

# Workflow Discovery Protocol

The Workflow Discovery Protocol is the first execution-cluster artifact for any agent workflow build. It provides a structured playbook for surfacing the real workflow — including undocumented decisions, informal handoffs, shadow exceptions, and the gap between written procedures and actual operator behavior — before any automation is designed. Without this grounding step, an agent workflow encodes the documented fiction rather than the practiced reality, guaranteeing silent failure modes the moment edge conditions arise. The protocol feeds e4-task-decomposition-framework and e4-workflow-artifacts with source-of-truth step inventory and a divergence log that becomes the seed of the exception register.

## When to Run This Protocol

Run before any workflow model, agent task spec, or process map is drafted. Revisit after any significant regulatory change, staff turnover, or observed deviation between agent outputs and human expert judgment. The canvas from s1-operating-context-canvas must already be complete; use it to scope the interview cohort and bound what is in scope vs. out of scope.

## Phase 1 — Preparation

Before any interview session, review the s1-operating-context-canvas for:

- agent\_objective — the single decision the workflow must produce
- hard\_constraints — rules no step can violate
- out\_of\_scope — adjacent tasks the agent must refuse; probe interviewees specifically about these edges
- irreversible\_impact\_boundary — steps at or near this boundary deserve extended interview time

Collect all available documented materials: SOPs, checklist templates, historical intake logs, prior audit findings, org charts, and any previous automation specs. These are the documented baseline; the protocol's job is to find where practice diverges.

Identify the minimum required cohort:

| Role                       | Why Needed                                       |
|----------------------------|--------------------------------------------------|
| Front-line practitioner    | Knows what actually happens at each step         |
| Supervisor / reviewer      | Knows exception paths and override patterns      |
| Back-office / filing staff | Knows the handoff and output format requirements |
| Occasional substitutes     | Expose undocumented tribal knowledge             |

Three to five interviews per workflow typically saturates signal. Proceed to modeling when two consecutive interviews produce no new decision points or exception patterns.

## Phase 2 — Interview Script

Use this script verbatim for the first pass. Deviate only to probe a specific answer. Record sessions (with consent) or take verbatim notes — paraphrase introduces bias.

### Opening (5 min)

"Walk me from the very start of this process — the moment it begins — to the moment you consider it done. Don't skip anything; I want to see it as you do it, not as the SOP says to do it."

Listen without interrupting. Note: every step named, every tool or system touched, every person consulted, every wait state.

### Step-level probes (15–25 min, one round per named step)

For each step the interviewee named:

1. "What do you need before you can start this step — what's the exact input and where does it come from?"
2. "What do you produce at the end of this step — what exactly goes where?"
3. "What are you deciding here, even if it feels automatic?"
4. "When does this step go differently than normal — what makes you stop or change course?"
5. "Who else is involved, and what do they need from you at this point?"
6. "If you were on vacation and someone unfamiliar had to do this step, what would go wrong first?"

### Exception probes (10 min)

- "What's the hardest call you ever had to make in this process?"
- "What sends something back to the applicant / requester?"
- "What escalates to your supervisor or outside expert?"
- "What is the one situation where you would never delegate this to a system or junior colleague?"
- "What did you do last week that wasn't in any procedure?"

### Documentation divergence probe (5 min)

Show the interviewee the written SOP (if one exists):

- "Where does this not match what you just described?"
- "Which parts do you actually follow, and which do you adapt?"
- "Is there anything in the SOP you think is wrong or outdated?"

Wrap (5 min)

- "Who else should I talk to who sees this differently?"
- "Is there a version of this process you handle that nobody else handles?"

## Phase 3 — Divergence Log

After each interview, populate one row per divergence discovered. This log is the primary input to the exception register section of e4-workflow-artifacts.

```
divergence_log:
- divergence_id: <kebab-case, e.g., permit-triage-dv-001>
  step_reference: <step name or tentative id>
  documented_behavior: <what the SOP / written procedure says>
  actual_behavior: <what the interviewee described actually happens>
  frequency: <always | often | sometimes | rarely>
  risk_class: <low | medium | high | critical>
  interviewee_role: <role, not name>
  follow_up_needed: <yes | no>
  notes: <free text>
```

Mark risk\_class: critical whenever actual behavior crosses or approaches the irreversible\_impact\_boundary from s1-operating-context-canvas. These rows become mandatory HITL gate candidates in e4-decision-logic-builder.

