DispatchTrack

White Paper

Multi-Agent AI for Last Mile Customer Delivery Experience

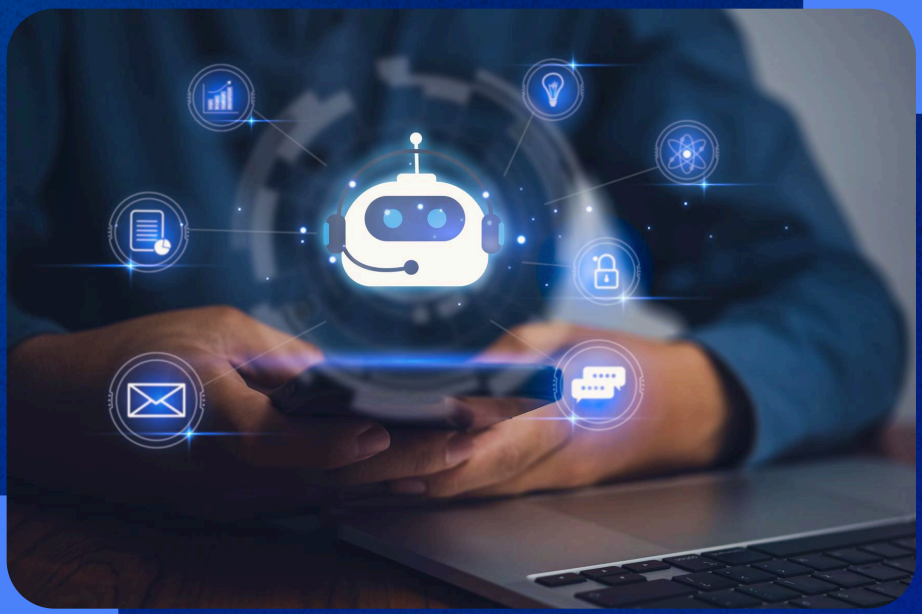


Table of Contents

01 Intro: Are Your Customers Already Sick of AI Chatbots?

02 Defining Multi-Agent Architecture: What Is It and Why Does It Matter?

03 Overcoming the Limitations of Traditional Chatbots Using Multiple Agents

04 How Is DispatchTrack Leveraging Multi-Agent Architecture to Resolve Customer Requests?

05 How Does This Framework Impact Delivery Metrics?

06 Conclusion: The Power of Moving Beyond Traditional Chatbots in Logistics

Introduction: Are Your Customers Already Sick of AI Chatbots?

We know that slow answers and long hold times on the phone don't satisfy customers in the delivery space, regardless of whether they're end consumers or business buyers like general contractors and grocery store managers. At the same time, studies have shown that **70% of customers** prefer to speak to a human being when they have questions.

With costs rising across the board and customer expectations inflating, this puts delivery businesses in a quandary: how are they supposed to operate efficiently without sacrificing attentive, expert-level customer support?

AI chatbots have frequently been pitched as the solution to this problem, especially around high volumes of WISMO ("where is my order?") calls, rescheduling requests, and frequent customer clarifications and requests. To be sure, they have their value there—but they have real limitations as a solution. A typical chatbot's performance suffers as an organization's requirements become more complex, leading to hallucinations and errors that negatively impact customers.

This is where the frustration on the customer's side shown in the data (and the hit to your CSAT score that it causes), can start to outweigh the increased efficiency and time savings that come from automatic answers to WISMO questions.

What's the solution? Upleveling AI deployments in customer delivery experience contexts from simple LLM-powered chatbots to multi-agent AI deployments that can handle more complex uses.

This has some distinct advantages over the traditional approach:

More specialized agents with tighter context windows are less likely to hallucinate than a single agent trying to act as a jack of all trades.

