AI AUGMENTATION CASE STUDY

Converting 200+ Hours of Live Content

Into a Structured Creative IP Pipeline

Darkeport Productions — A Real-World Implementation

83 Episodes | 9 Story Arcs | 3 Years of Content | 401 Output Files

Presented by: Corey Segall, ChangeEnable

Executive Summary

This case study documents the transformation of 200+ hours of recorded actual-play tabletop RPG content — spanning 83 episodes over three years of live-streamed production — into a comprehensive, structured creative intellectual property system. The project demonstrates how AI augmentation, combined with disciplined human-in-the-loop validation, can convert unstructured multimedia into production-ready assets at a fraction of the time and cost of traditional manual processes.

The client, Darkeport Productions, produces a serialized D&D campaign streamed live on YouTube. The content represents a rich, complex fictional world with 11 player characters, 58 named NPCs, 11 deities, 14 significant locations, and thousands of narrative moments spread across three years of 3-hour live sessions. The challenge: none of this content existed in any organized, searchable, or reusable form.

Key Results

- **Processing time reduction:** 97% — from ~33 hours to ~1 hour per 11 episodes
- **Total output files produced:** 401 files across 44MB of structured content
- **Episodes fully processed:** 42 episodes locked and validated
- **Episode summaries written:** 45 individual narrative summaries (1/2 to full page each)
- **Total API cost:** $130.60 total — $100/mo Claude Max + $30.60 AssemblyAI (covered by free credit)
- **Lore corrections applied:** 567 retroactive corrections across 165 files
- **Interactive wiki:** 82 episode pages, 36 full-text journals, 57 lore entries
- **Bible versions:** 7 major versions, each building on the last

1. The Challenge

Live actual-play content presents unique challenges for post-production organization. Unlike scripted media, the content is improvisational, features multiple simultaneous speakers, blends in-character dialogue with out-of-character table talk, and generates lore organically across hundreds of hours with no centralized reference document.

Specific Pain Points

Speaker Attribution: YouTube auto-captions achieved only ~38% speaker attribution. With 6-8 players plus a GM, 62% of all dialogue was unattributed, requiring manual voice-matching — the single largest time sink.

Lore Mangling: Auto-transcription consistently destroyed proper nouns. 'Juramentum' became 'Gerimentum,' 'Kithgi' became 'Kithkey,' 'Quaylithon' became 'Queen Lathon,' and 'Merick' became 'America.' Every episode required dozens of manual corrections cross-referenced against accumulated knowledge.

Content/Meta Blending: Live streams intermix dramatic roleplay with dice rolls, sponsor reads, bathroom breaks, and audience chat. Separating signal from noise required listening to every minute.

No System of Record: Three years of world-building existed only in the memories of the cast and scattered viewer notes. No canonical reference existed for spelling, timeline, character relationships, or narrative continuity.

Scale: At 3 hours per episode across 83+ episodes, the corpus exceeds 200 hours of dense, multi-speaker audio. Manual processing at the old rate would require 200-300 hours of human labor — effectively prohibitive.

2. How We Did It — The Technical Pipeline

The pipeline chains four tools and technologies into a repeatable, self-improving workflow. Each step feeds the next, and human validation gates ensure quality at critical junctures. The following details the exact tools, techniques, and workflow used to transform raw YouTube recordings into structured creative assets.

STEP 1: Content Acquisition — YouTube to MP4

Tool: yt-dlp (https://github.com/yt-dlp/yt-dlp)

The open-source command-line tool yt-dlp was used to batch-download full episodes from Darkeport Productions' YouTube playlists. Yt-dlp supports playlist-level downloading, allowing entire seasons to be acquired in a single command. Episodes were downloaded as MP4 files preserving the original audio quality, which is critical for accurate speaker diarization downstream.

Why this matters: YouTube's own auto-caption system produces SRT files with only ~38% speaker attribution and heavily mangled proper nouns. By downloading the raw MP4 and processing the audio through a superior transcription engine, we bypassed YouTube's limitations entirely.

Playlist structure mapped directly to the series organization: the main Gold, Green, and Red campaign (54 episodes), When Dreams Bleed (11 episodes), GGR Volume 3 (9 episodes), Age of Tezcatli (6 parts), Tales of Terra (1 episode), and Mithra's Tale (2 episodes). Each playlist was downloaded as a batch and tagged by series for pipeline tracking.

