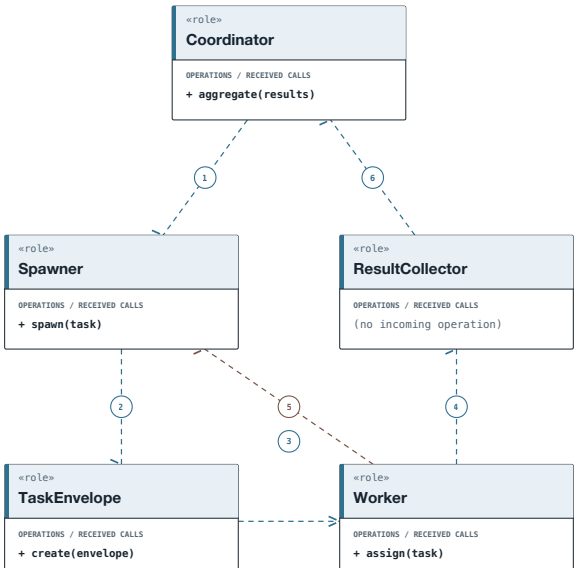
P12 Coordinator-worker spawn flow

Primitive / UML CLASS VIEW / ANALYTICAL MODEL

Delegate bounded subtasks.

UML CLASS VIEW

dashed arrow: call dependency · brown: return/failure · analytical, not an observed trace



POINTCUT / PATTERN INTERFACE

CONTEXT	A coordinator delegates a bounded task to another execution unit.
TRIGGER	A worker is spawned, called as a tool, or handed a task.
STRUCTURE	Task envelope with identity, instructions, authority, budget, and result target.
EFFECT	Worker output returns to the coordinator with explicit lifecycle and provenance.

CONTRACT / REQUIRED BEHAVIOR

REQUIRES	Parent and child identities, task scope, and shared data policy are known.
ENSURES	Each worker has one task record, bounded authority, and a correlated result.
INVARIANT	Delegation never silently expands permissions or budget.
VIOLATION	Cancellation, worker failure, timeout, and duplicate spawn are typed outcomes.

RELATIONSHIPS / ANALYTICAL CALLS

number · caller → callee · message

1 Coordinator → Spawner spawn(task)
2 Spawner → TaskEnvelope create(envelope)
3 TaskEnvelope → Worker assign(task)
4 Worker → ResultCollector return(result)
5 Worker ↩ Spawner report(failure)
6 ResultCollector → Coordinator aggregate(results)

VARIATION / TEST / NEGATIVE CONTROL

VARIATION	Sub-agent, agent-as-tool, background agent, child session, and worker pool.
TEST_OBLIGATION	Test authority, timeout, cancellation, duplicate results, and context isolation.
NEGATIVE_CONTROL	Sequential calls to independent agents without delegation do not count.