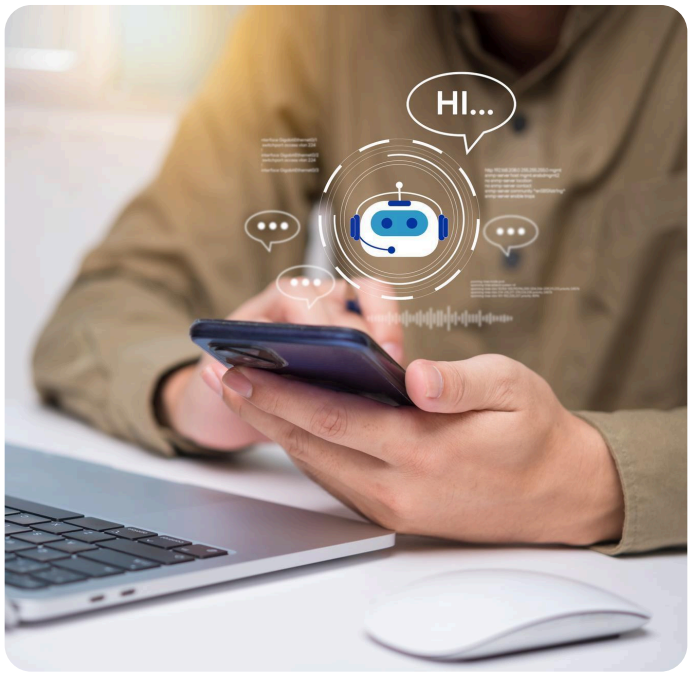
Multi-agent deployments are more easily scaled—just add new agents to the system that can handle new customer requests or requirements.

With a wider variety of specialized agents, you can more easily and confidently offer customers the ability to schedule, reschedule, and adjust orders without human intervention.

Chatbots are fundamentally reactive, whereas a multi-agent system can call in agents proactively if they determine that more information is needed from the end customer.

In this white paper, we'll provide a deep dive into how multi-agent AI can be leveraged in the context of customer delivery experience. We'll cover the advantages of this framework over traditional chatbots and the benefits that logistics operators can gain from deploying technology that leverages it.

Readers will come away armed with the knowledge they need about this topic to make informed decisions on how to evaluate and employ multi-agent AI for customer delivery experience. You'll also walk away with practical use cases and examples based on AI deployments on the DispatchTrack platform.



02 Defining Multi-Agent Architecture: What Is It and Why Does It Matter?

To effectively weigh the pros and cons of different AI architectures for your own, some grounding in the key concepts in AI can be helpful:

01 AI Agent:

"AI agents are software systems that use AI to pursue goals and complete tasks on behalf of users. They show reasoning, planning, and memory and have a level of autonomy to make decisions, learn, and adapt. Their capabilities are made possible in large part by the multimodal capacity of generative AI and AI foundation models."

02 Multi-Agent System:

"A multi-agent system consists of multiple artificial intelligence agents working collectively to perform tasks on behalf of a user or another system. Each agent within a MAS has individual properties but all agents behave collaboratively to lead to desired global properties."

Both the individual agents and the multi-agent systems are fundamentally grounded in the same kind of LLMs, but they're organized differently to achieve different results. While a single agent can act as a jack of all trades, it might be a master of none—and when it runs into problems that it can't solve, it may hallucinate or it may introduce other issues into the workflow.

You might remember the attempt by Anthropic's Claude to run a convenience store from a couple of years ago. It had a task that seemed simple to a human—taking orders from Anthropic employees in the office and placing orders with a "supplier" to stock the shelves of the convenience store (actually the office fridge)—but things rapidly spiralled out of control. As TechCrunch put it at the time: "Claudius, believing itself to be a human, told customers it would start delivering products in person, wearing a blue blazer and a red tie."

More recently, an OpenClaw AI agent tried to delete everything in a Meta AI security researcher's email inbox. She reports having to "RUN to [her] Mac mini like [she] was defusing a bomb" to prevent her data from vanishing.

Of course, these are extreme examples. And In many use cases, this is a worthwhile tradeoff to make for simplicity's sake. It's quicker to implement a single agent within a system, and there's a lot that individual agents can do well, including answering simple questions like "where's my order?" But as you ask the agent to do more and more (as the context window grows), the errors can start to pile up.

