pubsubplus-connector-nats-js

User Guide

Solace Corporation

Version 4.0.1

solace.

Table of Contents

Preface . . . 1
Configuring Micro Integration . . . 2
  Nats JetStream Connection Details . . . 2
Message Mapping & Transforms . . . 3
Management and Monitoring Micro Integration . . . 4
Configuration . . . 5
Configuration Source Connector . . . 6
  Nats Jetstream Configuration Options - For Source connector (Nats JS Consumer) . . . 6
Headers Source Connector . . . 12
  Nats Jetstream Headers - For Source connector (Nats JS Consumer) . . . 12
Configuration Sink Connector . . . 13
  Nats Jetstream Configuration Options - For Sink connector (Nats JS Producer) . . . 13
License . . . 16
Support . . . 17

Preface

The Solace NATS Jetstream Micro-Integration provides the ability to seamlessly integrate NATS Jetstream with Solace Event Broker and the broader Solace Event Mesh. This integration enables real-time, bidirectional data flow between NATS-powered applications and event-driven ecosystems powered by Solace. By bridging these technologies, organizations can extend the reach of their NATS-based systems, ensuring reliable message delivery, advanced event routing, and dynamic topic hierarchies across distributed environments. This capability is essential for enterprises looking to unify disparate event sources, enhance data streaming capabilities, and support hybrid or multi-cloud architectures within their event-driven infrastructure.

- Getting Started
- Resource Requirements for Self-Managed Micro-Integrations
- Enabling Workflows

Configuring Micro Integration

- Configuring the Datapath for Your Self-Managed Micro-Integration

Nats JetStream Connection Details

The Spring Cloud Stream Binder for Nats JetStream uses the following configuration to configure its session

In the `application.yml` for the source connector, this typically would be configured as follows:

```
spring.cloud.stream.nats-js.default.consumer:
  natsURL: nats://user:pass@localhost:4222
  natsNkey: /path/to/nkey/file/optional
  natsCreds: /path/to/jwt_cred_file/optional
  consumerName: pubsubplus-connector-nats-js
  queueGroupName: OptionalQueueName
```

In the `application.yml` for the sink connector, this typically would be configured as follows:

```
spring.cloud.stream.nats-js.default.producer:
  natsURL: nats://user:pass@localhost:4222
  natsNkey: /path/to/nkey/file/optional
  natsCreds: /path/to/jwt_cred_file/optional
  asyncPublish: true
```

In case of Docker deployment the json file can be placed in config folder and the config path relevant to docker container is used to point the file location.

solace.

Message Mapping & Transforms

Mapping Message Headers and Payloads

Management and Monitoring Micro Integration

Managing and Deploying Your Self-Managed Micro-Integration

Configuration

Configuring Self-Managed Micro-Integrations

Workflow Configuration

Configuration Source Connector

Nats Jetstream Configuration Options - For Source connector (Nats JS Consumer)

These configuration options are all prefixed by `spring.cloud.stream.nats-js.<binding>.consumer`:

Config Option	Type	Valid Values	Default Value	Description
`trace`	`boolean`	`true, false`	`false`	If set to true, traceConnection property is enabled for Nats JS connection
`natsURL`	`String`	`URL formats [nats://localhost:4222, nats://user:pass@localhost:4222]`	`empty`	NATS URL for connection. Single or multiple url's (commma, separated) can be configured
`connectionName`	`string`	`any`	`empty`	Set connection name property option in the NATS JS connection
`username`	`string`	`any`	`empty`	Set username property option in the NATS JS connection. Can also be provided directly in the URL Format
`password`	`string`	`any`	`empty`	Set password property option in the NATS JS connection. Can also be provided directly in the URL Format
`authToken`	`string`	`any`	`empty`	Authentication Token
`nkeyFilePath`	`string`	`any`	`empty`	Path to nKey File for Authentication
`credsFilePath`	`string`	`any`	`empty`	Path to Creds file for Authentication
`tls.enabled`	`boolean`	`true, false`	`false`	If set to true, SSL connection will be established to NATS server

solace.

