ACHOS PLAYER

Technical Manual — Appendix

COFI-Based Development Evidence

Documenting Human–AI Co-Creation Under the Co-Creative Framework for Interaction Design

Evidence Corpus: 49 sessions · 63 git commits · 31 structured prompt documents · 24 in-code diagnostic files

Development span: 44 days (2026-04-24 → 2026-06-06)

Martín Gallardo Arenas

Co-authored with Claude Sonnet 4.6 (Anthropic)

Santiago, Chile · 2026

1. Introduction

This appendix provides formal evidence that the development of Achos Player conforms to the academic framework known as COFI — the Co-Creative Framework for Interaction Design — as defined by Rezwana and Maher (2022, 2023) and published in ACM Transactions on Computer-Human Interaction.

Unlike the typical application of COFI to short creative sessions, the Achos corpus represents a sustained co-creative engagement of 44 consecutive development days. The evidence base is unusually rich: 49 identifiable sessions, 63 git commits co-authored by the human developer and the AI agent, 31 structured prompt documents written by the human before each major session, and 24 in-code diagnostic documents created by the AI agent during active debugging as living analysis artifacts.

Four sessions are analyzed in detail as case studies, selected to represent the full range of COFI interaction types present in the corpus: long-running iterative debugging with multiple failed attempts, non-deterministic AI behavior diagnosis, co-negotiation of architectural principles, and philosophy-driven design decision making.

Source document: "Achos (macamp) — Complete Interaction History with Claude AI" — reconstructed by Claude Sonnet 4.6 from 63 git commits, 31 Docs/ session files, in-code diagnostic documents, and Technical Manual v1.0. Prepared 2026-06-07.

2. Theoretical Framework

2.1 COFI: The Co-Creative Framework for Interaction Design

COFI was developed by Jeba Rezwana and Mary Lou Maher at the University of North Carolina at Charlotte, first presented at ICCC 2021 and published in full in ACM Transactions on Computer-Human Interaction (2023, Vol. 30, No. 5, Article 67). It defines Human–AI co-creativity as a process in which both parties collaborate on a shared creative artifact as partners, with interaction dynamics — including turn-taking, contribution type, and bidirectional communication — as the driving forces of the co-creative process.

COFI was specifically designed to address the gap between autonomous AI creativity tools (where AI produces, human curates) and passive creativity support tools (where human creates, AI assists). Achos occupies the center of that space: neither party could have produced it alone.

2.2 Core COFI Interaction Components

COFI Component	Definition
Turn-Taking	The alternating pattern of initiative — who initiates each contribution cycle and how control transitions.
Contribution Type	Function of each contribution: generative (new content), evaluative (assessment), or directive (steering).

Communication Channel	Medium of intent: text instruction, structured specification document, code, visual reference, in-code diagnostic document.
Contribution Similarity	Whether new contribution extends (near) or challenges (far) the partner's prior output.
Social Dynamics	Relational register: collaborative, Socratic, adversarial-constructive, or deferential.
Initiative Level	Degree of autonomy: from fully human-directed to AI-initiated, with mixed-initiative as the productive center.

2.3 The Docs/ File Pattern — A Novel COFI Communication Channel

The Achos corpus introduces a communication pattern not previously documented in the COFI literature: the pre-session structured specification document. Before each major development session, the human developer (H) wrote a Markdown document — ranging from 175 to 1,220 lines — that specified problem context, desired behavior, design constraints, and explicit rules. These files (stored in Docs/) represent H's intentional communication to the AI agent before the interactive session began.

This pattern extends COFI's "communication channel" component: instead of real-time text instruction only, H developed a practice of pre-formalizing intent in structured documents. The AI agent then had both the specification document and the live session dialogue as its input context. This is methodologically significant — it represents a more deliberate form of mixed-initiative setup than COFI's original formulation anticipated.

2.4 Primary References

Reference	Citation
COFI — canonical	Rezwana, J. & Maher, M.L. (2023). Designing Creative AI Partners with COFI. ACM Trans. Computer-Human Interaction, 30(5), Art. 67. DOI: 10.1145/3519026
COFI — original conference	Rezwana, J. & Maher, M.L. (2021). COFI: A Framework for Modeling Interaction in Human-AI Co-Creative Systems. ICCC 2021, pp. 444–448.
Mixed-Initiative Design Space	Lin, Z. et al. (2023). Beyond Prompts: Exploring the Design Space of Mixed-Initiative Co-Creativity Systems. arXiv:2305.07465.
MI-CCy Quantifier	Margarido, S. et al. (2025). MI-CCy Quantifier: A Framework for Quantifying Mixed-Initiative Co-creativity. Springer EPIA 2025, pp. 3–15.
Explainable AI for Designers	Zhu, J. et al. (2018). Explainable AI for Designers: A Human-Centered Perspective on Mixed-Initiative Co-Creation. IEEE CIG 2018. DOI: 10.1109/CIG.2018.8490433
AI Communication in Co-Creativity	Rezwana, J. & Maher, M.L. (2025). Human-Centered AI Communication in Co-Creativity. ACM C&C 2025. DOI: 10.1145/3698061.3726932

3. Evidence Corpus Overview

The complete Achos development corpus spans 44 days (2026-04-24 to 2026-06-06). The following table maps each version milestone to its session count, dominant COFI interaction type, and the documentary evidence available.

