# Context Handling in LLM + Autogen + Stategraph Approach

In the LLM + Autogen + Stategraph approach, context plays a critical role in coordinating tasks and ensuring consistent execution across components.   
This document explains how context is managed for each agent and task, showing how both shared and centralized contexts are updated as the workflow progresses.

## 1. Types of Context

There are two main types of context in this approach:  
1. \*\*Shared Context\*\*: A task-specific snapshot of data that is passed between the LLM, Autogen, and Stategraph to ensure they operate consistently.  
2. \*\*Centralized Context\*\*: A global repository managed by Autogen that tracks the overall workflow state, including completed tasks, pending tasks, results, and errors.

## 2. How Context Flows Across Components

### A. LLM

The LLM uses the shared context to predict the next step in the workflow. It considers the current workflow state, completed steps, and intermediate results to make informed predictions.

\*\*Example Context Provided to the LLM:\*\*

{  
 "document\_type": "claim",  
 "completed\_steps": ["Validation"],  
 "results": {  
 "Validation": {"status": "success", "details": "Document is valid"}  
 },  
 "pending\_tasks": ["Triage", "Eligibility", "Pricing", "Decision"]  
}

\*\*LLM Output:\*\* 'Triage' as the next step.

### B. Autogen

Autogen uses the centralized context to:  
1. Delegate tasks to the appropriate agents or tools.  
2. Resolve dependencies between tasks dynamically.  
3. Handle retries and fallback paths in case of errors.  
4. Update the centralized context after each task execution.

\*\*Example Centralized Context Managed by Autogen:\*\*

{  
 "document\_type": "claim",  
 "completed\_steps": ["Validation", "Triage"],  
 "results": {  
 "Validation": {"status": "success", "details": "Document is valid"},  
 "Triage": {"type": "claim", "intent": "process\_claim", "details": "Claim details"}  
 },  
 "pending\_tasks": ["Eligibility", "Pricing", "Decision"],  
 "errors": []  
}

### C. Stategraph

The Stategraph reads from the shared context to execute tasks and determine the appropriate transitions between states. After task execution, it updates the shared context with the results.

\*\*Example Usage in Stategraph (Eligibility Step):\*\*

eligibility\_details = shared\_context["results"].get("Triage")  
if not eligibility\_details:  
 raise ValueError("Triage details are missing!")  
  
result = EligibilityTool().run(eligibility\_details)  
  
# Update shared context  
shared\_context["results"]["Eligibility"] = {"status": "success", "eligibility": True}  
shared\_context["completed\_steps"].append("Eligibility")

## 3. Combined Flow of Context Updates

Here is how the context evolves as the workflow progresses:

#### Step 1: Validation

Initial Context:  
{  
 "document\_type": "claim",  
 "completed\_steps": [],  
 "results": {},  
 "pending\_tasks": ["Validation", "Triage", "Eligibility", "Pricing", "Decision"],  
 "errors": []  
}  
  
After Validation:  
{  
 "document\_type": "claim",  
 "completed\_steps": ["Validation"],  
 "results": {  
 "Validation": {"status": "success", "details": "Document is valid"}  
 },  
 "pending\_tasks": ["Triage", "Eligibility", "Pricing", "Decision"],  
 "errors": []  
}

#### Step 2: Triage

Before Triage:  
{  
 "completed\_steps": ["Validation"],  
 "results": {  
 "Validation": {"status": "success", "details": "Document is valid"}  
 },  
 "pending\_tasks": ["Triage", "Eligibility", "Pricing", "Decision"]  
}  
  
After Triage:  
{  
 "completed\_steps": ["Validation", "Triage"],  
 "results": {  
 "Validation": {"status": "success", "details": "Document is valid"},  
 "Triage": {"type": "claim", "intent": "process\_claim", "details": "Claim details"}  
 },  
 "pending\_tasks": ["Eligibility", "Pricing", "Decision"],  
 "errors": []  
}

#### Step 3: Eligibility Check

Before Eligibility:  
{  
 "completed\_steps": ["Validation", "Triage"],  
 "results": {  
 "Validation": {"status": "success", "details": "Document is valid"},  
 "Triage": {"type": "claim", "intent": "process\_claim", "details": "Claim details"}  
 },  
 "pending\_tasks": ["Eligibility", "Pricing", "Decision"]  
}  
  
After Eligibility:  
{  
 "completed\_steps": ["Validation", "Triage", "Eligibility"],  
 "results": {  
 "Validation": {"status": "success", "details": "Document is valid"},  
 "Triage": {"type": "claim", "intent": "process\_claim", "details": "Claim details"},  
 "Eligibility": {"status": "success", "eligibility": True}  
 },  
 "pending\_tasks": ["Pricing", "Decision"],  
 "errors": []  
}