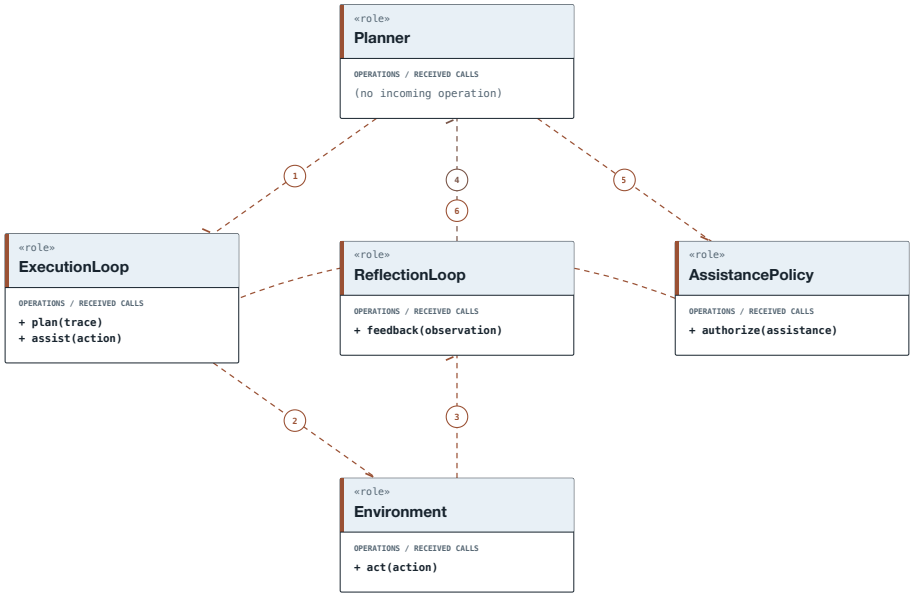
E09 Dual-loop assistance

Evolution / UML CLASS VIEW / ANALYTICAL MODEL

Separate planning and execution feedback.

UML CLASS VIEW

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



POINTCUT / PATTERN INTERFACE

CONTEXT	A planner or assistance loop and an executor loop operate concurrently or iteratively.
TRIGGER	Executor state or environment feedback updates planning.
STRUCTURE	Planner, executor, feedback channel, synchronization, and authorization boundary.
EFFECT	The next plan or assistance action changes in response to attributable executor feedback.

CONTRACT / REQUIRED BEHAVIOR

REQUIRES	Planner and executor states have compatible identities and timing.
ENSURES	Feedback reaches planning before the affected action when required.
INVARIANT	The outer loop cannot bypass executor authorization for physical or external actions.
VIOLATION	Stale, conflicting, or unavailable feedback has a safe coordination policy.

RELATIONSHIPS / ANALYTICAL CALLS

number · caller → callee · message

1 Planner → ExecutionLoop plan(trace)
2 ExecutionLoop → Environment act(action)
3 Environment → ReflectionLoop feedback(observation)
4 ReflectionLoop ↩ Planner reflect(summary)
5 Planner → AssistancePolicy authorize(assistance)
6 AssistancePolicy → ExecutionLoop assist(action)

VARIATION / TEST / NEGATIVE CONTROL

VARIATION	Reactor-assisted executor, dual loop, planner-executor, and assistance policy.
TEST_OBLIGATION	Test feedback timing, stale feedback, authorization, and loop termination.
NEGATIVE_CONTROL	Two unrelated loops without a feedback link do not meet the pointcut.