THE TERMINAL CATHEDRAL

TECHNICAL CASE STUDY / 001

BUILDING SANDWICH STUDIO

Human Decisions, AI Implementation, and the Bugs Between

A CASE ABOUT ACCOUNTABILITY, NOT AUTOMATION THEATRE

How a creator used AI agents to build a comic-production workspace - through design decisions, architectural trade-offs, broken assumptions, real agent trials and repeated verification.

DAN ALMER JENSEN

Independent creator | Copenhagen, Denmark

25 NOVEMBER 2026

TECHNICAL CASE STUDY

HUMAN DIRECTION / AI IMPLEMENTATION / ENGINEERING ACCOUNTABILITY

Contents

A guided route through the creative challenge, the engineering, the failures and the lessons.

Overview 03

The creative challenge 04

Human-directed collaboration 05

Engineering the studio 06

Field record 01 / Agent parity 07

Field record 02 / Reliability 08

Field record 03 / Codex pilots 09

Testing the system 10

Lessons and implications 11

References and credits 12

THE QUESTION BEHIND THE PROJECT

What does it take to keep a human in control while AI agents help build and operate creative software? This case follows the decisions that turned an image-generation experiment into a structured comic-production workspace - and the technical failures that forced the process to mature.

EXECUTIVE BRIEF

A product, a method, a test of control

The relevant question is not whether AI can write code. It is how a human retains meaningful direction while agents implement complex work.

IN ONE PARAGRAPH

Sandwich Studio is a creator-directed workspace for structured comic production: story and character references, artwork handling, editable lettering, revisions, previews and explicit review states. A central design goal was to let a person operate the workspace directly *or* instruct an agent to perform the same relevant operations. The creator retained authority over consequential creative decisions. [E1, E2, E5]

66
AGENT OPERATIONS
WO19 operation register

278/278
REGRESSION TESTS
WO19C engineering report

3
FIELD EPISODES
Architecture / defects / pilots

CORE THESIS

This case concerns **human-directed iteration**: framing the problem, giving bounded authority to agents, verifying their outputs, rejecting unsafe or unusable results and deciding when a change is worth keeping. The development history shows both the potential of agent-directed tools and the cost of achieving reliable, usable workflows. [E1, E5]

FIRST QUESTION

Can the agent act?

Can a creator delegate meaningful actions through chat without losing the possibility of direct manual control?

SECOND QUESTION

Who makes the call?

A technically successful edit is not automatically a creative approval. Selection and finalization are separate decisions.

WHAT MAKES THIS CASE DIFFERENT

The development record includes concrete requirement changes, agent-operation design, failed tests, corrections and hands-on Codex trials. Together, these reveal how accountability works in practice - not just how it is described in a methodology diagram. [E1-E6]

01 / THE CREATIVE CHALLENGE

A story in search of a system

A comic idea and an interest in controlling creative AI became the starting point for a much larger engineering project.

> *The creator wanted to tell a story - and to find out whether AI-generated imagery could remain consistent from page to page without losing characters, visual style or narrative intent.*
>
> CREATOR RECOLLECTION / PARAPHRASED FROM DANISH VOICE INTERVIEW

FROM IMAGE CONTROL TO PRODUCTION CONTROL

The initial curiosity was about image generation: persistent characters, repeatable visual style and page-by-page direction. Very quickly, the scope included reference management, scene continuity, changes, narrative sequencing and revision history. The creator recalls that a production environment was part of the idea almost from the start; what surprised him was the difficulty of making all its moving parts behave reliably.

CREATIVE PROBLEM

Keep a story coherent

Continuity across characters, panels, pages, scene references and evolving artwork.

ENGINEERING PROBLEM

Keep a process coherent

Preserve state, track revisions, inspect output and recover from bad operations.

INSIDE THE PRODUCTION WORKSPACE

The project combines story and reference material, artwork **import and replacement**, editable lettering, page previews, work orders, revision journals and distinct review states. The central engineering achievement is the workflow that connects these elements: managing context and change so each page can be inspected, revised and carried forward without silently losing an earlier decision. [E1, E2, E4, E5]

WHY THIS MATTERS

The engineering challenge was not to make a single image look convincing. It was to make a series of consequential creative operations **repeatable, reviewable and reversible**.

