CONVIVA RESEARCH REPORT

# Agents Burn More Than 2.5 Minutes Gathering Context

Conviva finds consumer-facing AI agents waste two minutes and 32 seconds gathering basic context before they can begin to address a customer's request.

## EXECUTIVE SUMMARY

Conviva analyzed AI agent interactions across major e-commerce and travel booking sites to quantify the cost of deploying agents without behavioral context. In nearly two-thirds of sessions, customers were asked to give information to the agent, including established booking and order information, tasks they'd attempted, and errors they'd hit. This resulted in wasted time, frustrated abandonment, and failed resolutions.

**2:32**
Average time spent establishing context

**3.4**
Average number of turns required for agent to gather context

**8.5%**
Of shoppers abandoned while establishing context or immediately after

## EXAMPLE CONTEXT SETTING CONVERSATION

**Agent** Hello! How can I help you today?

I was trying to add baggage to my flight... **Traveler**

**Agent** Sure! Can you share your booking number?

It's XYZABC. I tried on the app but it failed. **Traveler**

**Agent** What date is your flight?

## FINDINGS

Most consumer-facing AI agents open with a generic greeting and spends three to four turns gathering information that should have already been captured. **Why?** Because web analytics, app telemetry, and the chat agent are disconnected systems. Most agents simply can't see what happens outside the conversation they're currently in.

| | | |
|---|---|---|
| **2:32** | **Time wasted per session** | Average time consumed in context-gathering exchanges before the user's actual request could be addressed. 3.4 turns, taking 2 minutes and 32 seconds, on average. |
| **3.4** | **Turns wasted per session** | 3–4 back-and-forth exchanges about what shoppers already tried, what went wrong, and what they were trying to do. |
| **65%** | **Context re-establishment** | Sessions where the user had to explain what the platform should have known: pages visited, task attempted, order/booking details, error encountered, etc. |
| **5%** | **Abandoned during contect-setting** | Users who became frustrated mid-explanation — before stating their actual request — and left the session entirely. |
| **3.5%** | **Gave context, then abandoned** | Users who completed the full context-setting exchange and received a response, but abandoned anyway. The act of re-explaining drove them out, independent of answer quality. |
| **13.5%** | **Unnecessary clarification** | No-context sessions where the agent probed for information that should have been captured across the shopper's session. |
| **30%** | **Wrong intent assumed** | Instances where the agent misread the user's intent, pursued the wrong resolution path, and ended in session failure. |

## WHAT CHANGES WHEN AN AGENT STARTS WITH CONTEXT

Conviva's Context Graph injects pre-chat behavioral context into the agent's system prompt before the first message is sent. This includes pages visited, tasks attempted, self-service failures, and frustration signals. The agent knows why the user is there before they say a word. The result is that the re-establishment loop is eliminated at the source. Projections below assume approximately 70% reduction in the context gap, based on the share of sessions where behavioral signal was already available.

**Context re-establishment rate** — ↓ 45 percentage points




**Aggregate time wasted** — ↓ 70% reduction




**Wrong intent / session failure rate** — ↓ 21 percentage points




---

## METHODOLOGY

This analysis is based on customer-facing AI agent sessions across e-commerce and travel booking sites, captured and scored using Conviva.

Each session was assigned a Context Score (1–6) reflecting the volume and quality of pre-chat behavioral data available before the conversation opened — pages visited, tasks attempted, self-service failures, and frustration signals such as rage clicks and repeated actions. Sessions scoring 3 or above were classified as "context available," meaning Conviva had sufficient signal that a context graph injection could have informed the agent prior to the first message. Sessions below 3 were classified as "no context." Six proxy metrics were then computed across sessions to measure distinct failure modes: context re-establishment rate, time wasted per session, mid-context abandonment, and post-context abandonment.