AGENT SESSION INSIGHTS

Building Agent Session Insights: player, reviewer, and repository

This is the GitHub Copilot CLI conversation that built Agent Session Insights itself: a reusable reviewer, an interactive player, and a private GitHub repository. The scope starts with the sessionreviewer design request and ends with the completed repository push, excluding earlier prompt recovery and later Pages setup. It includes 12 typed prompts and three genuine form answers. User direction determines the multi-harness scope, fixed Venn theme, navigation, accounting, information hierarchy, reproducible example, privacy and project name. Historical replies mentioning 22 prompts, 176 requests or $12.587361 describe the earlier Codex fixture used during development, not this Copilot session's measured usage. Historical delivery and test claims remain agent-reported; this run independently verifies the newly generated report and player.

Open standalone session player

15	21	120	216	$5.67
User inputs	Assistant messages	Metered requests	Tool invocations	Estimated retail cost (USD)

Wall time: **3h 20m 15s** Total turn time: **1h 12m 48s (partial)**

11 of 15 turns timed; overlaps summed; not active compute time.

Recorded period: 2026-11-12 08:45:47 PST - 2026-11-12 12:06:03 PST

Scope and coverage

This report covers only the explicitly selected interaction. Logistics and approvals remain in the full replay; reviewed material directions are marked separately. Missing harnesses are outside the measured population.

Harness	Coverage	Inputs	Metered requests
GitHub Copilot	Supplied	15	120
Claude Code	No source supplied; not part of this selected Copilot CLI interaction.	Not measured	Not measured
Codex	Historical Codex logs were development inputs, not this example's conversation.	Not measured	Not measured
Cursor	No export supplied; not part of this selected interaction.	Not measured	Not measured
Replit	No export supplied; not part of this selected interaction.	Not measured	Not measured

▼ **Session details and warnings (1 sources, 1 warnings)**

Session start	Source / session	Selected bounds	Raw records	Outside scope
2026-11-12 08:45 PST (first recorded event)	S01 · 53fca206-b87f-4fad-bd90-f09c26d61fac	2026-11-12T16:45:47.965Z to 2026-11-12T20:06:03.385Z	372	0

S01: Event-log-only import: SQL-only and unrecorded auxiliary requests are not reconstructed.

Conversations: 1. Events without reliable timestamps: 0. Visible message records are separate from metered requests.

Observed usage and model mix

Inclusive input	Output	Cached reads (inside input)	Cache creation (inside input)
18,500,781	166,272	17,603,808	896,613

Estimated retail API-equivalent total: $5.67 USD. 120 of 120 observed requests priced.

Model	Requests	Input tokens	Output tokens	Total tokens	Token share	Estimated cost (USD)	Cost share
gpt-6.1-sol	120	18,500,781	166,272	**18,667,053**	100.0%	**$5.67**	100.0%

Input includes cached reads and writes; total tokens = input + output. Token shares require complete, nonzero totals. Cost shares require nonzero totals without ranges. Different workloads and cache use make this a usage comparison, not a model-efficiency benchmark.



▶ **Cost breakdown by model**

▼ **Spend and tokens by user turn (15)**

Turn	Prompt	Models	Calls	Input tokens	Output tokens	Estimated USD
H001	create a subdirectory, sessionreviewer. Then come up wiht a	gpt-6.1-sol	29	4,223,265	52,659	$1.38
H002	{ "framework": "Use the existing Venn report/player design	Unknown	0	Unknown	Unknown	Unknown (no recorded usage)
H003	next and back need to be arrow button on the right. Zoom sho	gpt-6.1-sol	15	3,013,615	17,083	$0.98
H004	is the "venn design" dynamically generated based on the sess	gpt-6.1-sol	23	2,294,841	36,637	$1.34
H005	review the UX to streamline. Too much space on zoomed vs tot	gpt-6.1-sol	14	1,875,459	18,735	$0.70
H006	okay, create a git repo for this, make sure that the readme	gpt-6.1-sol	15	2,462,988	25,831	$0.58
H007	{ "paper": "a paper generated for this session (it's a met	Unknown	0	Unknown	Unknown	Unknown (no recorded usage)
H008	go anead and push	gpt-6.1-sol	2	356,653	814	$0.05
H009	{ "repository": "wait, suggest a good name for this. give	Unknown	0	Unknown	Unknown	Unknown (no recorded usage)