02 / THE WORKING METHOD

Three roles. One decision boundary.

AI-Madpakken gave the team a practical separation of analysis, human authorization and technical execution - with iteration between them.

COULD

LAYER 1 / CHATGPT

Frame options, risks, requirements and work orders. Critically review results.

V

SHOULD

HUMAN / DAN ALMER JENSEN

Choose direction. Approve consequential edits, version changes and public release.

V

DID

LAYER 2 / CLAUDE CODE & CODEX

Implement authorized work, run checks and report evidence and limitations.

NOT A RIGID ASSEMBLY LINE

Real development rarely moved in a perfectly linear COULD → SHOULD → DID sequence. Some work was HUMAN → DID → HUMAN; failures triggered another analysis cycle; routine technical steps could proceed inside bounded authorization. The central principle was not an exact number of handoffs but the separation between an agent's ability to act and a person's authority to decide. [E1, E2, E5]

WHY DECISION BOUNDARIES MATTER

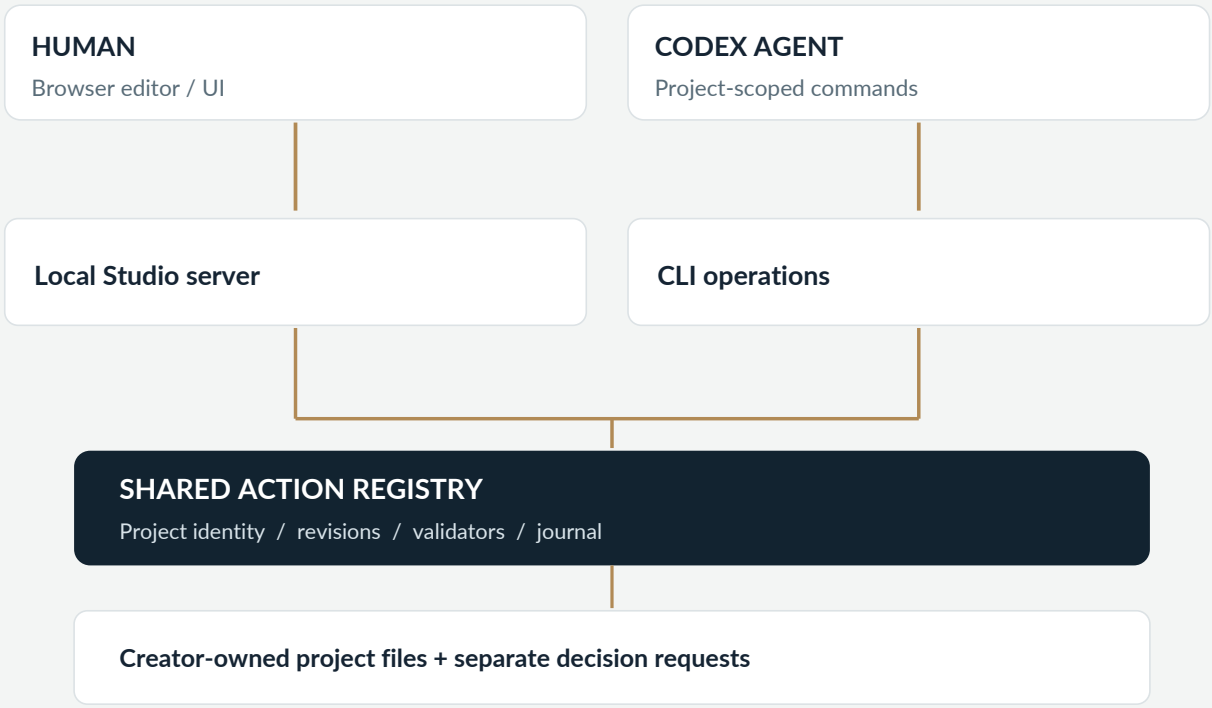
The interface distinguishes agent-executed actions from consequential creator decisions. An edit may be saved, reviewed and selected without automatically becoming final. Those state transitions make responsibility visible in the software itself. [E1, E2]

03 / ENGINEERING ANATOMY

Two interfaces. One project state.