Version / Period	Sessions · Evidence · Primary COFI Dynamic
v0.1.0 (Apr 24) Initial commit	1 session · Git commit · Turn-taking: H establishes working foundation, A receives direction
v0.1.1–v0.1.2 (Apr 28–29) DSD foundation	2 sessions · Git commits · Generative contribution: FIR filter, streaming buffer, DSD metadata pipeline
v0.2.0 (Apr 30–May 3) Dual Mode + Discogs	2 sessions · Git commits · Directive + generative: H defines UI paradigm, A implements dual Digital/Physical tab architecture
v0.3.0 (May 12–13) Professional audio path	2 sessions · Git commits · Complex generative: HogMode, BitPerfect, DoP DSD64 — 4 new Swift modules
v0.3.x DSD Bug Sprint (May 16–17) ★ Case Study 1	7–9 sessions · 7 commits + 2 stash entries + 5 diagnostic docs · Mixed-initiative iterative debugging
v0.4.0 + DoP Fix (May 18) ★ Case Study 2 (partial)	3 sessions · 5 commits + 5 diagnostic docs · Root-cause diagnosis: wrong CoreAudio property identified
v0.5.0–v0.5.2 (May 19–24) Classification + Spectral	3 sessions · 3 commits · Generative: 9-category engine, dual FFT, ITU-R BS.1770-4 True Peak
v0.6.0 (May 26–28) Playlist + Library	4 sessions · 4 commits · Directive + generative: AppKit NSTableView, GRDB SQLite schema, FTS5
v0.7.0 (May 28–30) Discogs OAuth + Collection	4 sessions · 3 commits · Generative: OAuth 1.0a PLAINTEXT, collection sync, pressing intelligence
v0.7.3–v0.8.x (May 30–31) Sandbox + Keychain	5 sessions · 4 commits · Diagnostic + generative: iterative auth prompt elimination
v0.8.0 (pre-git) AI Filemaster	2 sessions · Docs files (1,220-line UI spec) · Directive: H specifies every visual zone; A implements
v0.9.0 (Jun 2) GRaiL	1 session · 2 commits · 391-line brief · Mixed-initiative: H defines philosophy + persona, A implements 5 parts in single session
v0.9.1 (Jun 5) ★ Case Study 3 — 3-Queue Model	4+ sub-sessions · 1 commit · Docs files · Co-negotiation of architectural principle
v0.9.1–v0.9.2 (Jun 5) Guardian's Path + Gold system	6 sessions · Docs files · Directive: H specifies dot colors, philosophy, design language; A implements

v1.0.0 (Jun 6) ★ Case Study 4 — Verdict Taxonomy	5 sessions · 3 commits · Docs files · Philosophy-driven redesign: epistemic honesty over technical accuracy
v1.0.0 Polish (Jun 6) Onboarding + Menubar	3 sessions · 2 commits · Docs files (367 + 334 lines) · Directive + generative

★ = Session selected for detailed case study analysis in Section 4.

Corpus Metric	Value
Total sessions identified	~49
Total git commits	63 (including 2 stash entries)
Docs/ structured prompt files	31 (ranging 175–1,220 lines each)
In-code diagnostic .md files created by AI	24+
Development span	44 days (2026-04-24 → 2026-06-06)
Swift source files created	~80
AI features shipped	3 (Filemaster, GRaiL, Guardian's Path)
External API integrations	2 (Discogs OAuth 1.0a, Google Gemini)
Git commit co-author	Claude Sonnet 4.6

4. Case Studies

Four sessions are analyzed in detail, selected to illustrate the full range of COFI interaction dynamics present across the 44-day corpus. Each analysis maps documented evidence — commit messages, Docs/ file excerpts, in-code diagnostic documents — to specific COFI components.