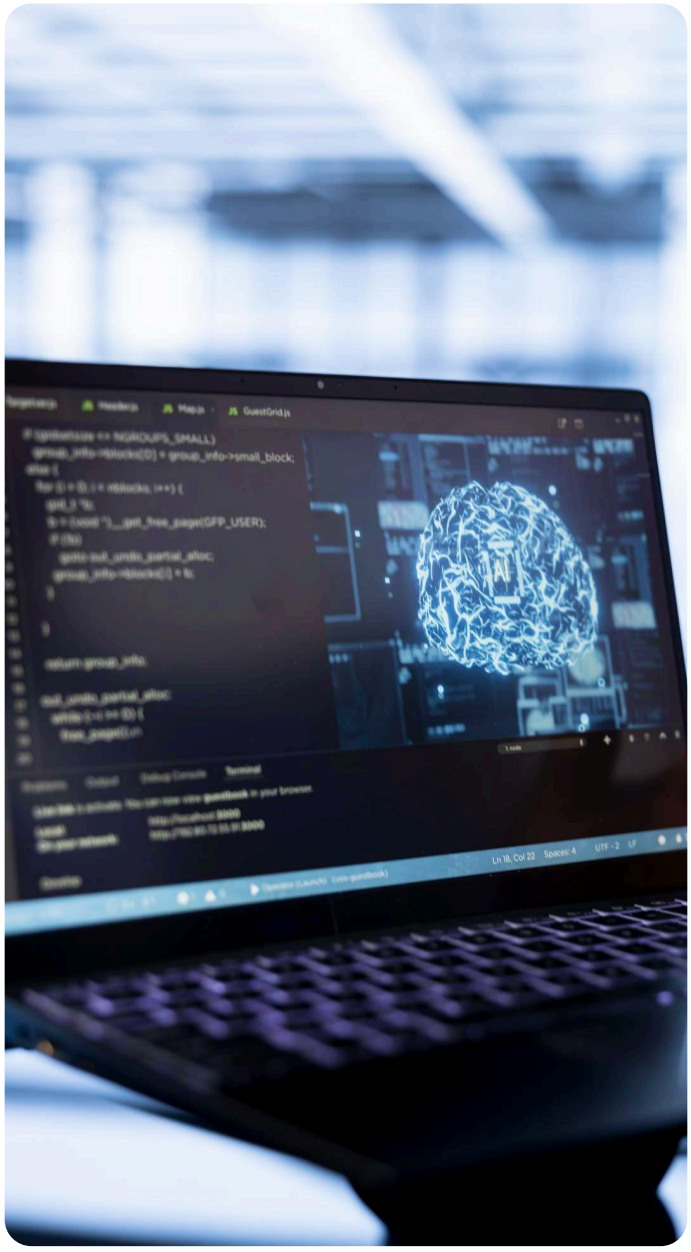
03 Context Window:

"A context window refers to the amount of information a large language model (LLM) can process in a single prompt. Context windows are like a human's short-term memory... So, in Q&A format applications like Anthropic's Claude, OpenAI's ChatGPT, or Google's Gemini, information loaded into a context window is used together with the model's vastly larger pretrained structure to deliver the best responses to user prompts."

In a complicated system like a last mile delivery management system, these errors—e.g. claiming that a reschedule request has gone through when it actually hasn't—can be a huge blow to customer confidence and they can cause chaos from an operational perspective. By leveraging a multi-agent architecture, it's possible to shorten the context window for each agent.

This means that you need more specialized agents, but you can increase the accuracy of each agent's action, decrease hallucinations, and ultimately increase the number of use cases your customer-facing AI can support.

Instead of a jack of all trades, you have a team of experts that can be called up in just the right situations.



03 Overcoming the Limitations of Traditional Chatbots Using Multiple Agents

In a last mile context, multi-agent deployments need to be rooted not just in AI best practices but in deep logistics domain knowledge.

At a high level, **the key to a multi-agent system is the supervisor agent in the middle of the process.** In the case of a last mile operation leveraging AI for delivery experience, the supervisor orchestrates the use of the rest of the team of agents:

- When a customer message comes through, the supervisor analyzes the text and determines from the context of the message which sub-agent needs to be called.
- Once the sub-agent has done its job, it hands the process back to the supervisor, which calls the next relevant sub-agent to keep the job moving.
- If the customer's goal is to reschedule their order, the next agent called might be for checking the existing schedule.

Depending on how complicated the customer's needs are, this can continue through a string of different agents. Each one will be given a small context window and the capabilities for a specialized task within the system, such as scheduling, address verification, etc. One agent never takes on more than it can handle, and the supervisor orchestrates them all seamlessly to provide expert-level support to the customer.

This is what the customer—and the software user—see. Beneath the hood, the supervisor agent will often use deterministic routing logic in addition to reasoning through the problem to improve reliability.

