EdSutra

Teaching guide

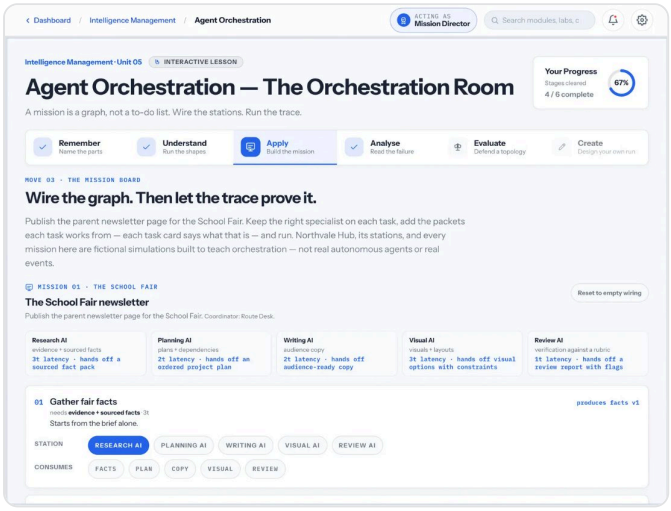
Module 09 · Intelligence Management · Unit 04 · Stage 3, Responsibility

Agent Orchestration

The Orchestration Room

Grades 7–12 | 30 minutes | Role: **Mission Director**

Grade 7: About age 12–13 · Year 8 in the UK · Class 7 in India



The big idea

A team of AI agents is a graph of boxes and arrows, not a to-do list: who does what, in what order, with which version of the shared context. Running agents side by side can be faster but adds handoffs that can break. Only the event log, not a busy animation, proves every handoff passed its check.

The learner's job

Learners are the **Mission Director** at Northvale Hub · Mission Control. Specialist stations wait for work, and work waits for packets. Your job is the graph between them: who does what, in what order, with which context — and what happens when a handoff fails. Their task: wire the stations, move the packets, and prove every handoff in the event log.

By the end, learners can say

- I can identify specialist, task, dependency, handoff, shared context, coordinator, and verification gate
- I can predict which sequential, parallel, or conditional workflow shape completes a mission reliably
- I can decompose an authored mission, assign specialists, define context packets, and execute the deterministic workflow
- I can inspect failed traces and diagnose capability mismatch, missing context, stale version, dependency race, or skipped verification
- I can compare orchestration plans for correctness, latency, cost/rework, observability, and human escalation
- I can design and successfully run an orchestration plan for a fresh mission with explicit interfaces and recovery paths

Before class

- One computer or tablet for each learner or pair, with EdSutra open (this Module is part of the full EdSutra studio).
- Open the studio once on each device before class, so the Unit still runs if the internet drops.
- Play the first two steps yourself, so you know where learners will click.
- Draw the School Fair mission on the board as five boxes: facts, plan, copy, header, review. Add the arrows together during Apply.
- Best after What Is Intelligence Management? and Delegation & Verification.
- Units open in order, and Module 1 comes first. To start a class here, switch on Open every Unit in Settings → Teachers & parents.

Words to introduce

Dependency: A task that must finish before another can start.

Shared context: Versioned packets that more than one task reads.

Coordinator: Routes work and packets without doing the work itself.

Verification gate: A check a packet must pass before it ships downstream.

Fan-out: Splitting work so two stations run side by side, then merge.

Tick: One step of simulated time; the event log counts them.

Event log: The record of what each station did and when; the source of truth.

Running the lesson 35 minutes

Open 4 min

"Five AI stations are building one newsletter page. Is it faster to let them all work at once? What could go wrong if they do?"

Before anyone clicks, ask: a line of stations one after another, or Writing and Visual side by side. Which finishes the newsletter page sooner, and by how much? Write the guesses on the board and come back to them in Understand.

