Structured Output Parser node common issues

# Structured Output Parser node common issues#

Here are some common errors and issues with the Structured Output Parser node and steps to resolve or troubleshoot them.

## Processing parameters#

The Structured Output Parser node is a sub-node. Sub-nodes behave differently than other nodes when processing multiple items using expressions.

Most nodes, including root nodes, take any number of items as input, process these items, and output the results. You can use expressions to refer to input items, and the node resolves the expression for each item in turn. For example, given an input of five name values, the expression {{ $json.name }} resolves to each name in turn.

{{ $json.name }}

In sub-nodes, the expression always resolves to the first item. For example, given an input of five name values, the expression {{ $json.name }} always resolves to the first name.

{{ $json.name }}

## Adding the structured output parser node to AI nodes#

You can attach output parser nodes to select AI root nodes.

To add the Structured Output Parser to a node, enable the Require Specific Output Format option in the AI root node you wish to format. Once the option is enabled, a new output parser attachment point is displayed. Click the output parser attachment point to add the Structured Output Parser node to the node.

## Using the structured output parser to format intermediary steps#

The Structured Output Parser node structures the final output from AI agents. It's not intended to structure intermediary output to pass to other AI tools or stages.

To request a specific format for intermediary output, include the response structure in the System Message for the AI Agent. The message can include either a schema or example response for the agent to use as a template for its results.

## Structuring output from agents#

Structured output parsing is often not reliable when working with agents.

If your workflow uses agents, n8n recommends using a separate LLM-chain to receive the data from the agent and parse it. This leads to better, more consistent results than parsing directly in the agent workflow.