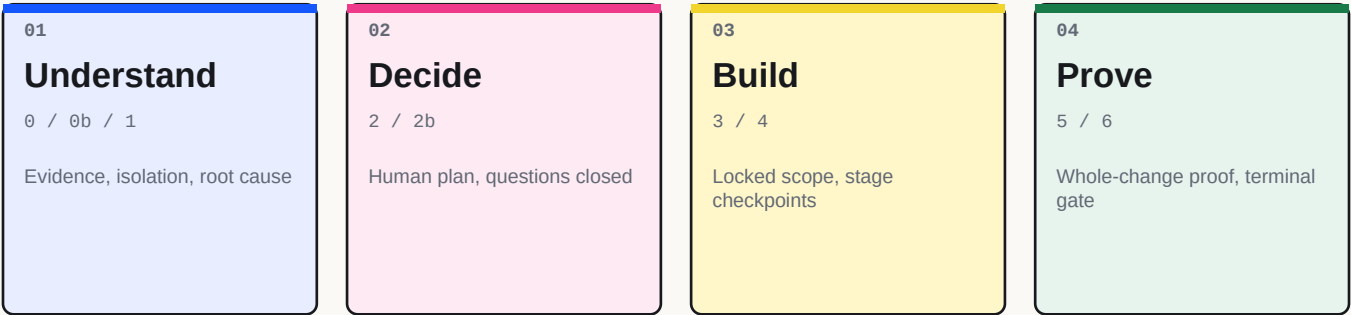


ONE TICKET IN / ONE VERIFIED DRAFT PR OUT

Factory Run Flow

One phase per fresh session. Each phase is gated before the next begins. Compact state and run artifacts carry decisions across sessions.



OPERATING CONTRACT

- 01

Fresh context

The phase starts in a new session. Chat history is not treated as durable state.
- 02

Persistent evidence

Compact state and run artifacts cross the session boundary.
- 03

Executable gates

A phase closes only when gate.py permits it. A failed predecessor blocks the next phase.