Case Study 1 — DSD Decimation & Speed Bug Sprint (Sessions 7–9)

Session IDs	Sessions 7, 8, 9 (v0.3.x fixes)
Dates	2026-05-16 – 2026-05-17
Commits	403c54a → 94d8a9a (7 commits + 2 git stash entries = 9 attempts)
COFI type	Iterative Mixed-Initiative Debugging — the most iterative session in the corpus
Diagnostic docs created	DSD_PIPELINE_FIX.md · DSD_OPTIMIZATION_SUMMARY.md · THREE_ADDITIONAL_DSD_FIXES.md · COMPLETE_DSD_FIX_SUMMARY.md · THREE_CRITICAL_DSD_FIXES.md
Artifact produced	Correct DSD playback speed in non-exclusive path + pre-buffer guard + DFF metadata rate fix

4.1.1 Problem Statement

After implementing DoP DSD64 playback in v0.3.0, DSD files played at the wrong speed in the non-exclusive (shared) audio path. The root cause was a mismatch between the decimation ratio applied in the DSD decoder and the clock rate of the output device. This type of bug — where the error is a ratio relationship between two independently configured subsystems — is characteristically resistant to linear debugging because each subsystem appears internally consistent.

4.1.2 The 9-Attempt Iteration Log

The following reconstruction is drawn from git commit messages and in-code diagnostic documents. It demonstrates the mixed-initiative pattern most clearly visible in the corpus: the AI agent proposes a fix, the human tests it in the real hardware environment, and the result drives the next hypothesis.

Attempt	Evidence · Action · Outcome
Attempt 1 commit 403c54a	fix: DSD decimation matches device clock rate Root cause identified. Initial fix applied. Problem persisted — speed still wrong in some configurations.
Attempts 2–3 stash d7d11fb, d22e4fa	WIP / index Exploratory branches. Two concurrent hypotheses tested. Neither fully resolved. Both stashed — AI agent preserves both approaches for evaluation.

Attempts 4–5 commits 311a9c7, 8d6d348	fix: DSD non-exclusive path — correct speed, isolated from exclusive path Two sequential attempts to isolate the non-exclusive path from the exclusive path. Second refined the first based on real-device test results.
Attempt 6 commit 84958b7	fix: non-exclusive path isolated, DSD correct speed Non-exclusive path fully isolated. Speed confirmed correct on real hardware. ✅
Attempt 7 commit 7f5c2f1	fix: pre-buffer guard — DSD only New issue discovered during speed validation: pre-buffer logic firing on PCM tracks. Guard added.
Attempt 8 commit 150d854	fix: PCM non-exclusive path — no delay, BitPerfect isolated PCM path validated clean. No startup delay. BitPerfect path confirmed isolated from DSD changes.
Attempt 9 commit 94d8a9a	fix: DSD seek bar duration, DFF metadata rate corrected Seek bar showing incorrect duration for DSD files. DFF metadata rate parsing also corrected. Two additional issues discovered and fixed during final validation.

4.1.3 COFI Turn Mapping

Turn	Party	Contribution Type	COFI Component
T1	H → A	Problem report: DSD plays wrong speed	Initiative: Human-initiated; Communication: behavioral observation
T2	A → H	Hypothesis: decimation ratio mismatch. Fix #1 generated	Contribution type: Generative diagnostic + fix
T3	H → A	Evaluative rejection: Fix 1 applied to hardware — speed still wrong. H returns null result, does not accept partial fix	Social dynamics: ADVERSARIAL-CONSTRUCTIVE — H rejects AI output with real-world evidence; forces deeper investigation
T4	A → H	Two parallel hypotheses stashed (d7d11fb, d22e4fa) — AI does not defend Fix 1, proposes broader exploration	Initiative: Mixed; Response to adversarial-constructive friction → AI escalates from linear to branching strategy
T5	H → A	Directive: test both branches, evaluate results	Turn-taking: H authorizes exploratory approach
T6-T7	A → H (×2)	Attempts 4–5: isolation of non-exclusive path, two iterations	Contribution similarity: NEAR (incremental) then FAR (rearchitecture)
T8	H → A	Evaluative: Attempt 6 works on real hardware	Social dynamics: convergent validation
T9-T12	A then H (×4)	Attempts 7–9: three additional issues discovered and fixed during validation sweep	Mixed initiative: AI discovers new issues during fix verification

4.1.4 COFI Analysis

This session is the clearest example in the Achos corpus of what COFI terms iterative mixed-initiative co-creation under real-world constraints. The interaction extends across 9 attempts not because of communication failure but because the hardware environment introduces ground truth that neither party can predict: each fix must be tested on a physical DAC connected to a macOS machine, and only that real-world test can validate or invalidate a hypothesis.

The two git stash entries (Attempts 2–3) are analytically significant. They represent the AI agent maintaining two parallel hypotheses simultaneously — a form of non-linear contribution that is unusual in standard command-response workflows but natural in co-creative debugging. The human participant evaluated both branches rather than forcing a premature convergence.

