TRAX: Bridging the Gap in Asynchronous Music Collaboration

JAYWOONG JEONG, TRAX, Republic of Korea and KAIST, Republic of Korea
DONGWOO KIM, TRAX, Republic of Korea and GIST, Republic of Korea
JAEHONG JUNG*, TRAX, Republic of Korea and GIST, Republic of Korea
DOHYUN KO*, TRAX, Republic of Korea and GIST, Republic of Korea
JIWON EOM*, TRAX, Republic of Korea and KAIST, Republic of Korea
SANGMIN LIM*, TRAX, Republic of Korea and GIST, Republic of Korea
JUCHAN LEE†, TRAX, Republic of Korea

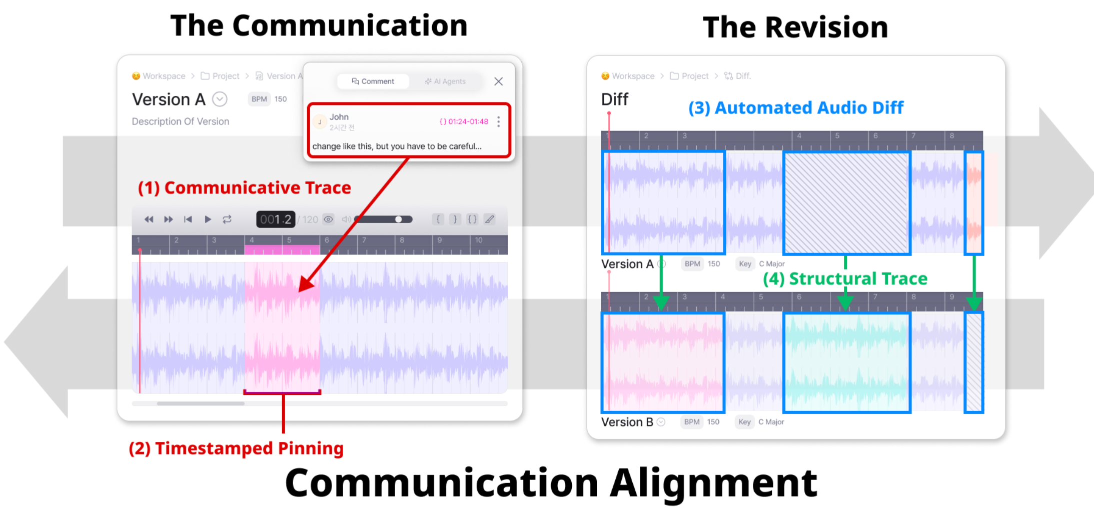

Fig. 1. The TRAX Workflow. TRAX visualizes the opaque trajectories of asynchronous music collaboration. By coupling *Timestamped Pinning* (left) to capture communicative traces with *Automated Audio Diff* (right) to reveal structural changes, the system explicitly aligns contextual feedback with structural traces.

This paper introduces TRAX, a web-based environment designed to visualize the opaque trajectories of asynchronous music production. While creativity involves exploring design spaces, capturing such trajectories in audio is difficult because opaque binaries decouple communicative traces from its spatial context and structural history. This disconnection often forces collaborators to manually re-listen to tracks to identify edits, fracturing the creative process. TRAX bridges these gaps by coupling *Timestamped Pinning* to anchor feedback to the waveform with *Automated Audio Diff* to visualize structural revisions on the timeline. By linking communicative traces with structural traces, this paper demonstrates how TRAX establishes a structured ground truth to analyze the dynamics of creative collaboration.

CCS Concepts: • **Human-centered computing → Empirical studies in HCI**.

*These authors contributed equally to this work.
†Corresponding author.

Authors' Contact Information: Jaywoong Jeong, TRAX, Republic of Korea and KAIST, Daejeon, Republic of Korea, jaywoong.jeong@kaist.ac.kr.

Additional Key Words and Phrases: Music Collaboration, Trace Analysis, Creativity Support Tools, Audio Diff

1 Introduction

Creativity is inherently an iterative process of exploring vast design spaces, requiring tools that support rapid experimentation and discovery [13, 15]. To deeply understand this exploration, recent research emphasizes capturing the "trajectories" of creative work—tracking the sequence of intermediate states and the discrete "design moves" that connect them [1]. In text-based domains like software engineering, version control systems naturally preserve these trajectories, allowing collaborators to transparently trace how specific intents evolved into concrete executions [4, 18].