## Phase 4 — Step Inventory Draft

Synthesize the interview findings into a flat ordered list. Do not impose a hierarchy yet — that is the work of e4-task-decomposition-framework. At this stage, capture:

```
step_inventory:
- step_draft_id: <kebab-case>
  description: <one sentence>
  inputs: [<source or artifact>]
  outputs: [<destination or artifact>]
  decision_present: <yes | no>
  exception_noted: <yes | no>
  handoff_present: <yes | no>
  primary_owner_role: <role>
  data_sources_touched: [<system / connector>]
  divergences: [<divergence_id refs>]
```

## Phase 5 — Saturation Check

Do not close discovery until:

- [ ] At least two consecutive interviews produced zero new steps, decision points, or divergences
- [ ] Every out\_of\_scope item from s1-operating-context-canvas was specifically probed and a boundary answer recorded

- [ ] Every step that touches irreversible\_impact\_boundary items has at least two interviewee accounts
- [ ] At least one substitute or backup practitioner was interviewed
- [ ] All known exception types have at least one concrete example with inputs, trigger, and resolution

## Worked Example: South Walton Stormwater Permit Triage

The following illustrates a completed discovery session for the stormwater permit triage agent whose canvas appears in s1-operating-context-canvas. All figures are illustrative.

Cohort: District intake clerk (primary), district engineer (reviewer), admin assistant (filing).

Selected divergences surfaced:

```
divergence_log:
- divergence_id: permit-triage-dv-001
  step_reference: wetlands-check
  documented_behavior: "Clerk checks parcel against printed wetlands map quarterly-
updated."
  actual_behavior: >
    Clerk verbally confirms with engineer if parcel is within 300 ft of
    a known wetlands boundary, because the printed map is often 6+ months stale.
  frequency: often
  risk_class: high
  interviewee_role: district-intake-clerk
  follow_up_needed: yes
  notes: >
    FWC data refresh cadence is unknown – flagged as a known_unknown in
    s1-operating-context-canvas. Engineer consultation is an undocumented
    informal HITL gate. Must be formalized.

- divergence_id: permit-triage-dv-002
  step_reference: return-to-applicant-notification
  documented_behavior: "Return notification sent by clerk via email from district address.
"
  actual_behavior: >
    Clerk sometimes calls applicant first if the issue is simple, then sends
    the email as confirmation. No call is logged.
  frequency: sometimes
  risk_class: medium
  interviewee_role: district-intake-clerk
  follow_up_needed: yes
  notes: >
    Unlogged verbal contacts are a compliance gap under FS Chapter 119 public
    records rules [VERIFY]. Agent workflow must not replicate this; all
    contacts must be logged.
```

Step inventory excerpt (pre-decomposition):

```
step_inventory:
- step_draft_id: intake-receipt
  description: Receive permit application from district portal and log receipt timestamp.
  inputs: [district-portal-submissions]
  outputs: [intake-log-entry]
  decision_present: no
  exception_noted: no
  handoff_present: no
  primary_owner_role: district-intake-clerk
  data_sources_touched: [district-portal, intake-log]
  divergences: []
```

- step\_draft\_id: wetlands-proximity-check  
description: Determine whether parcel is within regulated wetlands proximity threshold.  
inputs: [parcel-gis-layer, permit-application-v3]  
outputs: [wetlands-proximity-flag]  
decision\_present: yes  
exception\_noted: yes  
handoff\_present: yes  
primary\_owner\_role: district-intake-clerk  
data\_sources\_touched: [parcel-gis-layer, fwc-wetlands-overlay]  
divergences: [permit-triage-dv-001]
- step\_draft\_id: routing-decision  
description: Assign application to clerk-queue, engineer-review, or return-to-applicant.  
inputs: [wetlands-proximity-flag, completeness-check-result]  
outputs: [routing-recommendation]  
decision\_present: yes  
exception\_noted: yes  
handoff\_present: yes  
primary\_owner\_role: district-intake-clerk  
data\_sources\_touched: [intake-log, engineer-review-queue]  
divergences: [permit-triage-dv-001, permit-triage-dv-002]

## Usage Notes

Pass the completed step inventory and divergence log to e4-task-decomposition-framework for classification. Pass divergences with risk\_class: high or critical directly to e4-decision-logic-builder as mandatory escalation candidate inputs. Any divergence touching irreversible\_impact\_boundary items in s1-operating-context-canvas must be explicitly resolved before the workflow model is finalized — do not absorb it silently into a "normal" step.

Version the discovery output; do not edit the signed-off log in place. New discovery sessions against an existing workflow produce a delta log, not a replacement.