The in-code diagnostic documents created during this sprint (DSD_PIPELINE_FIX.md, COMPLETE_DSD_FIX_SUMMARY.md, etc.) represent an extension of COFI's communication channel component: the AI agent produced living analysis documents inside the codebase as a form of shared working memory — notes that both parties could reference across sub-sessions. This pattern is not captured in COFI's original formulation and represents a novel contribution of the Achos corpus to the framework.

Case Study 2 — DoP Capability Detection Root Cause (Session 11)

Session ID	Session 11 (v0.4.0 + DoP exclusive fix)
Date	2026-05-18
Commits	8ea5730 (UI rate fix), f18878d (DoP capability check)
COFI type	Root-cause diagnosis with property-level technical correction
Diagnostic docs	DOP_FIX_SUMMARY.md · DOP_SHARED_MODE_FIX.md · DOP_EXCLUSIVE_MODE_ENFORCEMENT.md · DOP_ACTIVATION_FIX.md · DOP_CLOCK_SYNC_DELAY_FIX.md
Artifact produced	DSD64 and DSD128 DoP working correctly via hardware — DAC reports native 2.8MHz and 5.6MHz

4.2.1 Problem Statement

After the DSD speed bug sprint, DSD64 and DSD128 DoP playback via hardware remained non-functional. The symptom was deceptive: the software appeared to complete the DoP activation sequence without error, but the DAC received incorrect data. Three additional issues were reported simultaneously: the UI displaying tag metadata sample rate instead of the engine-detected rate; DSD in Hog-no-BitPerfect mode displaying 88.2kHz instead of the correct 176.4kHz; and a 3–4 second startup delay for AAC/OGG files in the BitPerfect path.

4.2.2 The Wrong-Property Root Cause

The most analytically significant finding in this session was the identification of the wrong CoreAudio property in the DoP capability check. The original implementation used:

```
kAudioDevicePropertyClockSources  ← WRONG: checks current clock state
```

The correct property is:

```
kAudioDevicePropertyAvailableNominalSampleRates  ← CORRECT: checks device
capability
```

This is a subtle but decisive distinction: clock sources reflect the device's current state, not its capabilities. A device that supports DoP but is not currently in DoP mode will report no active DoP clock source — causing the capability check to fail spuriously. The available nominal sample rates, by contrast, report what the device can do regardless of current state.

A second compounding issue was also identified: a 500ms clock-sync delay was firing before the device rate change had settled, causing DoP marker bytes to arrive before the DAC was ready to interpret them. Removing this mis-timed delay was part of the same fix.

4.2.3 COFI Turn Mapping

Turn	Party	Contribution Type	COFI Component
T1	H → A	Bug report: 4 issues with behavioral descriptions	Communication channel: structured multi-issue report
T2	A → H	Fix 1: UI reads engine rate, not tag metadata	Contribution type: Generative + rationale (tag metadata unreliable for hi-res FLAC)
T3	H → A	Evaluative: UI fix accepted	Turn-taking: H validates, authorizes Fix 2
T4	A → H	Root-cause identification: wrong CoreAudio property. Fix 2 generated	Initiative: AI independently identifies property-level error — not prompted by H
T5	H → A	Evaluative: Fix 2 tested on real hardware. DSD64 → DAC: 2.8MHz ✅, DSD128 → 5.6MHz ✅	Social dynamics: convergent validation with measurable result
T6	A → H	5 diagnostic documents created in codebase	Communication channel: in-code shared working memory

4.2.4 COFI Analysis

This session illustrates the AI agent operating with genuine technical initiative. The identification of kAudioDevicePropertyClockSources as the wrong property was not prompted by the human — the human reported behavioral symptoms, and the AI independently traced those symptoms to a specific API misuse. This is an instance of autonomous contribution within a co-creative frame, as described by COFI: the AI does not merely respond but produces an original analytical conclusion that the human had not articulated.

The five diagnostic documents created in this session extend the pattern identified in Case Study 1: the AI agent externalizes its reasoning into the shared codebase as persistent artifacts. These documents served as reference material for subsequent sessions where related DoP issues emerged, demonstrating that the co-creative output extended beyond the code itself to include a knowledge base that both parties could rely on.