STEP 2: Transcription + Speaker Diarization — MP4 to Attributed Text

Tool: AssemblyAI Universal-2 (https://www.assemblyai.com)

Each MP4 episode was submitted to AssemblyAI's Universal-2 transcription engine with speaker diarization enabled. AssemblyAI processes the audio and returns a timestamped transcript with every utterance assigned to a speaker label (Speaker A, Speaker B, etc.). For a typical 3-hour episode with 6-8 distinct voices, the engine reliably separated speakers and produced 1,200-1,900 attributed utterance blocks.

Cost: $0.17 per hour of audio, or approximately $0.51 per 3-hour episode. The total cost for 60 episodes was $130.60 ($100 Claude Max + $30.60 AssemblyAI) — entirely covered by AssemblyAI's $50 free credit tier.

The raw output is a JSON transcript with timestamps, speaker labels, and text. This becomes the input for the next stage. The diarization is not perfect — speakers occasionally bleed between labels (especially when two players have similar vocal ranges), and some episodes split a single speaker across two labels. These issues are caught and resolved in the human validation step.

STEP 3: Speaker Mapping Validation — Human-in-the-Loop Gate

Tool: Human Validator (Show Creator)

Before any AI processing occurs, the speaker labels must be mapped to actual characters. Claude Opus analyzes the raw transcript and proposes a speaker mapping based on evidence: the first clearly identifying dialogue per speaker, utterance counts, vocal patterns, and contextual clues (e.g., a speaker discussing 'my bakery' is likely Weisa; a speaker referencing 'my grandfather' is likely Grip).

The proposed mapping is presented to the human validator (the show creator) with evidence lines for each speaker. The human confirms, corrects, or flags ambiguities. This step takes approximately 2 minutes per episode and is the most critical gate in the entire pipeline — every downstream deliverable depends on knowing who said what.

Key validation patterns discovered over 42 episodes:

Split speakers: When a single player's voice is split across two diarizer labels (e.g., Grip on labels B and F), the validator identifies the split and both labels are merged in processing.

Absent players: Not every player attends every session. The validator confirms which cast members are absent so the pipeline does not force-assign their character to an unrelated speaker.

NPC absorption: The GM voices NPCs, and sometimes a player's character is 'absorbed' into another speaker's label when they share scenes closely. The validator flags these cases.

Once validated, the speaker map is locked and tagged: 'Speaker map VALIDATED by user before processing.' No further changes occur without re-validation.

STEP 4: AI-Powered Transcript Processing — Raw to Animation-Ready

Tool: Claude Opus (Anthropic) (https://claude.ai)

With validated speaker mapping in hand, Claude Opus processes each episode's transcript through multiple parallel analysis passes, applying the show's world-setting standards, naming conventions, and narrative tone. This is the core transformation step — converting a raw timestamped transcript into a production-quality animation script.

4a. Content Classification

Every utterance is tagged with one of eight content categories:

IC-DIALOGUE: In-character speech between characters. IC-ACTION: Physical actions described by players in character. NARRATION: GM descriptions of scenes, environments, and NPC actions. LORE-EXPOSITION: World-building information delivered in narrative. OOC-TABLE: Out-of-character table talk (jokes, rule clarifications, snack runs). OOC-STREAM: Stream-specific content (sponsor reads, chat interaction, technical issues). MECHANIC: Game mechanics (dice rolls, stat checks, combat math). TRANSITIONAL: Scene transitions, break announcements.

The classifier started with a ~27% false positive rate on OOC detection and improved to under 5% through iterative learning. By the final episodes processed (GGR3 EP09), the false positive rate hit 0%.

4b. Spelling Normalization + Lore Correction

A canonical spelling dictionary — built and maintained across all sessions — is applied automatically to every transcript. The dictionary contains 200+ correction patterns covering proper nouns, location names, deity names, and world-specific terminology that AssemblyAI's general-purpose model consistently mishears.

Examples: Gerimentum/Gerimentom/Juramentom are all corrected to Juramentum. Kithkey/Kithki/Kiski become Kithgi. Tulloch/Toluc/Taleg/Tuluk become Tlaloc. America becomes Merick. Queen Lathon becomes Quaylithon.

When new canonical spellings are confirmed by the show creator (e.g., confirming 'Taweia' over 'Tawea/Tuea/Tuya'), corrections propagate retroactively across ALL previously processed files. The project executed 567 retroactive corrections across 165 files in a single automated pass.

