



This PDF is generated from authoritative online content, and is provided for convenience only. This PDF cannot be used for legal purposes. For authoritative understanding of what is and is not supported, always use the online content. To copy code samples, always use the online content.

# Integrated Capture Points Guide

Kafka Capture Point - Topic Partitioning

12/22/2025

---

## Contents

- 1 Kafka Capture Point - Topic Partitioning
  - 1.1 Inbound topic partitioning
  - 1.2 Notification topic partitioning

# Kafka Capture Point - Topic Partitioning

Each Kafka topic contains one or more partitions. In a practical sense, a partition is a minimum processing unit. So, when working with Kafka, special attention should be paid to the topic partitioning because it influences message handling strategies.

## Inbound topic partitioning

This section provides information on inbound topic partitioning.

### Number of partitions

For each partition of the inbound topic, Kafka Capture Point has only the offset, index of the processed messages. This is the standard behavior for Kafka consumers. This is a simple and reliable way of working, but it limits the number of messages that can be processed in parallel. The parallel processing of messages is the straightforward way to speed up processing (up to some level). So, the inbound topic partitions number is the effective hard limit for the Kafka Capture Point receiving threads number and for performance of Kafka Capture Point. Each receiving thread is capable of processing messages from several partitions.

The number of the inbound topic partitions should be chosen with some reserve. The recommended number is 32 or 64.

### Message partitioning

Another aspect of the partitioning is the way how the messages are spread across partitions. The usual requirement is to process requests related to an interaction sequentially, one at a time. Essentially, this means all such requests should be placed in one partition. This can be achieved by specifying the same partition key for these requests. Kafka uses partition keys to select partitions to put message to. Thus, specifying the same partition keys guarantees messages are placed in the same partition. Combining this with the processing of one request at a time by receiving thread of the Kafka Capture Point allows to satisfy the given requirement.

The natural candidates for the partition key are the value of Interaction ID generated by Interaction Server and the value of External ID supplied to Interaction Server from external system.

If the requirement is not needed or satisfied by other means, the partition key of the inbound messages can be left empty.

## Notification topic partitioning

This section provides information on notification topic partitioning.

### Number of partitions

The number of the notification topic partitions does not have any effect on Kafka Capture Point. So, it is not limited from the Kafka Capture Point side. This number should be chosen in accordance with the requirements and peculiarities of the consumers of this topic. But most probably, the recommendation for inbound topic partitions is applicable here as well.

The same applies to partitioning of error and processed topics.

### Message partitioning

There are two kinds of messages that are sent to the notification topic:

- Replies to inbound requests
- Unsolicited notifications

The value of `Interaction ID` of the reply or the unsolicited notification is used as the partition key for the notification topic. If the value is not present in the reply, then the partition key supplied in the inbound request is reused for the reply.