Case Study 3 — The Three-Queue Architecture (Session 31, Bugs #12–14)

Session ID	Session 31 sub-session (v0.9.1 — Bugs #12–14)
Date	2026-06-05
Commit	481b3c9 (part of 18-fix stabilization)
Docs file	v0.9.1-achos-091-fix12-13-14-threading.md
COFI type	Co-negotiation of architectural principle — human establishes invariant, AI operationalizes it
Artifact produced	Three-queue model: audio queue / main queue / library queue, strictly separated throughout codebase

4.3.1 The Design Principle

During a stabilization sprint addressing 18 bugs in a single session, three threading issues emerged that shared a common root: the boundaries between the audio engine, the main thread, and database operations were not being respected consistently. Rather than addressing each bug in isolation, the human developer formalized the invariant that would govern all future threading decisions:

> *"The audio engine must NEVER wait for the main thread. The main thread must NEVER wait for database or file I/O."*

This statement — taken verbatim from the Docs/ session file — is not a bug report or a feature request. It is a design principle: an architectural commitment that constrains all future implementation decisions. Its articulation represents a specific type of human contribution in the COFI framework: a directive contribution that does not specify what to build but how to evaluate any proposed solution.

4.3.2 What Was Built From the Principle

The AI agent operationalized the principle into a concrete three-queue model:

Queue	Responsibility · Constraint
Audio queue	AVFoundation + FFmpeg operations. Never blocks. Never waits for main or library queue. Pre-existing.

Main queue	SwiftUI rendering only. Never executes I/O. Never waits for audio or library queue. Enforced in this session.
Library queue	Dedicated serial Swift actor for all database reads/writes, FSEvents callbacks, and folder scanning. New in this session.

Three concrete fixes followed directly from applying the model:

- Bug #12 — Library browsing interrupting playback: dbQueue.read moved off main queue via Task.detached
- Bug #13 — Playlist eager start: first track plays immediately without waiting for library index
- Bug #14 — FSEvents folder watcher: FolderWatcher.swift with 2-second debounce, library updates without main-thread involvement

4.3.3 COFI Turn Mapping

Turn	Party	Contribution Type	COFI Component
T1	H → A	Design principle as invariant: audio never waits for main, main never waits for I/O. Implicitly rejects prior approach of fixing threading bugs in isolation	Social dynamics: ADVERSARIAL-CONSTRUCTIVE — H refuses to accept per-bug patches; forces architectural rethink. Friction outcome: three-queue model governs all remaining development
T2	A → H	Architectural proposal: three-queue model as operationalization of invariant	Initiative: AI proposes concrete structure from abstract principle
T3	H → A	Evaluative acceptance: principle correctly operationalized	Social dynamics: convergent — human validates AI's architectural interpretation
T4	A → H	Generative: three fixes derived from model (Bugs #12, 13, 14)	Contribution similarity: NEAR — each fix applies the same model to a different site
T5	H → A	Evaluative: all three fixes tested, threading confirmed clean	Turn-taking: Human validates across all three fix sites

4.3.4 COFI Analysis

This session represents the most philosophically interesting instance of co-creative emergence in the Achos corpus. The human contribution (the invariant statement) is abstract — it does not specify a data structure, a pattern, or a solution. The AI agent's contribution (the three-queue model) is concrete — it translates the abstract invariant into a specific architectural decision with precise boundaries.

Neither the invariant nor the model alone constitutes the co-creative output. The co-creative artifact is the combination: a codebase-wide threading discipline that derives its authority from the human's stated principle and its implementation from the AI's architectural interpretation. This matches what COFI describes as emergent co-creative output — an artifact whose form and content cannot be attributed to either party independently.

The three-queue model subsequently governed every new async operation through v1.0.0. It appears in the "Key Design Decisions Made With Claude" section of the project's interaction history summary as Decision #1. This persistence — an architectural decision made in a single debugging session that shaped all subsequent development — is evidence that the co-creative output exceeded the immediate session boundary.

Case Study 4 — Filemaster Verdict Taxonomy Redesign (Session 45)

Session ID	Session 45 (v1.0.0)
Date	2026-06-06
Commit	bec5403
Docs file	achos-v100-filemaster-verdict-taxonomy.md (350 lines)
COFI type	Philosophy-driven design decision — epistemic principle replaces technical classification
Artifact produced	Redesigned verdict taxonomy: .verified / .inconclusive / .upsampled / .converted with epistemic honesty framing

4.4.1 The Triggering Event

The Filemaster AI feature had been shipping with a verdict set that used accusatory language: .authentic, .suspect, .upsampled, .fraudulent. During real-world testing, a known legitimate recording surfaced an edge case. The human developer documented the issue precisely in the Docs/ file:

> *"The current verdict set uses accusatory language that overstates the certainty of spectral inference. A real-world case (Gayle Skidmore TTK0015 — DXD-sourced DSD, fully disclosed by NativeDSD) was correctly identified as non-native DSD but incorrectly labelled 'Fraudulent Hi-Res'. The new taxonomy is epistemically honest."*

The Skidmore TTK0015 case is significant: the file in question was legitimately produced from a DXD master converted to DSD — a common mastering practice in the audiophile market. Labeling it "Fraudulent" was technically defensible (it is not native DSD) but ethically incorrect (the source disclosed the conversion). The human developer identified this not as a bug in the classification algorithm but as a philosophical error in the language used to present results.

