

BUILD THE FOLDER BEFORE THE FIRST SCENE

Give an agent AVS evidence while keeping the writer in charge.

Page one shows how to build and test the context folder. Page two shows how to use it for development, drafting, revision, and self-coverage. Anything set as dialogue is an instruction you can give the agent word for word; adapt names and file paths to the production.

PREPARED FOR THE AVS SCREENWRITING TEAM · JULY 2026

FADE IN:

THE CONTEXT FOLDER COMES FIRST

Before an agent touches a scene, create the project folder it will work in. Its files keep the team brief, evidence, and decisions available between sessions. A folder does not prove that the agent read every file, and it is not a security boundary. Require the agent to list the files it used.

Rule 1: The agent drafts the folder structure and its working files; you correct and approve each one.

Rule 2: The standard is evidence, not generic “good writing.” The archive may reveal patterns, exceptions, and contradictions.

Rule 3: Every judgment is traceable; every change is visible and reversible.

1. **Clear the preflight; create the folder.** Before adding material, confirm the team-approved tool and account, which client or organization files may be stored there, what the vendor retains, and who may access them. Name the folder something like `AVS-Team-Brain` and open it as the agent's working folder.

Initialize local version control to restore approved states. A snapshot labels a point in that history; it is not a backup. Do not send confidential material to a remote unless AVS approves its destination and access controls.

2. **Use an interview to draft the master brief and working agreement.** Your first session:

The Writer
(to the agent)

Interview me about the audience, deliverables, review stages, reviewers, and what gets praised or sent back. Follow up on unclear answers. Then PROPOSE `team-brief.md` and `working-agreement.md` in full. The agreement must require source citations, unchanged source scripts and raw notes, labeled inferences, PROPOSE for creative edits, named targets before changes, and a stop when sources conflict or cannot be read. Save nothing.

You should get: two full previews and no file changes. Next: correct them, then APPLY only those two approved files. After saving, ask the agent to name the loaded brief and working folder. If needed: “Read [brief path]. State its scope and your working folder before doing anything else.”

3. **Pair each draft with its notes.** Keep passed and sent-back drafts with their verbatim notes; record the date, reviewer, stage, and outcome. Preserve originals. Use the review PDF for page references and Fountain or plain text for working drafts; recheck pagination after conversion.

Test access first: “Open every supplied file. List each file read and each failure with its error. Change nothing.” If the archive is thin, mark criteria unknown — do not manufacture a rule.

4. **Have the agent draft a rubric that shows its work.**

The Writer
(to the agent)

PROPOSE only. Read every file in `evidence/`. Group the evidence by review stage and reviewer. For each proposed criterion, quote the source note and identify the draft. Label the criterion explicit, inferred, disputed, or production-specific. Record the reviewer, stage, date, any contrary examples, and confidence. Keep committee-wide patterns separate from individual reviewer preferences. Flag insufficient evidence. Show the proposed rubric and profiles. Change nothing.

You should get: proposed criteria and profiles with citations, context, uncertainty, and confidence, with no files changed. Next: correct each proposal, then issue a separate APPLY instruction naming only the approved files in `rubrics/`.

The rubric is a diagnostic working document, not a recipe or a prediction. One line might read:

Structure & pacing — **REVIEW WHEN** the inciting incident lands after p. 12 in the cited 30-minute examples. Evidence: Ryan's notes on [Draft A/date], [Draft B/date], [Draft C/date], and [Draft D/date]; example: “We're waiting too long for the engine to start.” Contrary examples: [none found or citations]. Confidence: medium. Check production-specific exceptions.

5. **Add shared files without creating competing versions.** Put standards, templates, and a why-it-passed note beside each exemplar. Keep one authoritative copy. Give each production a work area and record its shared-context version. Folder scope helps focus; verified permissions still control access.

THE FOLDER AT A GLANCE

AVS-Team-Brain/

- └ `CLAUDE.md` · `AGENTS.md` agent instructions; verify which file loads
- └ `team-brief.md` purpose, audience, deliverables, review path
- └ `working-agreement.md` citation, change, and approval rules
- └ `rubrics/` committee criteria + stage reviewer profiles
- └ `evidence/` raw drafts + paired notes, unchanged
- └ `exemplars/` approved scripts + why they passed
- └ `standards/` · `templates/` voice, format, repeatable forms
- └ `projects/` one work area + context version each

6. **Test the folder; update it after reviews.** After every approval or rejection, archive the notes verbatim. Have the agent propose cited rubric updates without changing files, then apply only the updates a person approves.

