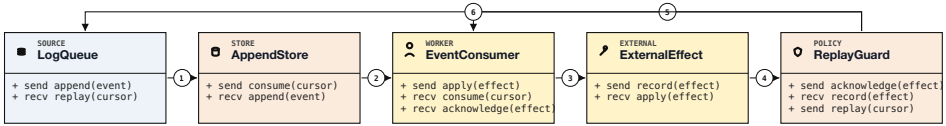
P13 Log-as-queue loops

Primitive · A² A1/A3 · stream role collaboration

Drive work from a durable execution record.

STRUCTURE · PARTICIPANT COLLABORATION



PATTERN INTERFACE

context
New work is represented by durable append-only records.

trigger
A consumer polls or subscribes to unseen records.

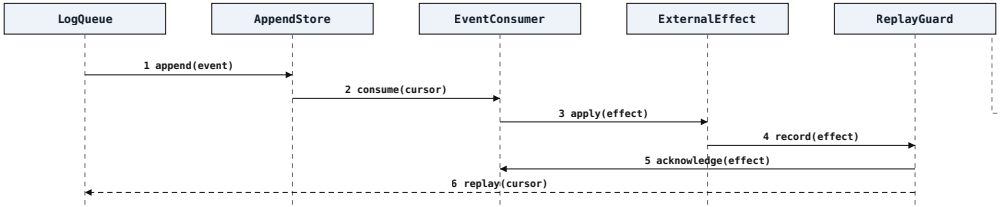
structure
Event identity, append order, consumer cursor, and acknowledgement.

effect
Work advances from the log without replaying already completed effects.

CONTRACT ANNOTATIONS

requires	Records have stable identity and consumers retain progress.
ensures	Each effectful unit has an idempotency or acknowledgement rule.
invariant	Consumer crash does not erase unacknowledged work.
violation	Duplicate delivery is deduplicated or documented as at-least-once.

INTERACTION



MESSAGE TRACE `1 append(event) → 2 consume(cursor) → 3 apply(effect) → 4 record(effect) → 5 acknowledge(effect) → 6 replay(cursor)`

Evidence: 24/41 repositories, 22/36 families; 24 trajectories (11 multi-checkpoint); exact audit: not sampled.

Variation: SQL log, file log, queue adapter, polling, and projection.

Negative control: An in-memory queue is not log-as-queue.