4.4.2 The New Taxonomy

Old verdict → New verdict	Design rationale

.authentic → .verified Label: "Verified"	No change — authentic verified files correctly labeled. Action: Keep.
.suspect → .inconclusive Label: "Inconclusive"	".suspect" implies guilt without evidence. ".inconclusive" accurately describes what spectral analysis can determine. Action: Review.
.upsampled → .upsampled Label: "Likely Upsampled"	Retained but reworded to "Likely" — spectral inference is probabilistic, not definitive. Action: Consider replacing.
.fraudulent → .converted Label: "Likely Converted"	".fraudulent" imputes intent. ".converted" is descriptive without moral judgment. Crucially: does NOT force Replace — a converted file may be high quality and legitimately disclosed. Action: Keep if quality satisfactory.

The key design decision encoded in the new taxonomy: "Likely Converted does NOT force Replace." A DXD-to-DSD conversion may be fully disclosed, of high quality, and legitimately marketed. The Filemaster verdict describes the recording chain, not the listening experience. This distinction — drawn explicitly in the Docs/ file — was implemented directly by the AI agent.

4.4.3 COFI Turn Mapping

Turn	Party	Contribution Type	COFI Component
T1	H → A	Triggering event: Skidmore TTK0015 case exposes ethical failure of .fraudulent label. H explicitly rejects prior taxonomy with real-world counterexample	Social dynamics: ADVERSARIAL-CONSTRUCTIVE — H rejects AI-implemented taxonomy on epistemic grounds, not technical ones. Friction outcome: redesigned verdict set adopted as product philosophy
T2	A → H	Proposed new taxonomy with four cases, labels, and action recommendations	Contribution type: generative framework; Initiative: AI proposes complete taxonomy from H's principle
T3	H → A	Evaluative: taxonomy accepted. Specific instruction: "Likely Converted does NOT force Replace"	Contribution similarity: NEAR — H refines one case; overall structure accepted
T4	A → H	Generative: full implementation — Verdict enum, display strings, Gemini prompt updated, Technical Manual updated	Contribution type: generative (code + documentation)
T5	H → A	Evaluative acceptance confirmed	Social dynamics: convergent

4.4.4 COFI Analysis

Session 45 demonstrates the most philosophically sophisticated co-creative exchange in the Achos corpus. The human contribution is not a bug report, a feature specification, or a performance requirement — it is an ethical position about how AI-generated verdicts should be communicated to users. The triggering event (the Skidmore TTK0015 case) is presented not as a classification error but as a failure of epistemic honesty.

The AI agent's response — generating a complete taxonomy that operationalizes the principle of epistemic honesty — mirrors the pattern in Case Study 3, where an abstract principle was translated into concrete structure. In both cases, the human articulates the "why" and the AI constructs the "what."

The update to the Technical Manual (committed in the same session) extends the co-creative output beyond the codebase: the redesigned taxonomy is documented in the product's narrative history, preserving the reasoning chain for future development decisions. This represents a form of co-creative knowledge preservation not described in COFI's original formulation but consistent with its emphasis on communication as the foundation of effective co-creative partnership.

5. Cross-Session Analysis

5.1 COFI Component Distribution Across the Full Corpus

COFI Component	Presence in Corpus (estimated)
Turn-taking (alternating initiative)	49/49 sessions (100%) — every session involves multiple exchange cycles
Contribution Type: Generative	49/49 (100%) — AI produces code, analysis, documentation in every session
Contribution Type: Evaluative	~40/49 (82%) — H challenges, accepts, or rejects AI proposals
Contribution Type: Directive (H)	~45/49 (92%) — H provides scope, constraints, principles
Contribution Similarity: FAR (challenge/rejection)	~20/49 (41%) — notable challenges include DSD root-cause reframes, taxonomy redesign, threading invariant
Contribution Similarity: NEAR (extension)	~42/49 (86%) — most sessions involve incremental refinement after initial direction
Mixed Initiative (AI proposes unsolicited)	~15/49 (31%) — strategy surfacing (market analysis), root-cause identification (CoreAudio property), architectural proposal (3-queue model)
Communication Channel: Structured Docs/ file	31/49 (63%) — unique channel pattern in this corpus
Communication Channel: In-code diagnostic document (AI-authored)	24+ sessions (49%+) — shared working memory pattern
Communication Channel: Git commit (co-authored artifact)	63 commits across 49 sessions — every code artifact is jointly signed