LEGEND

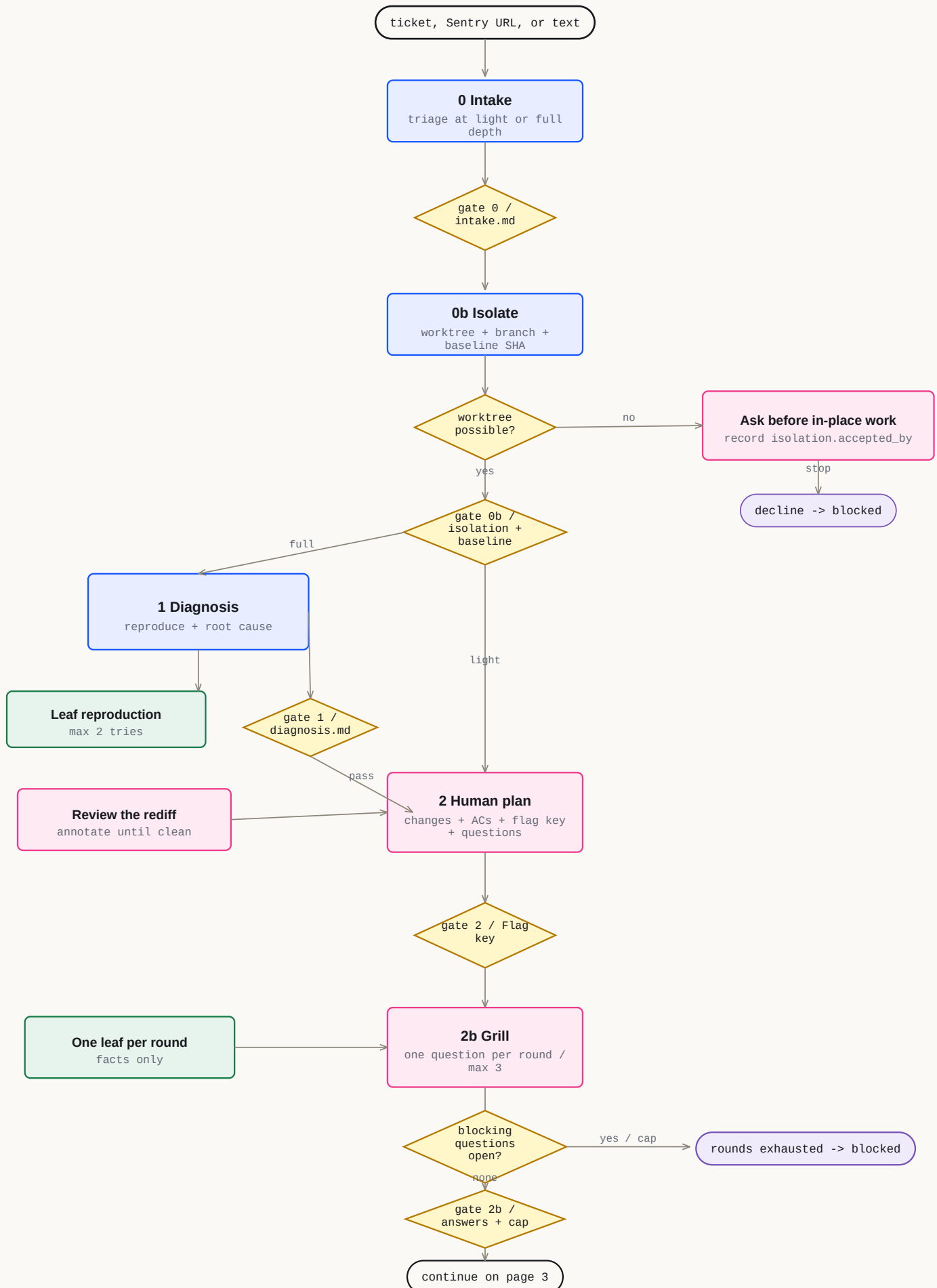
■ Autonomous phase■ Needs you■ Leaf agents■ Executable gate■ Terminal state

