
requestId	No	Query parameter	A UUID used to track requests through the system

Response:

This operation returns a payload with no specific successful response other than the HTTP status code.

Bulk Get Channel Status

The `Bulk Get Channel Status` operation returns the status of a channel for a specific client sequencer.

```
POST /v2/streaming/databases/{databaseName}/schemas/{schemaName}/pipes/
```

Request:

Attribute	Required	Component	Description
databaseName	Yes	URI	Database name, case-insensitive
schemaName	Yes	URI	Schema name, case-insensitive
pipeName	Yes	URI	Pipe name, case-insensitive
channel_names	Yes	Payload	An array of String channel names that the customer wants to get status for; the names are case-sensitive. For example, <code>{"channel_names": ["channel1", "channel2"]}</code> .

Response:

```

{
  "channel_statuses": {
    "channel1": {
      "channel_status_code": "String",
      "last_committed_offset_token": "String",
      "database_name": "String",
      "schema_name": "String",
      "pipe_name": "String",
      "channel_name": "String",
      "rows_inserted": "int",
      "rows_parsed": "int",
      "rows_errors": "int",
      "last_error_offset_upper_bound": "String",
      "last_error_message": "String",
      "last_error_timestamp": "timestamp_utc",
      "snowflake_avg_processing_latency_ms": "int"
    },
    "channel2": {
      "comment": "same structure as channel1"
    }
  }
  "comment": "potentially other channels"
}

```

Note

If no requested channel is found in the service, the response payload doesn't have an entry for that channel within the `channel_statuses` object.

Description of `channel_statuses` fields for each channel:

Field	Type	Description
channel_status_code	String	Indicating the status of the channel.
last_committed_offset_token	String	Latest committed offset token
database_name	String	The name of the database that the channel belongs to.

schema_name	String	The name of the schema that the channel belongs to.
pipe_name	String	The name of the pipe that the channel belongs to.
channel_name	String	The name of the channel.
rows_inserted	int	A count of all rows inserted into this channel.
rows_parsed	int	A count of all rows parsed, but not necessarily inserted into this channel.
rows_errors	int	A count of all rows that experienced errors when inserting into this channel and were therefore rejected.
last_error_offset_upper_bound	String	Upper bound for the latest offset token of the inserted row set that has the last corresponding to an error. The real offset token of the row set with the last error is either this one or strictly before it in the channel ingestion order.
last_error_message	String	A human readable message corresponding to the latest error code for that channel, with sensitive customer data redacted.
last_error_timestamp	timestamp_utc	Timestamp at which the last error occurred.
snowflake_avg_processing_latency_ms	int	Average e2e processing time for this channel.

Error response structure

You should see the following JSON payload shape for error responses from all APIs:

```
{  
  "error_code": "",  
  "message": ""  
}
```