

DOCUMENTATION PAGE

Queue Input Feasibility Receipt

Queue Input Feasibility Receipt

Purpose

This is a read-only, candidate-only Internal Knowledge handoff for workflow maintainers and workflow operators. It checks whether the fixed inputs for the already selected `internal` queue form a usable product contract. The upstream selection remains the Workstream Queue Selection Receipt: `internal` is selected, while `onboarding` and `customer` remain visible comparators. Feasibility is downstream of that selection and cannot replace it, schedule work, mutate queue state, or turn a product conclusion into a platform attestation. The receipt is public-safe because its page claims are empty and its material source references are the three sealed, public-safe logical IDs.

Fixed Representative Fixtures

`registries/queue-input-feasibility-fixtures.json` is the sole fixed fixture inventory for this receipt. It contains the complete three-row upstream set (`internal` , `onboarding` , and `customer`), one selected-queue case with ready sealed authority and a separately recorded freshness gap, and explicit `missing` , `stale` , `conflicting` , and `non-public-safe` cases. The latter are test-shaped blocked inputs, not observations about a live queue and not permission to repair authority. Each row retains age, age rank, eligibility, underserved status, scheduling tier, audience, intended outcome, selection reason, source references, freshness fields, and balance-window membership. The fixture file also records the upstream page and candidate identities, required artifacts, standard acceptance IDs, and command allowlist so another reader can derive the same result without inventing a queue source identity or revision.

Eligibility Checks

An input is eligible for a product feasibility conclusion only when it has a valid contract, uses the standard acceptance IDs AC-1 through AC-5, has complete non-exploratory journey coverage, and names only non-empty commands from the fixture allowlist. The required artifacts are the sealed authority, registered sources, upstream selection page, upstream candidate record, this page and its registry entry, the fixture inventory, the journey manifest, and the deterministic projection. The sealed authority is ready here at revision `2026-08-08` with exactly the three registered public-safe sources and `employee_may_modify: false`. The fixture has a complete balance window for all three queue IDs, but its independent queue-source identity and revision are unestablished; therefore queue-input freshness is explicitly `blocked`, not guessed from queue age. Missing, stale, conflicting, or non-public-safe inputs fail closed with an owner-visible finding.

Derivation Record

The deterministic derivation first checks the fixture schema and exact acceptance-ID set, then matches the upstream receipt identity and all three queue rows, then checks required artifacts, journey traces, command strings, sealed authority, source disclosure, freshness, and balance-window evidence. It next confirms that `internal` is still the upstream selected ID and that feasibility has not been used as a tie breaker. The authority readiness result is `ready`; the balance-window result is `ready`; the queue-source freshness result is `blocked` because no independent queue-source identity or revision is established by the candidate; and the product feasibility conclusion is `blocked-pending-freshness-evidence`. The derivation is deterministic and fixture-backed, while the unresolved platform reader follow-up remains platform-owned evidence rather than a product defect or attestation.

Read-Only Boundary

This receipt describes product evidence only. It does not schedule work, mutate queue state, alter the selected workstream, create or change a release pointer, promote, publish, roll back, activate, or write a live directory. It does not claim that a reader journey, independent review, route, identity, validator authority, evidence chain, approval, or platform gate has executed. The preserved validator record is authoritative evidence supplied for the handoff; it is not replaced or rerun here. A missing, stale, conflicting, or non-public-safe input remains visibly `blocked` with its reason and owner. In particular,

the environment-specific reader result is not converted into a recurring product harness defect, and no platform-owned fact is inferred from the product fixture or deterministic build.

Acceptance Checks

- **AC-1 — Authority:** Confirm the exact sealed authority revision, three logical source IDs, immutable employee flag, and public-safe source records; missing, stale, conflicting, or non-public-safe authority is blocked.
- **AC-2 — Contract and scope:** Confirm a valid candidate contract, complete required artifacts, the upstream receipt identity, and all three fixed queue inputs, with `internal` remaining selected before feasibility is considered.
- **AC-3 — Journey coverage:** Confirm standard IDs AC-1 through AC-5, complete non-exploratory `traces_to` coverage, natural-language reader journeys, and non-empty commands that occur in the JSON allowlist.
- **AC-4 — Evidence and derivation:** Confirm ready sealed authority, explicit freshness evidence, complete balance-window evidence, and deterministic fixture-only derivation; unresolved queue-source freshness stays blocked.
- **AC-5 — Boundary:** Confirm the result is a read-only product conclusion with open findings preserved and no schedule, mutation, platform attestation, approval, promotion, publication, rollback, pointer, or live directory effect.

Provenance: registry-listed (no candidate packet)

Registry Source: `content/queue-input-feasibility-receipt.md`