INTRODUCTION TO AGENTIC AI

From Single AI to AI Teams

UBUS 670 — Day 7 • Week 3

LEARNING OBJECTIVES

- **Define** what makes an AI agent different from a simple LLM call
- **Explain** agent anatomy using the perception-reasoning-action framework
- **Identify** the three orchestration patterns and when to use each
- **Build** a two-agent sequential workflow extending Beacon's email triage

FROM DAY 6 TO DAY 7

Day 6: You hardened Beacon's email triage system against adversarial attacks.
Day 7: What if you added a second AI to check the first — automatically?

Day 6 (One AI, Hardened)	Day 7 (Two AIs, Collaborating)
Humans red-team the AI	A second AI checks the first AI
Manual adversarial testing	Automated quality verification
One system prompt, hardened	Two agents, each with a specialized role

THE LIMITS OF A SINGLE AI

- **No self-correction:** If it makes a mistake, it doesn't know
- **No verification:** No one checks its work
- **No persistent memory:** Each conversation starts fresh
- **No tool access:** Can't look things up, send emails, or take actions

> **Beacon example:** Your Day 5 email triage system could

THE VISION: AI TEAMS

What if AI could work like a well-organized department?

Single AI	AI Team
One employee doing everything	Specialized roles with clear handoffs
No oversight or review	Built-in quality checks
Fails silently	Catches errors before they reach customers
One point of failure	Redundancy and verification

AGENT VS. LLM CALL

	LLM Call (Stranger)	Agent (Employee)
Identity	Anonymous, no role	Named role with responsibilities
Memory	Forgets after each call	Maintains context across tasks
Tools	Can only generate text	Can use tools, databases, APIs
Collaboration	Works alone	Passes work to other agents
Judgment	Answers once	Can iterate and improve

AGENT ANATOMY

PERCEPTION → REASONING → ACTION

Component	What It Does	Beacon Example
Perception	Receives and understands input	Reads the customer email
Reasoning	Analyzes and decides	Classifies category and urgency
Action	Produces output or takes action	Routes email, drafts response

> Every agent follows this cycle: perceive the world, reason about

AGENT INSTRUCTIONS

SYSTEM PROMPTS, EVOLVED

Day 5: System Prompt	Day 7: Agent Instructions
Role description	Role + collaboration rules
Output format	Output format + handoff format
Behavioral rules	Behavioral rules + tool access
Works alone	Knows about other agents

> Your Day 5 system prompt becomes Agent 1's instructions. We add: "Pass your classification to the Quality Checker agent for verification."

BUSINESS AGENTS IN THE WILD

Agent	Perception	Reasoning	Action
Email Triage	Reads email	Classifies category/urgency	Routes to queue
Expense Processing	Reads receipt	Checks policy compliance	Approves or flags
Compliance Review	Reads document	Checks against regulations	Flags violations
Resume Screening	Reads resume	Matches to job requirements	Ranks candidates

AGENT GOVERNANCE

AGENTS NEED OVERSIGHT, JUST LIKE EMPLOYEES

- **Trust boundaries:** What can each agent do? What's off-limits?
- **Human-in-the-loop:** When should an agent escalate to a human?
- **Monitoring:** How do you track what agents are doing?
- **Audit trail:** Can you reconstruct why an agent made a

CHECKPOINT QUIZ 1: AGENT CONCEPTS

What is the most accurate distinction between an AI agent and a simple LLM call?

A) An agent uses a newer, more powerful model

B) An agent has a role, memory, tools, and can collaborate with other agents

ORCHESTRATION PATTERNS

How Agents Work Together

SEQUENTIAL PATTERN

THE ASSEMBLY LINE

Agent A → Agent B → Agent C

> **Beacon example:**
> Email → **Triage Agent** (classifies) → **Quality Checker** (verifies) → Final output
>
> Each agent's output becomes the next agent's input. Like an assembly line, work flows in one direction.

Best for: Tasks with natural dependencies where step B

PARALLEL PATTERN

MULTIPLE ANALYSTS WORKING SIMULTANEOUSLY

Multiple agents process different inputs at the same time, then results are merged.

> **Beacon example:**
> 100 emails arrive → 5 Triage Agents work simultaneously → Results merged
>
> **Trade-off:** Faster processing, but agents don't share context.

LOOP PATTERN

ITERATIVE REFINEMENT

Agent produces output → Reviewer checks → If not good enough, try again.

> **Beacon example:**
> Triage Agent classifies email → Quality Checker reviews → If DISAGREE, Triage Agent reclassifies → Repeat until AGREE or max attempts reached.
>
> **Trade-off:** Higher quality output, but costs more (each loop = more API calls).

WHICH PATTERN SHOULD YOU USE?

Pattern	Use When	Tradeoff	Example
Sequential	Tasks have dependencies	Slower but reliable	Triage → Quality Check
Parallel	Tasks are independent	Fast but no shared context	Batch email processing
Loop	Quality > speed	Better output but costly	Draft → Review → Revise

> **Today's lab:** You'll build a **sequential** pipeline. The dependency is clear: the Quality Checker needs the Triage

CHECKPOINT QUIZ 2: PATTERN SELECTION

Beacon needs to process 500 emails overnight.

Each email can be classified independently.

Which pattern is most appropriate?

A) Sequential — process each email one after another

B) Parallel — multiple agents process emails simultaneously

TODAY'S LAB PREVIEW

BUILDING YOUR FIRST MULTI-AGENT SYSTEM

Part	What You'll Build	Time
Part 0	Setup — tool account and interface tour	15 min
Part 1	Agent 1 — Triage Agent (your Day 5 system prompt)	25 min
Part 2	Agent 2 — Quality Checker (new verification agent)	25 min
Part 3	Connect — sequential pipeline, end-to-end testing	20 min
Part 4	Document — architecture doc and CTO pitch	15 min

KEY TAKEAWAYS

1. **Agents > LLM calls** for complex business tasks — roles, memory, tools, collaboration
2. **Perception → Reasoning → Action:** every agent follows this cycle
3. **Three patterns:** Sequential (dependencies), Parallel (independent), Loop (quality)
4. **Start with verification:** The most practical first step in multi-agent AI

COMING NEXT: DAYS 8-9

RESUME SCREENING CAPSTONE

From email triage to talent acquisition — 3+ agents working together:

Agent	Role
Parser Agent	Extracts structured data from resumes
Evaluator Agent	Scores candidates against job requirements
Ranker Agent	Ranks candidates and generates shortlist

> Today you build a 2-agent pipeline. On Days 8-9, you'll build a 3+ agent system that screens real resumes. Same patterns,

QUESTIONS?

Before we head into the lab...