

# AI Agents 101

Bounded Agency Over Autonomous Agents

*Practical guidance for teams navigating AI hype*

Brock Webb

2026-02-10

*The views expressed are the author's own and do not necessarily represent the views of the U.S. Census Bureau or the U.S. Department of Commerce.*

# The Problem

**Everyone is talking about AI agents. Few agree on what the terms mean.**

When someone says “let’s use an AI agent for this” — does everyone in the room picture the same thing?

Usually not.

# The Salesforce Lesson

- ▶ **September 2025:** Cut ~4,000 support staff citing AI product “Agentforce”
- ▶ **December 2025:** Leadership had been “more confident than they should have been”
- ▶ Service gaps, quality issues, reliability problems followed

**They sell the AI product. They still got caught by the gap between expectations and reality.**

# What We'll Cover

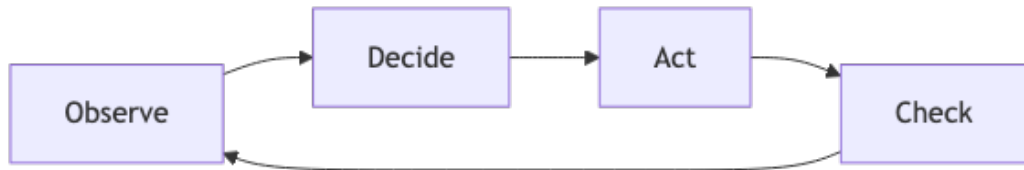
1. **Vocabulary** — shared definitions for the terms that matter
2. **Live Demo** — see the concepts in a simple workflow
3. **Case Study** — bounded agency at scale (briefly)
4. **Design Principles** — what to remember when the details fade

# Core Vocabulary

Term	What It Is	Key Question
<b>Workflow</b>	Structure, sequence of steps	What's the process?
<b>Agent</b>	Entity that does work	What's doing the work?
<b>Agency</b>	Granted decision-making authority	What decisions can it make?
<b>Agentic</b>	Behavior where agency is exercised	How much can it adapt?
<b>Tool</b>	Single discrete operation	What can it do?

# The Loop

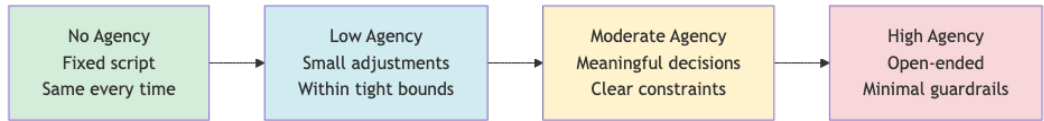
The heartbeat of agent behavior:



**Observe → Decide → Act → Check → (repeat)**

Every framework, every tool — this pattern remains constant.

# Agentic Is a Dial, Not a Switch



**Move right only when you have to. Stay left when you can.**

# The Loop in Practice

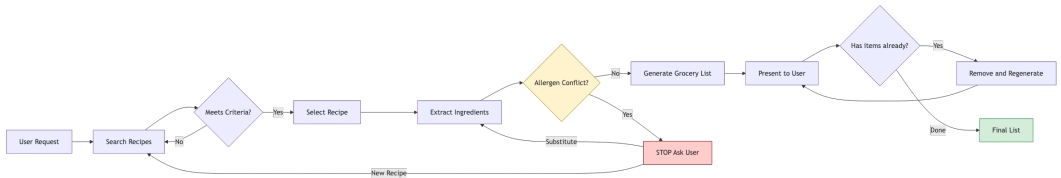
A recipe assistant — simple enough to understand, complex enough to be useful.

Phase	What Happens
<b>OBSERVE</b>	User says “I want to make chicken cacciatore”
<b>DECIDE</b>	Which recipe meets criteria? (4+ stars, reputable source)
<b>ACT</b>	Search → Select → Extract ingredients → Generate list
<b>CHECK</b>	Allergen conflict? User has items? Goal met?

The loop repeats until done — or until it needs to stop and ask.



# Live Demo: Recipe Workflow



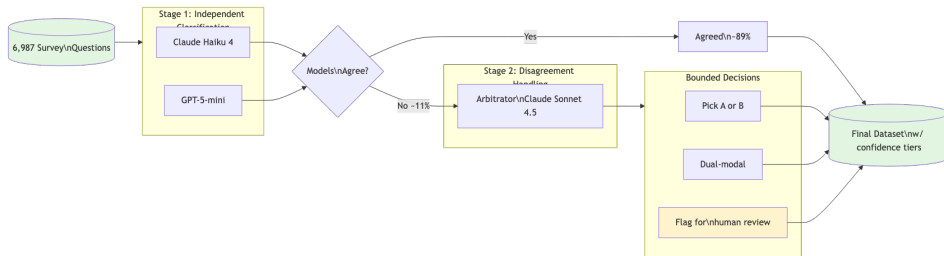
# What Makes It Agentic?

