

# Ordered Consumer

---

Metadata	Value
Date	2021-09-29
Author	@scottf
Status	Implemented
Tags	jetstream,client

## Context and Problem Statement

Provide an ordered push subscription for the user that automatically checks and recovers when a gap occurs in the consumer sequence. The subscription can deliver messages synchronously or asynchronously as normally supported by the client.

## Behavior

The subscription should leverage Gap Management and Auto Status Management to ensure messages are received in the proper order.

The subscription must track the last good stream and consumer sequences. When a gap is observed, the subscription closes its current subscription, releases its consumer and creates a new one starting at the proper stream sequence.

If hearbeats are missed, consumer might be gone (deleted, lost after reconnect, node restart, etc.), and it should be recreated from last known stream sequence.

You can optionally make the "state" available to the user.

## Subscription Limitations

The subscription cannot be for

- a pull consumer
- a durable consumer
- cannot bind or be "direct"

The subscription is not allowed with queues/deliver groups.

## Consumer Configuration Checks

The user can provide a consumer configuration but it must be validated. Error when creating if validation fails.

Checks:

- durable\_name: must not be provided
- deliver\_subject: must not be provided
- ack policy: must not be provided or set to none. Set it to none if it is not provided.

- `max_deliver`: must not be provided or set to 1. Set it to 1 if it is not provided.
- `flow_control`: must not be provided or set true. Set it to true if it is not provided.
- `mem_storage`: must not be provided or set to true. Set to true if it is not provided.
- `num_replicas`: must not be provided. Set to 1.

Check and set these settings without an error:

- `idle_heartbeat` if not provided, set to 5 seconds
- `ack_wait` set to something large like 22 hours (Matches the go implementation)