5.2 The Docs/ File Pattern — A New COFI Communication Channel

The most structurally novel finding in the Achos corpus is the Docs/ file communication pattern. Across 31 sessions, the human developer wrote structured specification documents ranging from 175 to 1,220 lines before initiating the interactive session. These documents served simultaneously as:

- Input context for the AI agent (replacing or augmenting real-time instruction)
- A record of the human's design intent at the time of the session
- A constraint set that the AI agent could reference during implementation
- A verifiable artifact linking the human's intent to the AI's output

This pattern is not described in COFI's original formulation, which focuses primarily on real-time turn-taking. The Achos corpus suggests that at sufficient complexity, Mixed-Initiative co-creation benefits from a pre-session formalization phase — a structured communication event that precedes and shapes the interactive session. The 1,220-line UI specification for AI Filemaster

(Session 28) is the most extreme example: the human's contribution to that session was a document longer than most academic papers, specifying every visual zone, color value, and interaction behavior of a complex window.

5.3 Adversarial-Constructive Dynamics Across the Corpus

The COFI category of adversarial-constructive social dynamics — where one party explicitly rejects the other's proposal and the friction drives a superior outcome — is the most diagnostically significant component for distinguishing genuine co-creation from assisted execution. The Achos corpus contains four documented instances where this dynamic is traceable from rejection to improved artifact.

Session · Rejecting party	Rejection · Friction · Improved outcome
Session 14 · H rejects threshold brickwall detection	H identifies false positives on warm/dark recordings. AI's threshold-based algorithm is rejected as insufficiently discriminating. Friction: AI proposes slope-based method (>30dB across 10 consecutive FFT bins). Outcome: algorithmically robust detection that correctly handles analogue jazz, vintage recordings, and dark masters — a materially better classifier than the original.
Sessions 23–24 · H rejects Fix 1 as insufficient	AI's update-or-add Keychain pattern reduces prompts but does not eliminate them. H rejects partial solution and returns with persistent symptom. Friction: AI diagnoses SwiftUI render loop as the true cause. Outcome: in-memory credential cache — Keychain read exactly once per session. 5+ auth prompts → 0.
Session 31 · H rejects per-bug threading patches	AI is fixing three threading bugs (12, 13, 14) as isolated issues. H rejects this approach by formulating an invariant: "The audio engine must NEVER wait for the main thread." Friction: AI proposes three-queue model as architectural operationalization. Outcome: a codebase-wide threading discipline that governed all subsequent development through v1.0.0.
Session 45 · H rejects .fraudulent verdict category	AI-implemented Filemaster uses .fraudulent as a verdict for non-native-DSD files. H rejects this on epistemic grounds using the Skidmore TTK0015 counterexample (DXD-sourced DSD, fully disclosed). Friction: H articulates the principle "the verdict describes the recording chain, not the listening experience." Outcome: redesigned four-case taxonomy built on epistemic honesty — the most philosophically significant design decision in the project.

A structural pattern is visible across all four instances: the human's rejection is never arbitrary. In each case, H presents a specific counterexample or principle (real hardware result, persistent symptom, architectural invariant, named real-world file) that makes the inadequacy of the prior AI output concrete and undeniable. This precision — rejecting with evidence rather than preference — is what makes the friction constructive rather than merely disruptive.

This pattern is consistent with what COFI describes as the most productive form of social dynamics in co-creative systems: adversarial-constructive exchanges where challenge is grounded in shared goals and specific evidence, not in stylistic disagreement or authority. In the Achos corpus, every adversarial-constructive event produced an outcome that both parties could identify as superior to what either had proposed before the friction.

5.3 Co-Authorship as Evidence

Every git commit in the Achos codebase carries the co-author attribution:

```
Co-authored-by: Claude Sonnet 4.6 <claude@anthropic.com>
```

This is not a courtesy attribution. It reflects the technical reality that each commit's content was generated by the AI agent under the human's direction, and neither party produced it alone. In COFI terms, the git history is a timestamped record of co-creative output — 63 artifacts, each with a traceable provenance to a specific human-AI interaction cycle.