RUN ONE PHASE

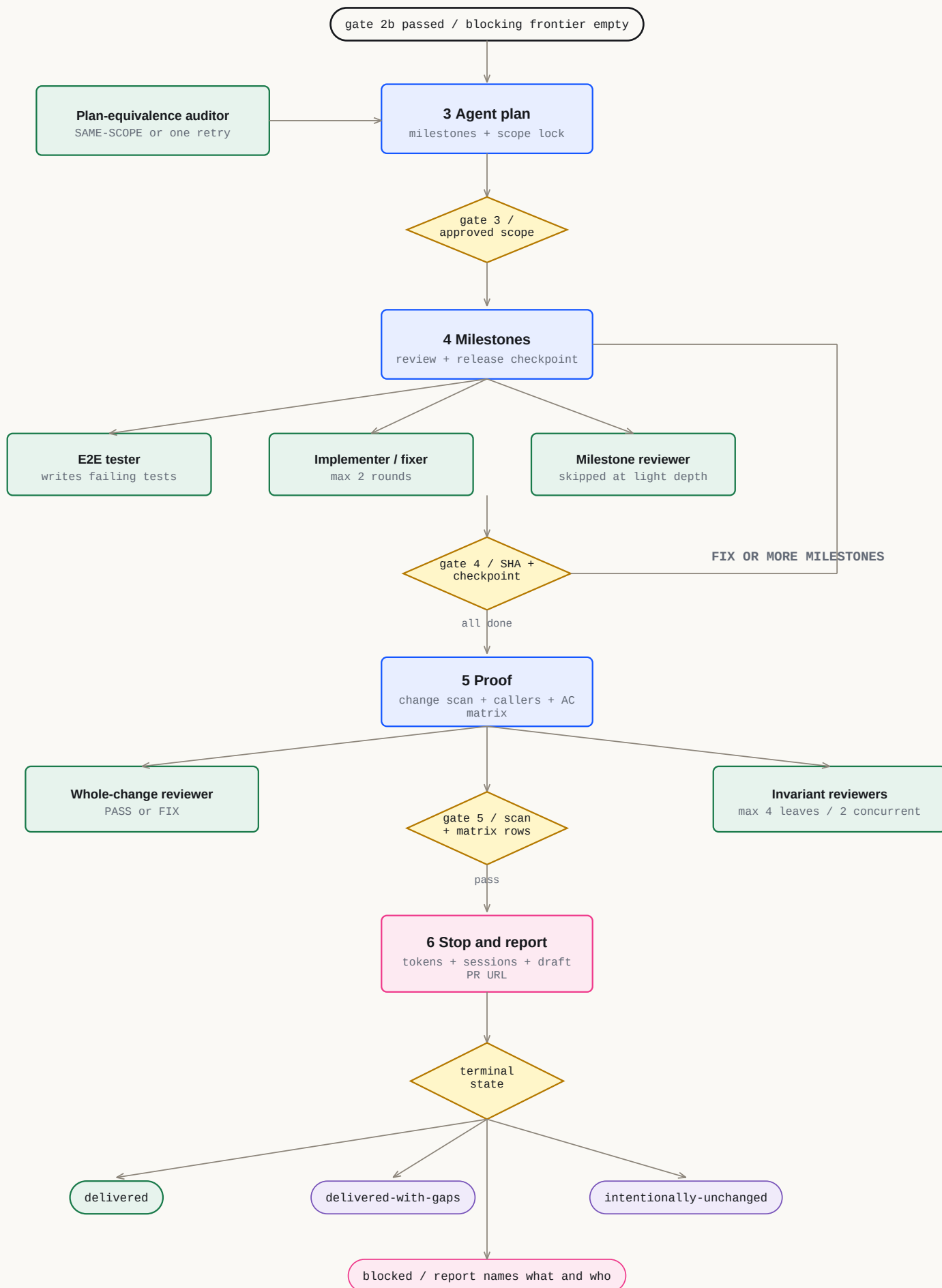
scripts/run.sh <slug>

Pages 2-3: flow / Page 4: roles and stop conditions

PART 1 / UNDERSTAND AND DECIDE



PART 2 / BUILD AND PROVE



WHO DOES WHAT, PER PHASE

Phase	Runs as	Speaks	Closes when
0 intake	opus	triage verdict	intake.md written
0b isolate	sonnet	where the run writes	worktree + baseline SHA, or consent to in-place
1 diagnosis	opus	root-cause synthesis	diagnosis.md; skipped at light depth
2 human plan	you	plan + rediff	plan has a Flag key line
2b grill	you + leaf	one summary per round	no blocking question unanswered
3 agent plan	opus	gate decision	SAME-SCOPE and plan lock
4 milestones	opus + leaves	per-milestone verdict	commit SHA and checkpoint receipt
5 proof	opus + leaves	Phase 5 verdict	scan + AC matrix + reviewer PASS
6 stop	you	final report + PR URL	terminal state set, gate quoted

WHAT STOPS THE LOOP

Signal	run.sh does
Next phase is 2, 2b, or 6	Prints NEEDS-YOU, exit 0. Run /factory resume <slug>, then rerun.
Gate on the closed phase says BLOCKED	Prints reasons, exit 1. Nothing starts on an unpassed predecessor.
state.terminal set	Exit 0. The run is over.
sessions_used hits sessions_max	Exit 0 with the cap named.
Two sessions change neither phase nor milestone	Reruns the gate for the reasons, exit 1.
leaves_used > leaves_max	Gate blocks. Only blocked may still close.

Default caps: sessions_max 16 / leaves_max 40

Override autonomous phase models with FACTORY_MODEL_<PHASE>