Turn	Prompt	Models	Calls	Input tokens	Output tokens	Estimated USD
H010	suggest a name for the project. give me 3 names	gpt-6.1-sol	1	178,984	68	$0.02
H011	look to see if anyone else is using that name	gpt-6.1-sol	3	542,914	799	$0.07
H012	come up with 3 more, after you've done conflict searches. If	gpt-6.1-sol	3	555,090	2,164	$0.09
H013	come up wiht a few more. I like "session", "analysis", "ins	gpt-6.1-sol	2	377,876	1,444	$0.06
H014	how about AgentSessionInsights?	gpt-6.1-sol	2	381,061	254	$0.04
H015	okay, proceed	gpt-6.1-sol	11	2,238,035	9,784	$0.36

Unattributed: 0 calls; Unknown (no recorded usage).

Observed activity (not measured working time)

Bin size (minutes)	User occupied bins	Model occupied bins	Shared bins	Model bins without an input
5	12	22	12	10
10	11	14	11	3
15	10	12	10	2

Clock timezone: America/Los_Angeles. Different bin widths show sensitivity; no claim of continuous attention, runtime, or unattended work follows from a bin.

▼ **Activity by hour**

Local hour	User inputs	Model events
2026-11-12 08:00 PST	1	1
2026-11-12 09:00 PST	2	30
2026-11-12 10:00 PST	4	53
2026-11-12 11:00 PST	6	23
2026-11-12 12:00 PST	2	13

How the user shaped the work

Reviewed interpretation, with verbatim input and response evidence. Responses establish what the historical agent reported, not independent proof of delivery.

Turn a one-off replay into a reusable reviewer

Architecture and scope · H001 · S01:1

> create a subdirectory, sessionreviewer. Then come up wiht a design for an agent that includes skills and prompts to gather copilot, claude, codex, replit, and cursor logs, perform the analysis, create a report html and pdf with a standalone player with the design framework. And add some things to th eplayer: a forward back buttons by the session text view, zoom abaility and time on the timeline axis. Then run it on the same creation playback logs so that I can see how it comes out. Put the generated report/player in a temp directory under the root of this project

The user requests a standalone sessionreviewer directory, agent skills and prompts, collection across five harnesses, analysis, an HTML/PDF report and a standalone player. The response reports implementing the package and testing it on the earlier creation-session fixture. The requested product is a report of agent work, not merely a raw trace viewer.

Agent-reported response: R004 · S01:90

> Created [sessionreviewer](sessionreviewer/README.md) with the agent design, three skills, prompts, and runnable collection/analysis pipeline.
>
> Generated in `tmp/sessionreviewer-creation/`:
> - [HTML report](tmp/sessionreviewer-creation/report.html)
> - [PDF report](tmp/sessionreviewer-creation/report.pdf)
> - [Standalone player](tmp/sessionreviewer-creation/player.html)
>
> The Venn-style player adds adjacent back/forward message buttons, timeline zoom/pan, and clock-axis labels. All **22 prompts, 58 responses, and 176 model events** reconcile with the original replay. Twelve tests plus browser/PDF checks passed.
>
> Cursor and Replit require supplied conversation exports; no real exports from those products were available for validation.

Preserve the existing Venn visual framework

Explicit design choice · H002 · S01:4

> {
> "framework": "Use the existing Venn report/player design"
> }

This recorded form answer selects the existing Venn report/player design. The assistant confirms it will preserve that framework. The theme is fixed; the later clarification distinguishes log-driven content and metrics from dynamically generated visual design.

Agent-reported response: R001 · S01:12

> I'll build `sessionreviewer` as a reusable agent package with collection skills, analysis prompts, and an executable report pipeline. I'll preserve the Venn design, add transcript navigation, timeline zoom, and time-axis labels, then generate and verify a sample under `tmp/sessionreviewer-creation/`.

Use direct manipulation for transcript and timeline navigation

Player interaction · H003 · S01:91

> next and back need to be arrow button on the right. Zoom should be interactive click and drag and have scrollbar

The user specifies arrow buttons on the right, click-and-drag zoom and a scrollbar. This replaces a vague zoom requirement with concrete interaction behavior. The assistant reports right-aligned arrows, drag selection and scroll-based panning; the new publishing checks independently exercise those controls.