However, preserving such trajectories in asynchronous music collaboration presents a unique challenge. Unlike code, audio is a continuous, time-based medium typically shared as opaque binaries, making it difficult to parse the continuous signal into discrete, legible changes [2]. Consequently, the creative trajectory is fractured by critical information gaps. **Communicative traces**, including musical discussion and contextual feedback, are often scattered across external channels such as email or instant messaging, completely detached from the Digital Audio Workstation (DAW) [10, 11], the central environment where structural audio edits manifest as **structural traces**. This separation strips communicative intent of its spatial context ("Where?"), while the lack of structural visualization forces collaborators to manually re-listen to the track to detect edits ("What changed?") [12, 14]. Ultimately, this makes it difficult to map vague aesthetic requests to their concrete parameter adjustments ("How/Why?") [5].

To bridge these gaps, we introduce Trax, a system designed to visualize the opaque trajectories of music collaboration. While existing tools range from real-time jamming [7] to asynchronous repositories [3], they fail to integrate communicative intent with structural history. Trax restores the creative trajectory by coupling two mechanisms: *Timestamped Pinning*, which creates strict spatial context for feedback [9]; and *Automated Audio Diff*, which visualizes structural changes on the timeline [6, 16]. By systematically aligning strictly located feedback with structural traces, Trax establishes a ground truth for analyzing the semantics of creative collaboration.

2 System Description

Trax is a web-based environment designed to streamline asynchronous music collaboration. By integrating audio playback, feedback mechanisms, and version control, it addresses the disconnection between communication and the audio artifact. The system's core workflow consists of two complementary mechanisms: establishing *spatial context* for feedback through timestamped pinning, and revealing *structural changes* through automated audio diffing.

2.1 Capturing Communicative Traces

In typical music collaboration workflows, feedback is exchanged through external channels such as email or instant messengers, divorced from the audio artifact itself. Even when feedback references a specific moment in a track (e.g., "the transition around one minute in"), such temporal references remain imprecise and easily lost as conversations grow.

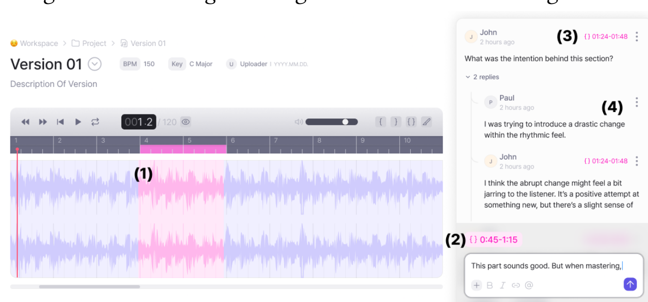

Fig. 2. The *Timestamped Pinning* Interface. (1) A collaborator selects a time region on the waveform, then (2) composes feedback anchored to that region. (3) Posted pins are displayed with their associated time range. (4) Each pin supports threaded replies for contextual discussion.

Trax addresses this by allowing collaborators to leave feedback as Pins directly anchored to the waveform (Figure 2). When composing a comment, the user first selects a region on the timeline, and the system automatically binds the feedback to that temporal context. Pins support two anchoring modes: a *point pin*, which marks a single timestamp (e.g., 01:32), and a *range pin*, which spans a time region (e.g., 0:45–1:15). Comments can also be left without a timestamp for general, non-temporal feedback. Each pin supports threaded replies, enabling contextual discussion around a specific piece of feedback while preserving the link to the original temporal anchor.

From a trace perspective, each pin constitutes a structured data record containing the author, timestamp of creation, anchoring region, rich-text content (including @-mentions to direct feedback at specific collaborators), and optional file attachments (e.g., reference tracks). As these records accumulate across successive versions of a track, they form a *version-linked communicative trace*—a structured history of feedback that can be systematically analyzed. This structured capture is what enables the alignment analysis between *communicative traces* and *structural traces* discussed in Section 3.

2.2 Capturing Structural Traces

To address the opacity of traditional version control where collaborators must mentally parse changes from monolithic files, Trax employs an *Automated Audio Diff* algorithm to decompose batch updates (e.g., a new mix file) into discrete changes.

First, the algorithm aligns versions *A* and *B* on a common timeline. Second, to ensure granularity, the tracks are segmented into atomic units (e.g., bar-length sections derived from BPM). Third, by comparing these corresponding units, the system untangles the bundled update, labeling specific differences as *deletion* or *insertion* events.