Config Option	Type	Valid Values	Default Value	Description
tls.opentls	boolean	true, false	false	If set to true, Open TLS will be enabled and trust store verification for the Server CA root will not be required
tls.keyStore	string	any	empty	Path to Key Store, if 2-way SSL is required to be configured on client side.
tls.keyStorePassword	string	any	empty	Password to Key Store, if 2-way SSL is required to be configured on client side.
tls.trustStore	string	any	empty	Path to Trust Store, if opentls is false and the server certification verification is required. Install Server Root CA in the trust store
tls.trustStorePassword	string	any	empty	Password to Trust Store
tls.algorithm	string	any	SunX509	TLS Algorithm
connectionTimeout	int	Integer Value > 0	10	Connection Timeout in seconds
pingInterval	int	Integer Value > 0	20	Ping Interval in case of idle in seconds to keep the connection open
reconnectWait	int	Integer Value > 0	3	Reconnect wait time in seconds
reconnectWait	int	Integer Value	-1	Reconnect max count. Default is -1, unlimited
consumerName	string	any	empty	Name of the Consumer for Source Connector

Config Option	Type	Valid Values	Default Value	Description
subscribeSubject	string	any	">" wildcard for all subjects	Filter subjects inside a Stream. Default value ">", means events from all the subjects in a stream would be received. Add multiple subject comma separated for multiple subscriptions per consumer. For example "subject1,subject2,subject3 " (without the spaces after or before a comma)
queueGroupName	string	any	empty	Optional Queue group as a consumer for the Stream.
deleteExistingConsumer	boolean	true, false	false	If set to true, the consumer with the provide consumer name in the consumerName property will be removed and recreated with new consumer configuration options. This is helpful when old consumer name with the same consumerName has configuration conflicts with the new consumer configuration options defined in the configuration file.
maxAckPending	long	Long Value	1000	Maximum number of unacknowledged messages a consumer can have pending before the server pauses delivery, preventing overload.

Config Option	Type	Valid Values	Default Value	Description
deliverPolicy	enum	All, New, Last, ByStartSequence, ByStartTime	New	All: Delivers all messages from the stream's earliest sequence. New: Delivers only messages published after consumer creation. Last: Delivers only the most recent message at consumer creation. ByStartSequence: Delivers messages starting from a specified sequence number. ByStartTime: Delivers messages starting from a specified timestamp.
startSequence	long	Long Value	0	Start sequence value when deliverPolicy is ByStartSequence.
startTimeEpochMilli	long	Long Value	0	Start time epoch value when deliverPolicy is ByStartTime in epoch milliseconds.
startTimeTimezone	string	any	System Default	Start time Timezone value when deliverPolicy is ByStartTime. For example "America/New_York" or "Asia/Tokyo" or "GMT+05:30"
subscriptionMode	enum	PUSH, PULL	PUSH	Push Subscriber: Automatically receives messages from the NATS server as they are published, using a dispatcher to route them to a message handler. Pull Subscriber: Actively fetches batches of messages from the stream on demand, controlled by application logic with configurable batch size and timeout.

Config Option	Type	Valid Values	Default Value	Description
pushOptions.flowControl	boolean	true, false	false	If set to true, Enables flow control for push subscriptions, allowing the server to pause message delivery when the consumer is overloaded.
pushOptions.idleHeartbeatMilli	long	Long Value	0	Sets the interval (in milliseconds) for server-sent heartbeats to keep push subscription connections alive during low message activity. Default value 0 means disabled.
pullOptions.batchSize	int	Integer Value	10	Number of messages fetched at once by a pull subscriber in a single request (default 10).
pullOptions.batchTimeoutMilli	long	Long Value	100	Maximum time (in milliseconds) a pull subscriber waits for a batch of messages (default 100).
pullOptions.pullIntervalMilli	long	Long Value	0	Delay (in milliseconds) between consecutive pull requests for a pull subscriber (default 0, immediate). Together with the combination of these 3 values (batchSize, batchTimeoutMilli, pullIntervalMilli) the throughput, performance and load on the Nats server can be controlled when using PULL subscriptionMode

Config Option	Type	Valid Values	Default Value	Description
dlqEnabled	boolean	true, false	false	If set to true, failed messages will be acked as Term which Instructs the server to stop redelivery of a message without acknowledging it as successfully processed. If set to false, Nak is sent and Signals that the message will not be processed now and processing can move onto the next message, NAK'd message will be retried
jetStreamDomain	string	any	empty	Optional Jetstream Domain name. If not defined NATS Jetstream default is used.
jetStreamPrefix	string	any	empty	Optional Jetstream Prefix.
jetStreamRequestTimeoutMs	int	Integer Value	3000	Jetstream Request Timeout in milliseconds.