5.4 Key Design Decisions as Co-Creative Artifacts

The interaction history document identifies 12 key design decisions made with Claude. Each represents a co-created conclusion that neither party held independently before the relevant session:

Decision	Origin in co-creative exchange
Three-queue model	H's invariant + AI's architectural operationalization (CS3)
Guardian's Path philosophy "no GRaiL without verified provenance"	H's epistemic principle + AI's gating implementation (Sessions 30, 33)
Slope-based brickwall detection	AI identifies false positive pattern on warm recordings; H approves revised algorithm (Session 14)
Dual FFT architecture	H requests independent subsonic analysis; AI proposes separate decimated-signal pass (Session 14)
Genre-aware thresholds	H identifies spectral bias against electronic music; AI proposes 8-bucket genre system (Sessions 41–43)
Epistemic honesty in verdicts	H triggers redesign with Skidmore case; AI generates epistemically honest taxonomy (CS4)
Gold as AI design language	H specifies visual consistency requirement; AI proposes gold as AI color across all features (Session 37)
DSD bitDepth=1 invariant "never fall back to 16"	AI identifies non-determinism root cause: nil bitDepth → 16 misreports DSD to Gemini (Session 29)
In-memory Keychain cache	AI diagnoses repeated prompt pattern from SwiftUI render-triggered Keychain reads (Session 24)
DoP capability detection via available sample rates	AI identifies wrong CoreAudio property as root cause (CS2)
Gemini temperature 0.15	AI recommends determinism-focused temperature after non-determinism diagnosis (Session 39)
Pre-flight volume check before bookmark resolution	AI identifies sandbox auth trigger from unmounted volume resolution attempts (Session 25)

6. Conclusion

This appendix has demonstrated, through a 44-day corpus of 49 documented sessions and four detailed case studies, that Achos Player was developed through a process that conforms to the academic definition of Human–AI Co-Creation as formalized in the COFI framework (Rezwana & Maher, 2022/2023) and the Mixed-Initiative Co-Creativity literature (Lin et al., 2023; Margarido et al., 2025).

Five conclusions follow from the evidence:

1. All six COFI interaction components are present and documentable across the corpus. The evidence base — 63 co-authored git commits, 31 structured prompt documents, and 24 in-code diagnostic documents — provides an unusually dense record for a COFI analysis.
2. The interaction pattern is consistently mixed-initiative across all 49 sessions. The human developer exercised directive, evaluative, and challenge contributions throughout; the AI agent contributed generative, diagnostic, and architectural proposals that were not prompted by explicit instruction.
3. The Achos corpus introduces a novel COFI communication channel not previously documented in the literature: the structured pre-session specification document (Docs/ file). This pattern, present in 31 of 49 sessions, represents a deliberate formalization of human intent before the interactive session — an extension of COFI's communication channel component.
4. Multiple co-creative artifacts — the three-queue model, the epistemic verdict taxonomy, the dual FFT architecture, the DSD bitDepth=1 invariant — emerged from the human-AI dialogue and cannot be attributed to either party independently. These 12 identified key design decisions constitute the most durable evidence of co-creative emergence in the corpus.
5. Achos is the first audiophile-grade macOS music player to be developed under a documented COFI-conformant co-creative process. Its development methodology is not an informal or ad hoc use of AI assistance — it is a 44-day record of Mixed-Initiative Human–AI Co-Creation, with traceable provenance from human intent to co-authored code.

---

7. References

Lin, Z., Kreminski, M., & Chung, J.J.Y. (2023). Beyond Prompts: Exploring the Design Space of Mixed-Initiative Co-Creativity Systems. arXiv:2305.07465.

Margarido, S., Roque, L., Machado, P., & Martins, P. (2025). MI-CCy Quantifier: A Framework for Quantifying Mixed-Initiative Co-creativity in Human-AI Collaborations. In Progress in Artificial Intelligence (pp. 3–15). Springer, Cham. ISSN: 1611-3349.

Rezwana, J., & Maher, M.L. (2021). COFI: A Framework for Modeling Interaction in Human-AI Co-Creative Systems. Proceedings of ICCC 2021, pp. 444–448.

Rezwana, J., & Maher, M.L. (2023). Designing Creative AI Partners with COFI: A Framework for Modeling Interaction in Human-AI Co-Creative Systems. ACM Transactions on Computer-Human Interaction, 30(5), Article 67. DOI: 10.1145/3519026

Rezwana, J., & Maher, M.L. (2025). Human-Centered AI Communication in Co-Creativity: An Initial Framework and Insights. Proceedings of ACM C&C 2025. DOI: 10.1145/3698061.3726932

Zhu, J., Liapis, A., Risi, S., Bidarra, R., & Youngblood, G.M. (2018). Explainable AI for Designers: A Human-Centered Perspective on Mixed-Initiative Co-Creation. IEEE CIG 2018. DOI: 10.1109/CIG.2018.8490433

Achos Player Technical Manual — Appendix A: COFI-Based Development Evidence

Version 4.0 (final) · June 2026 · Martín Gallardo Arenas · Santiago, Chile

Evidence corpus: 49 sessions · 63 commits · 31 Docs/ files · 24 diagnostic documents · 4 adversarial-constructive events · 44-day span

Source: "Achos (macamp) — Complete Interaction History with Claude AI" · Reconstructed 2026-06-07