A shared action layer reduced the gap between manual work and agent-directed work while keeping revisions and approvals explicit.

WORKSPACE ARCHITECTURE / HUMAN AND AGENT ROUTES



THREE DESIGN CHOICES

01 / REVISIONS

Do not overwrite intent

Saved, SELECTED and FINAL are distinct conditions. An accepted edit is not automatically final.

02 / RECOVERY

Upgrade with a route back

Versioned local installs and snapshots favor auditability over silent upgrades.

03 / ACCESS

Agent parity, not free rein

The UI and commands reach shared project actions; decisions stay bounded.

04 / VERIFY

Make evidence repeatable

Frozen builds, validation rules and targeted regressions turn a change into inspectable engineering work.

04 / FIELD RECORD ONE

When the agent needed the whole dashboard

A design change turned AI from a background helper into an alternate way to operate the same creative workspace.

> *The creator did not want a conventional interface with a little AI hidden behind it. He wanted to choose between doing the work manually and asking an agent to carry out the same relevant actions.*
>
> CREATOR RECOLLECTION / PARAPHRASED FROM DANISH VOICE INTERVIEW

THE DECISION

Creator testing exposed repeated movement between Studio and Codex. Rather than accept a narrow list of prepared agent tasks, the HUMAN required meaningful **UI/agent parity**. Layer 1 framed an authorized implementation request; Layer 2 reported a shared operation register with project identification, expected revisions, idempotent request IDs and structured error reporting. [E1, E2, E5]

66

DOCUMENTED AGENT OPERATIONS

WO19 described a shared 66-operation registry and a parity matrix covering creator actions. The associated engineering gate reported 177/177 tests passing.

WHAT CHANGED - AND WHAT DID NOT

The architecture was designed so a person could use the local browser UI or direct an agent through the available project-scoped commands. The creator also described using **Codex's existing remote access** to reach an agent running on his computer from a phone. That remote capability was not invented by Sandwich Studio; the contribution here was making Studio operable through agent actions. [E1, E5; creator recollection]

A DESIGN LESSON

Completing an operation and enjoying the workflow are different things. Creator testing exposed too many clicks, confirmations and status surfaces, leading to a simpler review experience focused on the decisions that mattered. [E5]

05 / FIELD RECORD TWO

The bugs between green checkmarks

The most revealing engineering evidence is often the defect that survived an earlier successful test.

FOUR KINDS OF FAILURE

01

ARTWORK INTEGRITY

JPEG artwork could appear blank in some preview or approved outputs; image orientation also caused inconsistency.

02

TRANSACTION SAFETY

Interrupted writes sometimes left incomplete or misleading project state, prompting recoverable write patterns.

03

PENDING-CHANGE TRUST

A crafted local change record exposed unsafe mutation behavior. Review demanded remediation; arbitrary local process security was not proved.

04

HUMAN APPROVAL

An early confirmation approach leaked its code into browser-accessible data; the design was replaced and its limits made explicit.

THE RESPONSE WAS NOT A SINGLE HEROIC FIX

The reports describe successive isolated stabilization passes from 0.3.1 through 0.3.3, with failures driving another round of changes, negative tests and review of a newly frozen build. Some fixes introduced additional regressions. Earlier candidates were preserved rather than quietly relabeled as successes. [E1, E5]

278 / 278

REGRESSION CHECKS IN THE WO19C REPORT

A documented engineering checkpoint after repeated stabilization passes. The real lesson was learning to test the failures that earlier green summaries had missed.

06 / FIELD RECORD THREE

A real agent, a wrong project, a useful correction

The first real Codex pilot illustrates how model capability can be undermined by something as ordinary as project identity.

01 / WRONG CONTEXT

A fresh Codex session entered an older but valid synthetic project and concluded that newer edit actions were missing.

02 / HUMAN INTERVENTION

The creator identified the project mismatch. The test was repeated against the intended project and version.

03 / BOUNDED EDIT

Codex reported moving a lettering balloon on a synthetic comic page and saving revision 3 for review.

04 / SEPARATE APPROVAL

The HUMAN reported approving that revision as SELECTED; a later postcheck reported that it had not been promoted to FINAL.