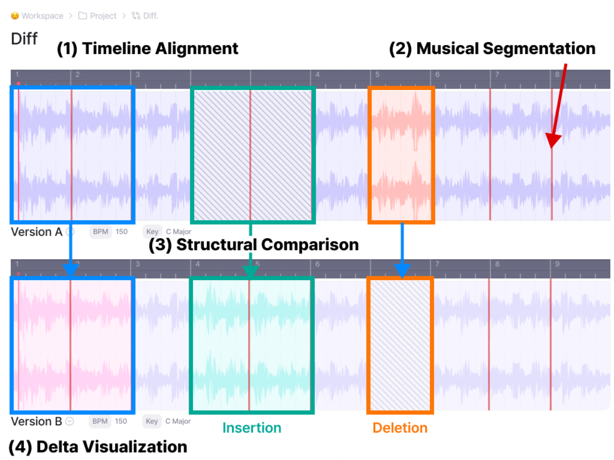

Fig. 3. The *Automated Audio Diff* process. The workflow proceeds through (1) timeline alignment, (2) BPM-based musical segmentation, (3) structural comparison, and (4) delta visualization. Green and orange blocks represent *Insertions* and *Deletions*, respectively. By abstracting opaque audio updates into discrete visual markers, the interface provides a legible history of structural traces embedded directly on the timeline.

Finally, as illustrated in Figure 3, these changes are visualized directly on the timeline. Adapting visualization strategies from software engineering, Trax aims to mitigate *information overload* by presenting only the relevant delta rather than the full file history [18]. However, unlike code where semantic changes are syntactically clear, audio changes can be ambiguous. By abstracting these changes into discrete visual blocks, we provide a *legible history* embedded within the artifact itself, echoing Hill and Hollan's concept of history-enriched digital objects [8]. It is important to note that this current implementation focuses on *arrangement-level* structural changes; detecting *timbral* nuances (e.g., EQ or compression adjustments) requires advanced signal processing and remains future work. This structural logging creates a granular foundation for mapping communicative intent.

3 Analytical Possibilities

By systematically linking communicative traces (via *Timestamped Pinning*) and structural traces (via *Automated Audio Diff*), TRAX converts the opaque processes of music production into structured data. We propose three primary directions for future research utilizing this framework.

3.1 Quantifying Communication Alignment

A persistent challenge in creative collaboration is bridging the gap between abstract requests and the resulting technical adjustments [5]. TRAX enables the measurement of "communication alignment" by correlating the spatial location of feedback pins with the timestamps of subsequent audio edits. For example, if a pin at 0:45 is followed by a spectral change in the same region in the next version, it suggests a direct translation of the request into action. Conversely, a spatial disconnect between the feedback and the edit may indicate a misunderstanding or a divergent interpretation of the critique [11]. This metric allows researchers to objectively assess how different types of instructions (e.g., affective vs. technical) influence the accuracy of revisions.

3.2 Visualizing Creative Trajectories

Beyond individual revisions, TRAX can visualize the broader patterns of the collaborative process over time. Drawing on concepts of design trajectories and "edit wear" [1, 9], aggregated diff logs reveal the team's movement through the design space. High densities of edits in specific regions may indicate phases of *convergent refinement*, where collaborators polish a fixed idea. In contrast, scattered edits across the timeline may signal *divergent exploration*, where the team experiments with multiple possibilities. Analyzing these patterns provides insight into the iterative nature of audio production, distinguishing between productive exploration and lack of focus.

3.3 Ground Truth for AI-Assisted Editing

Finally, the dataset generated by TRAX serves as a foundational ground truth for AI creativity support tools. While current generative models excel at creating new assets, they often lack the context to support precise, instruction-based editing [17]. The paired data of *[Communicative Traces]* → *[Structural Traces]* captures the nuanced relationship between human intent and machine manipulation. This data is crucial for training models that respond to specific editing commands (e.g., interpreting "less muddy" as specific EQ cuts) rather than generating generic outputs, potentially mitigating the homogenization of AI-generated content [1].

4 Conclusion

We presented TRAX as a system to surface and analyze the opaque processes of music collaboration. By linking versioned communicative traces with structural traces, TRAX enables an objective analysis of collaborative trajectories in time-based media. We hope this framework contributes to the discussion on expanding the design space and theoretical foundations of trace analysis tools.

References

[1] Barrett R Anderson, Jash Hemant Shah, and Max Kreminski. 2024. Evaluating creativity support tools via homogenization analysis. In *Extended Abstracts of the CHI Conference on Human Factors in Computing Systems*. ACM, New York, NY, USA, 1–7.

[2] Tina Blaine and Sidney S Fels. 2003. Contexts of collaborative musical experiences. Zenodo.

[3] Fabio Calefato, Giuseppe Iaffaldano, and Filippo Lanubile. 2018. Collaboration Success Factors in an Online Music Community. In *Proceedings of the 2018 ACM International Conference on Supporting Group Work* (Sanibel Island, Florida, USA) *(GROUP '18)*. Association for Computing Machinery, New York, NY, USA, 61–70. doi:10.1145/3148330.3148346

[4] Laura Dabbish, Colleen Stuart, Jason Tsay, and Jim Herbsleb. 2012. Social coding in GitHub: transparency and collaboration in an open software repository. In *Proceedings of the ACM 2012 conference on Computer Supported Cooperative Work*. ACM, New York, NY, USA.