Agent-reported response: R006 · S01:132

> Updated the [player](tmp/sessionreviewer-creation/player.html): right-aligned **← →** message buttons, **drag-to-zoom**, and a horizontal scrollbar for panning.
>
> Regenerated the report/PDF. All 14 tests and browser checks passed.

Make model usage and resource costs inspectable

Accounting and transparency · H004 · S01:133

> is the "venn design" dynamically generated based on the session logs. Player should include model uses, estimated total spend, spend in the zoomed area, and spend by turn, including tokens in/out.

The user asks for models, estimated session spend, spend within the zoomed region, and per-turn input/output tokens and cost. The response distinguishes the fixed theme from computed content and reports adding cache-aware historical API-equivalent estimates. The dollar amount quoted in that historical response belongs to the old Codex test fixture; this report computes a separate estimate for the selected Copilot requests.

Agent-reported response: R009 · S01:208

> The Venn theme is fixed; its content, timeline, and metrics are log-driven.
>
> Updated the [player](tmp/sessionreviewer-creation/player.html), [report](tmp/sessionreviewer-creation/report.html), and [PDF](tmp/sessionreviewer-creation/report.pdf) with model usage, tokens in/out, total and zoom-window spend, and per-turn estimates.
>
> Creation session: **$12.59 estimated API-equivalent cost**, not an actual bill. **22 tests passed**, plus offline browser/PDF checks.

Attach costs to prompts instead of duplicating summary cards

Information hierarchy · H005 · S01:209

> review the UX to streamline. Too much space on zoomed vs total. Total should be in one place. And cost for session text should show model and cost input/output tokens attached to the user prompt, not at the top. Costs should also be showin on the fill session log

The user rejects the space used by competing full-session and zoom summaries. The requested hierarchy puts the total in one place and attaches model, input/output tokens and cost to each user prompt, including the full log. The response reports a compact totals strip and prompt footers in the player and complete HTML/PDF record.

Agent-reported response: R011 · S01:252

> Updated the [player](tmp/sessionreviewer-creation/player.html): one compact session total, with a range subtotal only when zoomed or filtered. Model, estimated cost, and input/output tokens now appear beneath user prompts and in the full session log.
>
> HTML/PDF reports refreshed. **24 tests passed**, plus desktop/mobile, offline, and PDF checks.

Package a reproducible repository with an embedded example

Reproducibility · H006 · S01:253

> okay, create a git repo for this, make sure that the readme describes how to run it with this specific session log as the example with the embedded player and a link ot the paper in the repo for it.

The user requests a Git repository, a README explaining how to run this session, an embedded player and a paper link. The assistant reports a local repository and a clean-checkout reproduction. The bundled example initially used the earlier Codex development fixture; this corrected example now contains the Copilot conversation that built the package and repository.

Agent-reported response: R014 · S01:299

> Created the local Git repo in `sessionreviewer/`, branch `main`, commit `aaede7d`. Nothing pushed.
>
> The [README](sessionreviewer/README.md) documents setup and running this exact bundled session, with links to the [meta-report paper](sessionreviewer/examples/creation/report.pdf) and [HTML report with embedded player](sessionreviewer/examples/creation/report.html).
>
> **25 tests passed**, and the documented workflow reproduced successfully from a clean checkout.

Make the paper a meta-report about the project

Report framing · H007 · S01:269

> {
> "paper": "a paper generated for this session (it's a meta report for the project)"
> }

The form answer explicitly asks for a paper generated for this session, a meta-report for the project, rather than the original cross-harness study. The response agrees to include a creation-session paper alongside the HTML report and embedded replay. This clarification establishes the intended subject of the example.

Agent-reported response: R012 · S01:271

> I'll include the generated creation-session meta-report as the paper, alongside its HTML version with the embedded replay. I'll also make the example runnable without files outside the new repository, using a scoped, redacted log export and the frozen rate card.

Keep the repository private and resolve the name before pushing

Publication control · H009 · S01:305

> {
> "repository": "wait, suggest a good name for this. give me 3 options",
> "visibility": "private"
> }

The recorded form answer selects private visibility and pauses repository naming for suggestions. The subsequent conversation explores alternatives before the final approval. The final response reports a push to the private agent-session-insights repository. This is a genuine user decision captured through a form, not a synthetic harness instruction.