- ▶ **Decision points with criteria** — which recipe? (4+ stars, reputable source)
- ▶ **Conditional branching** — allergen check is a hard stop
- ▶ **Iteration** — “anything you already have?” creates a refinement loop
- ▶ **Transparency** — “explain why you chose it”

This is a **workflow with bounded agency**, not an autonomous agent.

# Case Study: Multi-Survey Concept Mapper

6,987 questions across 46 surveys → mapped to Census topic and subtopic lists



# Why Bounded Agency?

Autonomous Agent Approach	This Workflow
One model decides everything	Cross-validation catches errors
Confidence is implicit	Confidence tiers are explicit
Edge cases get guessed	Edge cases get flagged
Failures are silent	Failures are surfaced

**Result:** 99.5% success rate, ~\$15 total cost\*, complete audit trail.

*\*Reality check: That was the working version. Getting there took ~8 hours of setup, testing, failures, and restarts. Always start with a small sample run. It's a do-loop until done right.*

# Design Principles

#	Principle	One-Liner
1	Good judgment upfront	Design quality bounds output quality
2	Agency requires governance	Less agency is often better
3	Most problems don't need agents	Simple solutions beat complex ones
4	Specification is the skill	Clarity beats capability
5	Design for uncertainty	Plan for failure, not just success
6	Digestible chunks	Focused beats sprawling

# The Stakes: What Ignoring These Causes

Microsoft's AI Red Team documented failure modes in agentic systems.<sup>1</sup>

Nearly every one traces back to violating these principles:

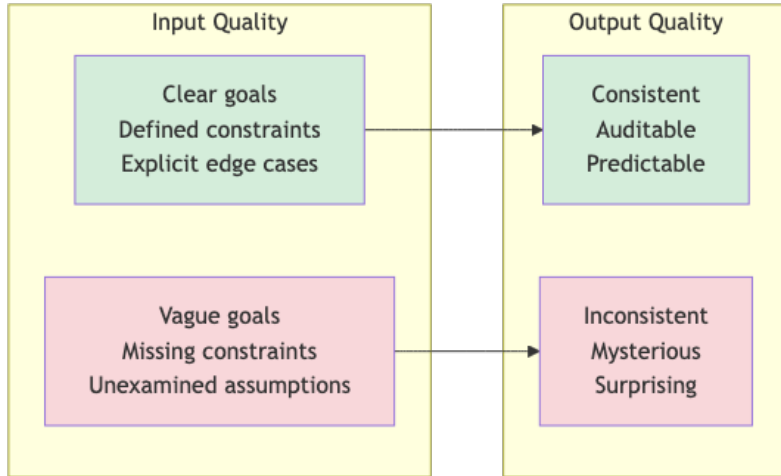
Ignore This...	...And Get This
Good judgment upfront	Misalignment, hallucinations, misinterpretation
Agency requires governance	Actions outside scope, user harm
Simple solutions first	Attack surface, knowledge loss
Clear specifications	Wrong permissions, accountability gaps
Designing for uncertainty	Cascading failures, denial of service

**These principles aren't caution. They're how you build systems that work.**

---

<sup>1</sup>Microsoft AI Red Team, *Taxonomy of Failure Modes in Agentic AI Systems* (2025)

# The Meta-Principle



**AI amplifies your process.** Good process + AI = faster good outcomes.

The test: *Would I sign my name under this action?*

# The Bottom Line

You don't need expensive tools or deep technical skills to work with AI agents effectively.  
You need:

- ▶ Clear thinking
- ▶ Well-structured prompts
- ▶ Appropriate skepticism about what automation can and should do

**Start simple. Add complexity only when justified. Design for uncertainty. Keep humans in the loop where it matters.**

*That's it. The rest is practice.*



**Student Handout:** Vocabulary, templates, copy-paste recipe prompt

**Course Companion:** Full speaker notes and reference material

**Code:** [github.com/brockwebb/federal-survey-concept-mapper](https://github.com/brockwebb/federal-survey-concept-mapper) (*case study source*)

**Further Reading:**

- ▶ Microsoft AI Red Team, “Taxonomy of Failure Modes in Agentic AI Systems” (2025)
- ▶ OpenAI, “A Practical Guide to Building Agents” (2025)
- ▶ Anthropic, “Building Effective Agents” (2024)

# Questions?



*"When you forget to bound the agency"*