Headers Source Connector

Nats Jetstream Headers - For Source connector (Nats JS Consumer)

The following list of Nats Jetstream headers and meta data params are handled by the connector

Name	Header Key	Type	Description
Nats-Subject	njs_subject	Header	The original subject of the message.
Nats-Reply-To	njs_replyTo	Header	Reply To Subject
Nats-Is-Jetstream	njs_isJetStream	Header	Messages consumed in JetStream
Nats-Consumer-Byte-Count	njs_consumeByteCount	Header	Consume Byte Count in bytes
Nats-SID	njs_sid	Header	Nats SID
Nats-Stream	njs_stream	MetaData	Name of the stream the message was republished from.
Nats-Domain	njs_domain	MetaData	Domain value
Nats-Consumer	njs_consumer	MetaData	Consumer Value
Nats-Delivered-Count	njs_delivered	MetaData	Delivered Count
Nats-Sequence	njs_streamSeq	MetaData	Stream Sequence
Nats-Consumer-Sequence	njs_consumerSeq	MetaData	Consumer Sequence
Nats-Time-Stamp	njs_timestamp	MetaData	The original timestamp of the message.
Nats-Pending-Count	njs_pending	MetaData	Pending Count

solace.

Configuration Sink Connector

Nats Jetstream Configuration Options - For Sink connector (Nats JS Producer)

These configuration options are all prefixed by `spring.cloud.stream.nats-js.<binding>.producer`:

Config Option	Type	Valid Values	Default Value	Description
`trace`	`boolean`	`true, false`	`false`	If set to true, traceConnection property is enabled for Nats JS connection
`natsURL`	`String`	`URL formats [nats://localhost:4222, nats://user:pass@localhost:4222]`	`empty`	NATS URL for connection. Single or multiple url's (commma, separated) can be configured
`connectionName`	`string`	`any`	`empty`	Set connection name property option in the NATS JS connection
`username`	`string`	`any`	`empty`	Set username property option in the NATS JS connection. Can also be provided directly in the URL Format
`password`	`string`	`any`	`empty`	Set password property option in the NATS JS connection. Can also be provided directly in the URL Format
`authToken`	`string`	`any`	`empty`	Authentication Token
`nkeyFilePath`	`string`	`any`	`empty`	Path to nKey File for Authentication
`credsFilePath`	`string`	`any`	`empty`	Path to Creds file for Authentication
`tls.enabled`	`boolean`	`true, false`	`false`	If set to true, SSL connection will be established to NATS server

Config Option	Type	Valid Values	Default Value	Description
tls.opentls	boolean	true, false	false	If set to true, Open TLS will be enabled and trust store verification for the Server CA root will not be required
tls.keyStore	string	any	empty	Path to Key Store, if 2-way SSL is required to be configured on client side.
tls.keyStorePassword	string	any	empty	Password to Key Store, if 2-way SSL is required to be configured on client side.
tls.trustStore	string	any	empty	Path to Trust Store, if opentls is false and the server certification verification is required. Install Server Root CA in the trust store
tls.trustStorePassword	string	any	empty	Password to Trust Store
tls.algorithm	string	any	SunX509	TLS Algorithm
connectionTimeout	int	Integer Value > 0	10	Connection Timeout in seconds
pingInterval	int	Integer Value > 0	20	Ping Interval in case of idle in seconds to keep the connection open
reconnectWait	int	Integer Value > 0	3	Reconnect wait time in seconds
reconnectWait	int	Integer Value	-1	Reconnect max count. Default is -1, unlimited
asyncPublish	boolean	true, false	true	Publish to NATS in async Mode. Only applicable for Sink Connector
ackWaitTimeoutMilli	int	Integer Value	5000	Maximum acknowledgement wait time in milliseconds from NATS Jetstream server for published messages asynchronously. Default is 5000ms

Config Option	Type	Valid Values	Default Value	Description
ackWaitTimeoutGraceMilli	int	Integer Value	1000	Additional time (in milliseconds) a consumer waits beyond the ackWait period for message acknowledgment before redelivering, providing a grace period to handle processing delays.
jetStreamDomain	string	any	empty	Optional Jetstream Domain name. If not defined NATS Jetstream default is used.
jetStreamPrefix	string	any	empty	Optional Jetstream Prefix.
jetStreamRequestTimeoutMs	int	Integer Value	3000	Jetstream Request Timeout in milliseconds.

License

See the license terms that apply to Self-Managed Micro-Integrations.

Support

See Support for how to get help or report an issue with a Self-Managed Micro-Integration.