When context is provided to each sub-agent, state management plays a crucial role as well. Maintaining structured states (e.g., order status, customer requests) outside the model helps ensure consistency.

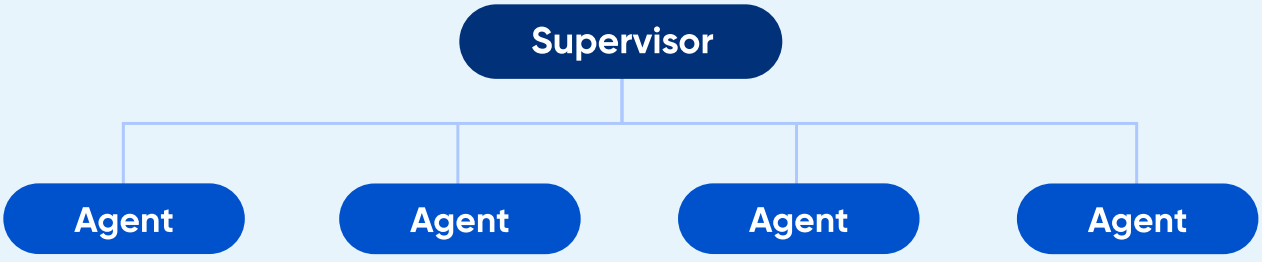
Across the workflow. each agent's outputs should be validated before being executed, to further reduce the risk of incorrect actions.

Advantage of multi-agent systems for complex use cases like last mile logistics:

Traditional chatbots can only handle a limited set of use cases with strong guardrails, or else they can spiral out of control and start to hallucinate. Multi-agent systems with proper grounding enable you to offer a much wider variety of capabilities without risking hallucinations, which in turn improves customer experience. The number of use cases where a customer can get nearly instant gratification grows, and the workload for your teams decreases.

Challenges of integrating multi-agent systems into logistics technology:

Effectively developing a multi-agent system requires you to ensure clearly defined domains for each agent and narrow context windows across the board. The agents can't overlap with each other in terms of what they manage. In short, it's a more technically demanding way to set up a system, requiring both a higher degree of engineering capabilities as well as deeper domain knowledge in logistics. As a logistics operator, it's crucial to be cognizant of the fact that not every software provider can offer this type of architecture and the benefits that come with it.



The supervisor agent routes tasks to the appropriate sub-agent.

04 How Is DispatchTrack Leveraging Multi-Agent Architecture to Resolve Customer Requests?

DispatchTrack's DT Agent is a multi-agent system purpose-built for last mile delivery management. It's been developed specifically to overcome the challenges presented by more limited chatbot deployments and enable last mile delivery and logistics organizations to streamline customer engagement and reduce overall workloads in an efficient and low-risk way.

Our experience:

DispatchTrack's platform is built to handle the entire last mile logistics workflow—including connections to the first and middle mile—from end to end. Orders come into the system, and we schedule, route, dispatch, track, and document them. All the while, our system enables users to connect with customers at numerous touchpoints for scheduling, delivery reminders, ETA updates, and exception management.

In order to build an AI system that could slot into the entire workflow—rather than limiting it to acting as a simple chatbot—we knew we needed a team of specialized sub-agents that could interact with different parts of the system.

By building out those sub-agents to support our users and their end customers—and constraining the sub-agents appropriately and validating their actions—we've been able to improve reliability while increasing the number of tasks that the system can help customers with.

By orchestrating multiple sub-agents for different customer use cases, we're able to support end customers:

- Scheduling their deliveries based on the client's capacity parameters and scheduling rules
- Rescheduling their deliveries within the time windows that the client allows and escalating to a human for last-minute requests that are outside that window
- Sending notes and instructions for the driver or delivery, including information relating to delivery site access
- Providing feedback on the overall quality of the delivery

The system can also act proactively when needed. If we determine based on address verification that the customer is receiving the delivery in an apartment complex, for instance, DT Agent can send a pre-delivery site readiness survey to unearth any potential access issues: unexpected stairs, narrow doorways, etc.

A single AI agent would struggle to context switch between these different tasks, and there would be a high risk of hallucinations and the customer anguish that comes with them.