Agent-reported response: R021 · S01:372

> Pushed to **[markrussinovich/agent-session-insights](https://github.com/markrussinovich/agent-session-insights)** as a **private** repository.
>
> Branding, README, example paper, and embedded player are updated. **25 tests passed**; remote `main` verified at `44a06f0`.

Set a practical threshold for naming conflicts

Decision criterion · H012 · S01:318

> come up with 3 more, after you've done conflict searches. If the conflict has < 100 stars then it's not a conflict

The user requires conflict searches but excludes projects with fewer than 100 stars from disqualifying a name. The assistant applies that criterion to further suggestions. Search results and star counts are historical agent reports; this replay does not claim that they remain current.

Agent-reported response: R017 · S01:329

> Checked GitHub names and hyphenated variants. **None has a conflict meeting your ≥100-star cutoff.**
>
> | Name | Why it fits | Search result |
> |---|---|---|
> | **SteerTrace** | Traces human direction, agent decisions, and outcomes | No exact or hyphenated match found |
> | **TurnAtlas** | Explores sessions through turns, timelines, and costs | No exact or hyphenated match found |
> | **AgentChronicle** | Turns agent histories into reports and replays | Existing [`agent-chronicle`] (https://github.com/xls2020/agent-chronicle) has **2 stars**—below your cutoff |
>
> My pick: **SteerTrace**—distinctive and emphasizes human steering rather than just playback.

Name the analysis product rather than the underlying trace

Product positioning · H013 · S01:330

> come up wiht a few more. I like "session", "analysis", "insights", "summary", etc. type names to more directly capture that this isn't a trace, its a report of a trace

The user asks for session, analysis, insights and summary terminology, explicitly distinguishing a report of a trace from the trace itself. The response shifts the proposed names toward readable reports and findings. This reframes the product rather than merely choosing a label.

Agent-reported response: R018 · S01:337

> These emphasize the report rather than the underlying trace. **No ≥100-star GitHub conflicts found**, including hyphenated variants.
>
> | Name | Emphasis |
> |---|---|
> | **SessionBrief** | A readable account of what happened and why |
> | **SessionInsights** | Analysis of decisions, outcomes, and costs |
> | **SessionDigest** | Distilling lengthy histories into meaningful findings |
> | **SessionReport** | The most direct description of the output |
>
> My pick: **SessionBrief**—concise, memorable, and clearly suggests a synthesized report rather than a logging tool.

Choose Agent Session Insights and authorize the branded repository

Naming and delivery · H014 · S01:338

> how about AgentSessionInsights?

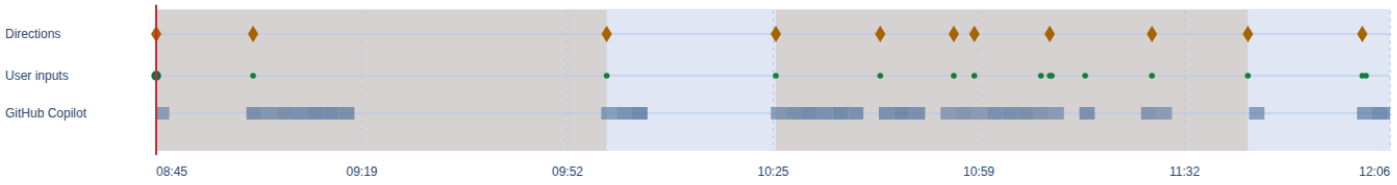
The user proposes AgentSessionInsights and then approves proceeding in H015. The assistant agrees to the display name Agent Session Insights and reports rebranding the paper/player and pushing the private repository. The selected interval ends with that delivery response, before the later GitHub Pages discussion.

Agent-reported response: R021 · S01:372

> Pushed to **[markrussinovich/agent-session-insights](https://github.com/markrussinovich/agent-session-insights)** as a **private** repository.
>
> Branding, README, example paper, and embedded player are updated. **25 tests passed**; remote `main` verified at `44a06f0`.

Interactive session playback

Open the standalone player. Use the arrow buttons on the right of the session text to move between user prompts and follow their timeline positions. Drag across the timeline to zoom and use its scrollbar to pan. Search the separate full session log below. Axis labels include clock time.



Full session log

Select a message to navigate Replay above. The standalone player also includes the full conversation.

Each user prompt includes its complete recorded turn's model, estimated cost and tokens. Agent rows do not repeat the charge.

Data

Sessions · Analysis · Messages and usage · Rates · Verification