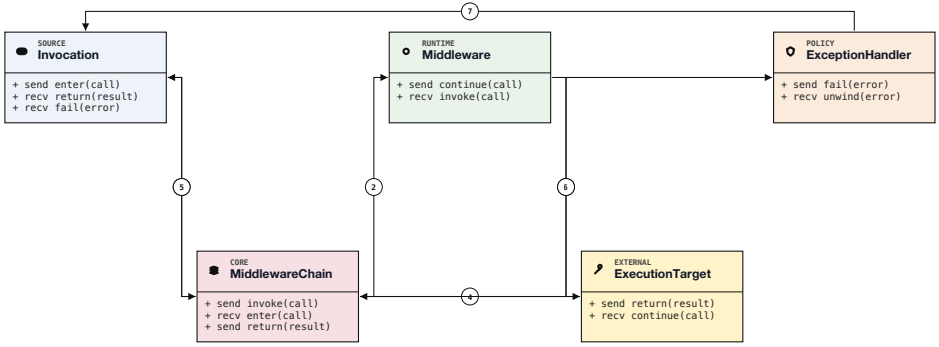
P04 Middleware-pipeline loop

Primitive · A² A1 · layered role collaboration

Layer behavior around model/tool calls.

STRUCTURE · PARTICIPANT COLLABORATION



PATTERN INTERFACE

context
A harness surrounds model or tool invocation with extensible behavior.

trigger
A call enters or leaves the runtime boundary.

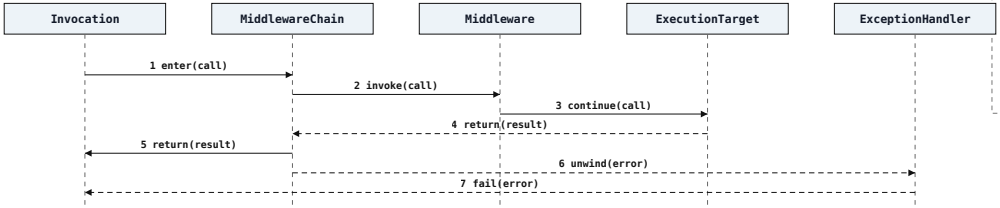
structure
Ordered middleware entries with a continuation or short-circuit result.

effect
Logging, policy, retry, or context behavior runs in declared order.

CONTRACT ANNOTATIONS

requires	The boundary exposes a composable continuation contract.
ensures	Middleware executes in deterministic order and the final result reaches the loop.
invariant	A middleware does not silently bypass required downstream layers.
violation	Short circuits and exceptions unwind deterministically with one terminal result.

INTERACTION



MESSAGE TRACE `1 enter(call) → 2 invoke(call) → 3 continue(call) → 4 return(result) → 5 return(result) → 6 unwind(error) → 7 fail(error)`

Evidence: 38/41 repositories, 34/36 families; 38 trajectories (19 multi-checkpoint); exact audit: not sampled.

Variation: Registration order, retry, redaction, tracing, and policy middleware.

Negative control: A hard-coded helper call is not a middleware pipeline.