FLOWX.AI

BUILDING BLOCKS / Node / Message events / message-catch-start-event

Contents

- BUILDING BLOCKS / Node / Message events / Message Catch Start Event
  - Configuring a Message Catch Start Event

BUILDING BLOCKS / Node / Message events / Message Catch Start Event

QUICK INTRO

What is it? It represents the starting point for a process instance based on the receipt of a specific message. When this event is triggered by receiving the designated message, it initiates the execution of the associated process.

Why it is important? The Message Catch Start Event is important because it allows a process to be triggered and initiated based on the reception of a specific message.

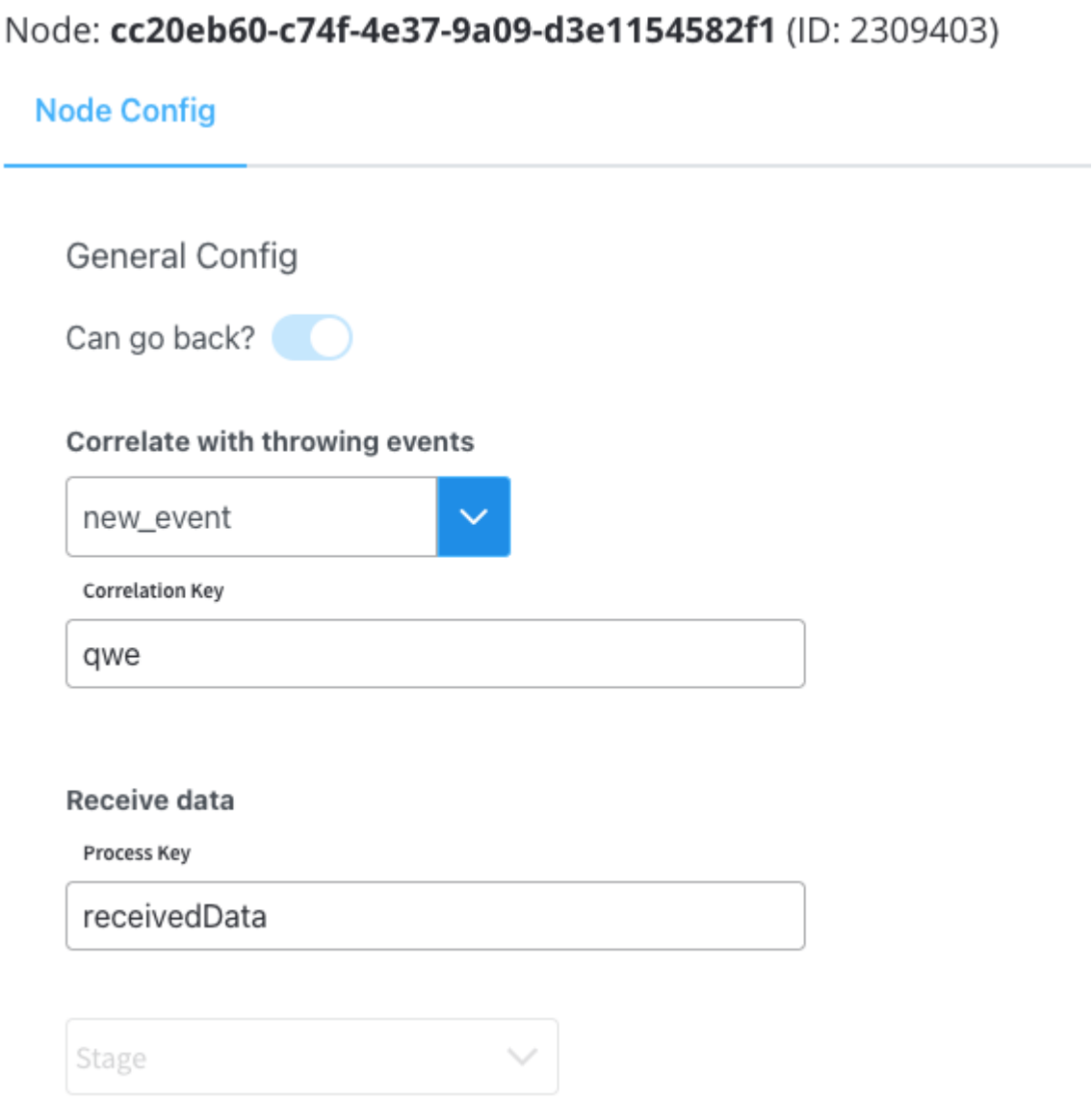
Configuring a Message Catch Start Event

A Message Catch Start Event is a special event in a process that initiates the start of a process instance upon receiving a specific message. It acts as the trigger for the process, waiting for the designated message to arrive. Once the message is received, the process instance is created and begins its execution, following the defined process flow from that point onwards. The Message Catch Start Event serves as the entry point for the process, enabling it to start based on the occurrence of the expected message.

> **CAUTION**
>
> It is mandatory that in order to use this type of node together with task management plugin, to have a service account defined in your identity solution. For more information, check our documentation in how to create service accounts using Keycloak, **here**

General config

- **Can go back?** - setting this to true will allow users to return to this step after completing it, when encountering a step with `canGoBack` false, all steps found behind it will become unavailable
- **Correlate with catch events** - the dropdown contains all catch messages from the process definitions accessible to the user
- **Correlation key** - is a process key that uniquely identifies the instance to which the message is sent
- **The data field** - allows the user to define a JSON structure with the data to be sent along with the message
- **Stage** - assign a stage to the node



Was this page helpful?