Notes on our multi-agent system deployment:

Something that we've found to be crucial for providing a high quality experience to delivery customers is the ability to interrupt sub-agents in the middle of processes if the customer's needs change.

This isn't something that all multi-agent systems support, but we've found that the fluid nature of deliveries makes it quite valuable. Customers' needs and requirements can shift quickly, especially as deliveries unfold. The ability to stop sub-agents on a dime and resteer towards the most immediate need helps keep everything running smoothly.



05 How Does This Framework Impact Delivery Metrics?

When we say that we've found the multi-agent architecture to be a better fit for logistics and delivery than traditional AI-powered chatbots, we're grounding that claim in real-world delivery outcomes that matter to distributors.

Here are some of the ways that a system like DT Agent can directly decrease logistics costs, improve customer delivery experience, and logistics visibility:

Reduced workloads for customer support teams.

The more tasks the system is capable of handling without a human in the loop, the more time you can give back to your team. For leaner teams that are already trying to do more with less, this can be a huge boon.

Fewer delivery exceptions across the board.

DT Agent can reduce the amount of friction involved in things like rescheduling, verifying site details, and ensuring successful on-site execution, meaning customers are more likely to be proactive about working with you to ensure smooth deliveries.

Instant, round-the-clock responses to customer questions and requests.

Customers may be leery of chatbots, but they know a good tradeoff when they see one. In this case, the ability to ask a question and receive a response almost instantly at any time of day or night can be a huge boon for customer satisfaction.





Better documentation and delivery intelligence.

Because all of the interactions between customers and DT Agent are centralized in one place, it's easy for the team to drill down into details when an exception or a potential exception arises. DT Agent also helps ensure that the right information gets to the right place at the right time, improving visibility and speeding up exception management.



Greater visibility into customer satisfaction.

You can't optimize what you can't measure—which is why leveraging DT Agent to collect surveys in a conversational manner can be so powerful. It improves the quality of your customer satisfaction data, so you can drill down into potential areas for improvement or increased operational efficiency.



Improved capacity utilization.

By streamlining the customer delivery scheduling process, you can utilize more of your available capacity with the administrative overhead required to manually call customers and try to get their deliveries scheduled. Even when customers need to reschedule, you're able to enforce your capacity rules while offering a completely seamless experience.

These impacts only increase in magnitude as the system is expanded to cover new use cases. At the same time, the impact of DT Agent's capabilities are amplified by underlying logistics functionality designed to simplify last mile management from end to end.

This means that when DT Agent makes promises to customers, your delivery and service teams are actually able to keep them.

Conclusion: The Power of Moving Beyond Traditional Chatbots in Logistics

With the speed that AI moves, it's no wonder that LLM-powered chatbots had such a brief honeymoon. But with customer expectations evolving and the requirements for meeting them becoming more complex, multi-agent AI systems are increasingly the best way forward for improving customer delivery experience at scale.

What does this mean for your business? It means that when you're evaluating new logistics technology deployments, there are a few questions you should be asking:

- Is the software equipped for multi-agent architecture? Does it follow the industry best practices for calling the right sub-agents with the right context windows.
- Can the AI handle interruptions? If the customer changes gears in the middle of a conversation, can the supervisor agent put an existing task on pause?
- What is the technology provider doing to prevent hallucinations and other errors that might impact customer experience? How are agents' actions validated?
- What use cases can the multi-agent system handle by itself? Which use cases require a human in the loop?

This will give you a sense of whether the technology in question is built to actually meet the complex demands of last mile delivery experience management. If it is, you can reduce customer support costs and maintain visibility, all while improving the overall delivery experience offered to the customer.

No matter how the technology landscape changes in the future, that's a sure recipe for logistics success.



About DispatchTrack

DispatchTrack offers logistics native AI built for the last mile. We help top brands around the globe to achieve visibility and efficiency in every mile and deliver delightful, predictable customer experiences. With a comprehensive suite of industry-specific solutions, DispatchTrack's end-to-end capabilities enable logistics teams to achieve increased efficiency, greater visibility and control, and huge improvements in customer satisfaction.

DispatchTrack is trusted by **over 2,500 customers** including industry leaders like Ferguson, Samsung, Ashley Furniture, Mattress Firm, and Ryder.

Scan the QR to Book a Demo:



DispatchTrack

Promise • Deliver • Delight