Step	What learners do	Ask the class	Listen for
1. Remember 4 min	Learners read seven short examples, such as "Writing cannot start until the fact pack exists", and match each to its part: specialist, task, dependency, handoff, shared context, coordinator or verification gate.	• What is the difference between a coordinator and a specialist? • Why does shared context need a version number?	A coordinator routes work but does none of it. Versions tell a task whether its packet is current.
2. Understand 5 min	Three pairs of plans: a line or a fan-out, one writer or two, ship fast or verify. Learners predict which works, then run both traces and read the ticks in the event log.	• The fan-out took 9 ticks and the line 12. What did the fan-out cost? • Two stations wrote the same copy packet. What went wrong at Review? • Look back at our guesses on the board. Were we right?	Parallel work is faster but adds a merge to watch. One owner per packet, and a check costs a tick.
3. Apply 6 min	On the School Fair mission board, learners keep the right station on each of five tasks, add the packets each task reads, and run until every handoff passes.	• Which task needs two packets, and why? • Why can the header start before the copy is written?	Each task reads exactly what it needs. Copy needs the facts and the plan; the header needs only the plan.
4. Analyze 5 min	Five missions failed. Learners read each trace and name its one cause: capability mismatch, missing context, stale version, dependency race or skipped verification. Two right diagnoses open the next step.	• In the v1 fact pack case, where did the trace fail: the producer or the consumer? • Research AI was given the writing task. Why did it fail before it started?	Each failure lives in the wiring: the wrong station, a missing or old packet, two owners, or no check.
5. Evaluate 5 min	Two decisions: line or fan-out for the School Fair, and ship fast or add a review gate. Learners pick a verdict and back it with two evidence chips that fit.	• Can both the line and the fan-out be the right choice? When? • Why is shipping fast ruled out for a page every family receives?	No plan shape is simply correct. A choice counts when its evidence fits, such as a tight deadline or an untested merge.
6. Create 6 min	Learners design the Science Week poster run: assign a station to each of six tasks, say which packets each task reads, and make Review read the layout as the recovery path.	• What does the layout task need before it can start? • Why must Review read the layout, not just run at the end?	A plan that runs and can be checked: every task names its inputs, and Review checks the layout before it ships.

Steps follow Bloom's revised taxonomy, in the order the Unit presents them.

Deepen the thinking

Critical thinking	An animation showed packets flying between stations. Why does the Unit trust the event log instead? What makes a record trustworthy?
AI thinking	Some AI systems split a job among many agents. When would adding one more agent slow the job down instead of speeding it up?
Ethics and fairness	If a newsletter with old numbers reaches every family, who is accountable: the coordinator, the station that used v1, or the person who designed the plan?
Connect to life	Have you ever worked from an old version of a shared document or timetable? What went wrong further down the line?

Misconceptions to listen for

"More agents are always better."
In Understand, the fan-out saves 3 ticks, but its two handoffs must merge at Review, and a merge is where races happen.

"If the packets moved, the mission worked."
The animation is not proof. Only handoffs whose checks passed count, and the event log is the record to read.

"Context is just background text."
Shared context has versions. In Analyze, Writing needs facts v2 but only v1 exists, so the trace fails at the consumer.

In real life the card learners see at the end of the Unit

- A restaurant kitchen is orchestration: cooks work in parallel, and no plate leaves until it's checked.
- Some AI systems split a job among several agents. More agents can be faster, or just add handoffs to break.
- Working from an old version of a shared document is a stale-context failure, and it spreads downstream.

Try it this week. Draw a family task, like making breakfast, as boxes and arrows. Which steps can run at once, and which must wait?

Watch out for. More agents aren't always better. A handoff only counts once its check passes, and stale context quietly spoils every later step.

Talk about it. In a group project, what goes wrong when nobody coordinates? Would an AI coordinator help?

Exit ticket

1. What is the difference between a dependency and a handoff?

2. Why did the fan-out finish sooner than the line, and what did it risk?

3. Name two of the five ways a mission failed in Analyze.

Going further

For homework, learners draw a family task, like making breakfast, as boxes and arrows, mark which steps can run at once, and add one check before it is served. The In real life card's "Try it this week" has the same idea.