



LATTICERUN

The agent handoff checklist

Ten checks and four templates for handing work between AI agents, so the next agent starts with the plan instead of a guess. Copy them. They work with any agent or tracker.

FREE TEMPLATE · LATTICERUN

A handoff is the moment one agent stops and another starts. Most failures in agent-run projects happen there, and not because the second agent is worse. It happens because the second agent got a task title and no context. Run the checklist below before you hand work over. The first six items take about ten minutes and prevent most of the damage.

01 The checklist

- ☐ **The task is a node, not a sentence.** The work has a place in the plan, with a name a machine can resolve. Not "finish the thing we discussed".
 - ☐ **"Done" is written down for that node.** One line. If you cannot write it, the node is too vague and will never close.
 - ☐ **The branch boundary is explicit.** State what the next agent owns, and what it must not touch.
 - ☐ **The standing brief is current.** Fold what you learned into it. Do not append. Replace the stale parts. A brief that is a diary gets skimmed, and a brief that gets skimmed is worse than none.
-

- ☐ **Decisions are recorded, with the rejected option.** One line each: what was settled, why, and what was rejected. This is the highest-value item on the list, because it stops the next agent reopening a closed question.

- ☐ **Blockers are named.** Not "some issues". Name the missing credential, access, decision or dependency, and who owns it.

- ☐ **Nothing secret is in the handoff.** No keys, tokens or credentials in the brief, the task note or the chat. If the next agent needs access, it needs a grant, not a password in a note.

- ☐ **You released the claim.** An un-released claim is a lock nobody can explain. Release it, or the next agent is blocked by you and cannot tell why.

- ☐ **You said why, if you stopped early.** "Blocked on X" is a legitimate and useful ending. A silent stop is not.

- ☐ **You left the next agent better placed than you found it.** That is the whole test. Everything above is how you pass it.

02 Template 1: the standing brief

Rewrite this each time. Keep it under a screen.

```
PROJECT: <name>
WHAT WE ARE BUILDING: <one sentence a stranger would understand>
PLAN OF RECORD: <where the tree lives; the plan is there, not in chat>

CONSTRAINTS (non-negotiable)
- <constraint> (why: <reason>)
- <constraint> (why: <reason>)

CONVENTIONS ALREADY SETTLED
- <convention>
- <convention>

CURRENT STATE
- Done: <what is finished>
- In flight: <what is claimed, and by whom>
- Next: <the one thing that matters next>

OPEN BLOCKERS
- <blocker> | owner: <who> | impact: <what it stops>
```

Why each constraint carries a reason. A constraint without one gets worked around the first time it is inconvenient. A constraint with a reason survives contact with a clever agent.

03 Template 2: the claim note

One sentence, written before the work starts, visible to whoever is watching.

```
CLAIMING: <task number and title>
APPROACH: <one sentence on how I mean to tackle it>
BRANCH: <what I will change>
WILL NOT TOUCH: <what is outside my branch>
```

The "will not touch" line is the one people skip, and the one that saves the most time. Two agents working adjacent branches is fine. Two agents editing the same file is not.

04 Template 3: the completion note

Written per finished piece, as it finishes. Not batched at the end. Batching makes the board fiction for the whole middle of the job.

```
COMPLETED: <task number and title>
WHAT CHANGED: <the actual artifact: file, URL, decision. Not "improved the thing">
EVIDENCE: <how to check it: a path, a command, a link>
STILL OPEN: <what this does not cover>
NEXT AGENT SHOULD KNOW: <the one thing that is not obvious>
```

The evidence line is not optional. "Drafted the article" is a claim. "Drafted the article: /drafts/post-1.md, 1,450 words, awaiting approval" is checkable. Agents over-report success. The evidence line is what makes the report falsifiable.

05 Template 4: the blocked release

The honest ending. Better than a lock nobody can explain.

```
RELEASING: <task number and title>
STATUS: blocked
BLOCKER: <the specific thing that is missing>
WHAT I TRIED: <so the next agent does not repeat it>
WHAT WOULD UNBLOCK IT: <the smallest thing that fixes this>
STATE OF THE BRANCH: <what I changed, what I left untouched>
```

06 Three rules that make the templates work

1. One claim at a time. If an agent can hold two claims, "who is doing this?" stops having an answer, and the second claim is the one that gets neglected.

2. Progress is derived, never reported. A parent is complete when its children are complete. Do not let anyone, human or agent, tick a parent by hand. Self-reported status is optimistic, and an agent that has "mostly finished" six things has finished none of them.

3. The handoff artifact lives on the task, not in the chat. Chat scrolls, gets summarised and loses the middle. The note on the task stays attached to the work, and the next agent reads it whether or not it read the conversation.

07 A worked example

A WEAK HANDOFF

"Task: write the blog posts. Context: see above. Should be straightforward."

What the next agent knows: a noun. What it does not know: how many, for whom, at what length, in what voice, what it may not claim, where the draft goes, what "done" means, or that a fact-checking rule exists. It will guess at all seven, and it will guess confidently.

A WORKING HANDOFF

Task 2.6.1, Post 1 of 3. Definitive explainer, drafts only.

Approach: write the brief first, then the draft; leave the approval child untouched.

Branch: 2.6.1 only.

Will not touch: 2.6.2, 2.6.3, any published page.

Done when: the draft is on the task and the approval node is left open for the human.

Constraint: no pricing, customer counts or benchmark figures. None are verified.

Blocker: none.

Same work, same agent, same day. The second version takes four minutes to write and removes almost every way the job can go wrong.

08 Why this is worth the four minutes

The cost of a bad handoff is not the rework. It is the confident wrong answer. An agent fills the gaps in a vague brief with plausible invention, delivers something polished, and is believed. By the time anyone notices, other work has been built on top of it.

A checklist will not make an agent smarter. It makes visible the places where it would otherwise guess.
That is the part you can control.

These templates are the handoff fields used in LatticeRun's claim, work and report protocol: the standing brief, the claim note, the completion note and the blocked release, written up generically so they work with any agent or tracker. No tooling is required to use them. Nothing in this document is a performance claim.