READY WHEN

The agent can state the audience, deliverable, and stage; name files read and unreadable; trace two criteria to original notes; and run one coverage pass without editing the draft.

CUT TO:

PUT IT TO WORK: THE DAILY PRACTICE

The folder does not make the agent a mind reader. It gives the agent sources, boundaries, and a way to show its work. Name the mode before each task:

ASK — analyze and report; change no files · **PROPOSE** — show options or a proposed file or diff; change no files · **APPLY** — make only specifically approved mechanical or record-keeping changes

Creative work defaults to **PROPOSE**.

Development: you supply the premise; the agent tests it. The idea remains yours. Ask for beat-level findings and cited alternatives, not a verdict:

The Writer
(to the agent)

PROPOSE only. Read `projects/[production]/outline-v3.md`, `rubrics/committee.md`, and the notes cited by that rubric in `evidence/[production]/`. Identify up to three act-two beats most at risk under those sources. For each supported concern, cite the rubric criterion and original note, explain the inference, and offer two alternatives that preserve the midpoint. If the sources support fewer than three, say so. Change nothing.

You should get: up to three cited beat-level concerns and two alternatives for each supported concern. *Next:* choose, reject, or request another option.

Drafting: you write; the agent supports the work. Use it for research from named sources, formatting, and continuity checks against `projects/[production]/story-bible.md`. You may ask it to rough out two versions to compare structure or rhythm, but keep those roughs as reference material, never canonical pages. You write the version that enters the script. Keep the review PDF canonical for pagination, and verify page references after conversion.

Revision: one note per pass. Use **APPLY** only for mechanical work you have decided: formatting, continuity fixes, dictated trims. Anything touching voice stays in **PROPOSE**:

The Writer
(to the agent)

PROPOSE only. In `projects/[production]/draft-v3.pdf`, address only the pacing note for pages 12–18. Quote each affected line and show one suggested change beneath it. Do not edit the file. Stop after the list.

You should get: a quoted, marked list and no file changes. *Next:* approve or reject each item. Then issue a separate **APPLY** instruction naming only the approved items and inspect the diff.

Save an approved snapshot. If the agent expands the scope or cites a source it cannot show, stop the pass.

The self-coverage gate. Nothing goes up for approval until the agent has read it like a story analyst:

The Writer
(to the agent)

ASK only. Read `projects/[production]/draft-v3.pdf` against `rubrics/committee.md` and `rubrics/[stage]-reviewer-profile.md`. Report what works, what current evidence suggests may be flagged, and where. Cite the criterion and original note behind each concern. Separate evidence, inference, and conflict. Do not propose fixes. Do not edit or save any file.

You should get: cited coverage with uncertainty and no fixes. *Next:* decide what, if anything, merits revision.

The gate is a preflight, not a simulation or prediction of a person. You decide what changes, in your own voice. Compare its concerns with the notes that actually come back; use the difference to calibrate the rubric.

Habits for every session.

- **State the goal and constraints**, not twenty micro-instructions: “tighten the teaser to two pages without losing the reveal.”
- **Name the mode and files.** Say what the agent may read and what, if anything, it may change.
- **Ask for evidence-backed options.** Then make the next decision yourself.
- **One work area per production.** Record its shared-context version and verify access; location alone is not permission.
- **Record decisions in files.** When you make a decision, write it to the story bible or decision log before the session ends.

WHEN THE AGENT MISSES

If a citation is missing, a source is unreadable, the scope is wrong, or the edit is too broad, stop. Inspect the source and diff. Restore only unauthorized changes, preserving any later human work, then narrow the request. A longer prompt does not repair a trust failure.

Guardrails. You are the author — the voice, credit, and responsibility are yours. Use only approved tools, accounts, and material; follow AVS and client policies, contract terms, and any applicable guild or union rules. Source scripts and raw evidence stay unchanged unless a person explicitly authorizes a change. Self-coverage is a gate, not a sign-off. Nothing is submitted outside the working team without human approval.

FADE OUT.