[5] Thomas Deacon, Patrick Healey, and Mathieu Barthet. 2022. "It's cleaner, definitely": Collaborative Process in Audio Production. *Comput. Supported Coop. Work* 32, 3 (Nov. 2022), 475–505. doi:10.1007/s10606-022-09448-1

[6] Paul Grünbacher and Markus Neuwirth. 2024. Towards Feature-based Versioning for Musicological Research. In *Proceedings of the 18th International Working Conference on Variability Modelling of Software-Intensive Systems* (Bern, Switzerland) *(VaMoS '24)*. Association for Computing Machinery, New York, NY, USA, 77–82. doi:10.1145/3634713.3634723

[7] Michael Gurevich. 2006. JamSpace: a networked real-time collaborative music environment. In *CHI '06 Extended Abstracts on Human Factors in Computing Systems* (Montréal, Québec, Canada) *(CHI EA '06)*. Association for Computing Machinery, New York, NY, USA, 821–826. doi:10.1145/1125451.1125613

[8] William C. Hill and James D. Hollan. 1994. History-enriched digital objects: Prototypes and policy issues. *The Information Society* 10, 2 (1994), 139–145. arXiv:https://doi.org/10.1080/01972243.1994.9960164 doi:10.1080/01972243.1994.9960164

[9] William C. Hill, James D. Hollan, Dave Wroblewski, and Tim McCandless. 1992. Edit Wear and Read Wear. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '92)*. 3–9. doi:10.1145/142750.142751

[10] Ian Jeffery, Ethan Kim, Afroza Sultana, May Yu, David Oliva Wiebe, Jack Morrissey, and Ali Mazalek. 2025. SnapTunes: An Introductory Cross-device Application for Collaborative Music Production. In *Adjunct Proceedings of the 27th International Conference on Mobile Human-Computer Interaction (MobileHCI '25 Adjunct)*. Association for Computing Machinery, New York, NY, USA, Article 12, 7 pages. doi:10.1145/3737821.3749554

[11] Lily E Montague and Mathieu Barthet. 2023. Collaboration on the Tracks: Ethnographically-Informed Design for Computer-Assisted Music Collaboration between Producers and Performers. In *Proceedings of the 15th Conference on Creativity and Cognition* (Virtual Event, USA) *(CC '23)*. Association for Computing Machinery, New York, NY, USA, 391–392. doi:10.1145/3591196.3596619

[12] Andreas Schreiber, Lynn von Kurnatowski, Annika Meinecke, and Claas de Boer. 2024. Visualization ofnbsp;Software Development Provenance. In *Human Interface and the Management of Information: Thematic Area, HIMI 2024, Held as Part of the 26th HCI International Conference, HCII 2024, Washington, DC, USA, June 29–July 4, 2024, Proceedings, Part II* (Washington DC, USA). Springer-Verlag, Berlin, Heidelberg, 121–139. doi:10.1007/978-3-031-60114-9_10

[13] Ben Shneiderman. 2007. Creativity Support Tools: Accelerating Discovery and Innovation. *Commun. ACM* 50, 12 (2007), 20–32. doi:10.1145/1323688.1323689

[14] Sarah Sterman, Molly Jane Nicholas, and Eric Paulos. 2022. Towards Creative Version Control. *Proc. ACM Hum.-Comput. Interact.* 6, CSCW2, Article 336 (Nov. 2022), 25 pages. doi:10.1145/3555756

[15] Michael Terry and Elizabeth D Mynatt. 2002. Recognizing creative needs in user interface design. In *Proceedings of the fourth conference on Creativity & cognition - C&C '02*. ACM Press, New York, New York, USA.

[16] Dakuo Wang, Judith S. Olson, Jingwen Zhang, Trung Nguyen, and Gary M. Olson. 2015. DocuViz: Visualizing Collaborative Writing. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems* (Seoul, Republic of Korea) *(CHI '15)*. Association for Computing Machinery, New York, NY, USA, 1865–1874. doi:10.1145/2702123.2702517

[17] Zhen Wu, Serkan Kumyol, Shing Yin Wong, Xiaozhu Hu, Xin Tong, and Tristan Braud. 2025. Orchid: A creative approach for authoring LLM-driven interactive narratives. In *Proceedings of the 2025 Conference on Creativity and Cognition*. ACM, New York, NY, USA, 774–791.

[18] YoungSeok Yoon, Brad Myers, and Sebon Koo. 2013. Visualization of fine-grained code change history. In *Proceedings of IEEE Symposium on Visual Languages and Human-Centric Computing, VL/HCC*. 119–126. doi:10.1109/VLHCC.2013.6645254