A SECOND PILOT: LOCAL VERSION SWITCHING

In a separate synthetic-project exercise, Codex checked the installed application versions and a recovery snapshot, requested explicit HUMAN approval and then switched the active Studio installation. The post-switch record marked **172 snapshot-covered project files IDENTICAL**. The exercise connected agent capability, human authorization and preservation of project state. [E5, E6]

THE ENGINEERING TAKEAWAY

An agent can be technically capable and still begin in the wrong context. Project identity, predictable actions and a distinct HUMAN decision boundary made the later edit and version switch understandable, reviewable and recoverable. [E5, E6]

07 / TESTING THE SYSTEM

Verification is a design activity

The project was repeatedly challenged with failures that ordinary success-path testing had not revealed.

THE RECURRING ENGINEERING LOOP

01 IMPLEMENT

02 CHALLENGE

03 REVIEW

04 CORRECT

05 RETEST

FOUR STRESS POINTS THAT CHANGED THE SYSTEM

RENDERING

Observed: JPEG artwork could appear blank in approved output.
Response: Unified image handling and more focused render checks followed.

DATA INTEGRITY

Observed: Interrupted writes could leave partial or confusing project state.
Response: Staged, validated and recoverable operations became central.

TRUST BOUNDARIES

Observed: A crafted pending-change record exposed unsafe behavior.
Response: Independent scrutiny and negative tests drove corrective changes.

CONTEXT BINDING

Observed: A Codex pilot started in a different valid test project.
Response: Project identity checks became a core usability and reliability concern.

278 / 278

REGRESSION TESTS / REPORTED IN WO19C

Repeated test cycles mattered because the team had to catch failures in rendering, state changes and agent actions - then verify each repair rather than assuming the first fix had worked.

08 / LESSONS AND IMPLICATIONS

Where the human responsibility lived

The case illustrates an engineering practice - not a proof that one workflow is universally superior.

01 **Delegation is a system-design problem.**

A useful agent interface required operations, scopes, identity checks and state semantics - not just better prompting.

02 **Approval is different from execution.**

An agent can complete an edit without deciding it belongs in the final creative artifact.

03 **Verification must include negative evidence.**

Missed JPEG rendering, incomplete writes and unsafe state handling mattered more than a reassuring green summary.

04 **Usability can defeat rigor.**

The creator rejected workflows where too many procedural interactions obscured a few meaningful decisions.

05 **An evidence trail is part of the product.**

Versioned builds, test receipts and the record of rejected candidates made decisions easier to inspect and revisit.

WHY IT IS WORTH PUBLISHING

For engineers, this is a story about requirements, interfaces, data integrity and recovery. For researchers, it offers a concrete example of human oversight and agent delegation. For anyone wondering what building with AI feels like in practice, it shows the less glamorous part: choosing, rejecting, checking and repeating. [E1-E6; editorial synthesis]

09 / REFERENCES AND CREDITS

The work behind the work

The case study connects engineering records, real agent trials and the creator's account of the decisions that shaped the product.

RESEARCH AND ENGINEERING SOURCES

[E1] WO19 technical engineering report: shared operation design, tests, defect findings and review states.

[E2] WO19 UI/agent parity matrix: creator actions mapped to the agent operation interface.

[E4] Project attribution, license notices and dependency inventory.

[E5] Dated development and work-order records: requirements, tests and revisions, WO18-WO22.

[E6] Creator-provided Codex pilot records, screenshots and follow-up checks.

[E10] Creator decisions and spoken recollections, including design goals and lived experience.

A NOTE ON RESEARCH MATERIAL

Technical counts and engineering outcomes are attributed to the relevant project records. Creator recollections are paraphrased rather than presented as verbatim quotations. This report was developed from the *Sandwich Studio R&D Evidence Dossier* (25 November 2026) and the associated project materials.

EXPLORE THE RELATED WORK

- **The Terminal Cathedral** OPEN >
- **AI-Madpakken / Human Sandwich Model** OPEN >
- **Creator profile on LinkedIn** OPEN >

DAN ALMER JENSEN

Independent creator / Hardware Design Engineer / Copenhagen, Denmark

terminalcathedral.com | github.com/TheTerminalCathedral