4c. Narrative Extraction — Gold Nuggets

Claude identifies the most dramatically significant moments in each episode: journal entries (preserved verbatim), key plot revelations, character-defining dialogue, emotional peaks, and mechanical achievements. These are extracted into a separate Gold Nuggets file that serves as the curated highlight reel and the source material for episode summaries.

4d. Lore Discovery + Updates

New lore discovered during processing — characters, locations, relationships, revelations, game mechanics — is extracted and cross-referenced against the Master Lore Bible. New discoveries are flagged for Bible update. Contradictions with existing canon are flagged for human arbitration. Each episode's Lore Updates file documents what was found and how it integrates with existing knowledge.

STEP 5: OOC Validation — Teaching the Classifier

Tool: Claude + Human Validator

Every block classified as out-of-character is presented to the human validator for review. The validator confirms correct classifications and flags false positives — moments where the AI incorrectly tagged in-character content as OOC.

Each false positive becomes a learned exception that permanently improves future accuracy. Over 42 episodes, 20+ unique exception patterns were codified:

> 'The table' — when used to describe in-world furniture, not the game table. 'Thank you' — when spoken between characters, not players. 'The audience' — when the GM describes how NPCs in the scene react. 'Candle' — the in-world Candle of Tales, not the stream prop. 'Session' — when characters reference a court session, not a game session.

The validation step took approximately 5 minutes per episode at the start and trended toward under 2 minutes as the classifier matured. By the final processed episodes, there were typically only 2-3 blocks to review.

STEP 6: Master Lore Bible — Growing the World Setting

Tool: Claude Opus + Human Creative Authority

After each batch of episodes is processed, lore discoveries are organized into the Master Lore Bible — a comprehensive world-setting reference document maintained across the entire project. Claude structures the content; the show creator validates canonical decisions.

The Bible evolved through seven major versions (v1.0 through v7.0), growing from a simple correction dictionary into a 13-section, 108KB reference document covering:

> Characters (11 PCs with full bios + 58 NPCs with descriptions). Locations (14 significant places with history). Deities (11 gods with domains, servants, and relationships). Factions and organizations. Game mechanics and world rules. Key story events (EP01-EP54 + all specials). Journal entries and in-world documents. Complete correction dictionary (200+ patterns). Episode tracker with canonical titles. Version history documenting every update.

The Bible serves as the single source of truth for the entire project. Every other deliverable — wiki, summaries, animation scripts — draws from and feeds back into the Bible.

STEP 7: Interactive Wiki — Making Knowledge Accessible

Tool: Claude + HTML/JavaScript

Claude generates a single-file interactive HTML wiki that makes the accumulated knowledge browsable without any server or database. The wiki is rebuilt from the Bible with each major update and includes:

> Tabbed navigation across 7 sections: Home, Characters, Locations, Deities, Lore, Journals, and Episodes. Search and filter across all content. Arc-based episode filtering with color coding. Clickable character badges linking to detailed profiles. 82 episode pages — 53 with full detail views including narrative summary, characters present, and key lore reveals. 36 full-text journal entries with author color-coding. Custom Darkeport branding with transparent phoenix logo.

The wiki opens in any browser with zero dependencies — a creative team member can receive the single HTML file and immediately have access to the entire world setting.

STEP 8: Episode Summaries — Narrative Writing from Structured Data

Tool: Claude Opus (narrative writing mode)

Using the Gold Nuggets and Lore Updates as source material, Claude writes individual narrative summaries for each episode — not recaps but original prose that captures the emotional weight and dramatic arc of each session. Each summary follows a consistent format:

> Canonical YouTube title, arc designation, and episode number. Characters present (with player names). 1/2 to full page of narrative prose written to serve as both reference material and marketing content. Key Lore Reveals section listing the most important discoveries.

The summaries serve triple duty: reference for the production team, marketing teasers to attract new viewers (the wiki uses one-line versions as clickable teasers), and foundation material for a potential novelization or companion guide.

STEP 9: Retroactive Quality Assurance — The Correction Cascade

> **Tool: Automated scripting + Claude**

Whenever new canonical information is established — a confirmed spelling, a character's full name revealed 30 episodes after their introduction, a location's proper designation — corrections cascade backwards through every previously processed file.

The most significant retroactive pass in the project applied 567 corrections across 165 files in a single automated operation. This covered new canonical spellings (Taweia, Ateel, Tlaloc, La'Calla, Trisley) that were confirmed during GGR Volume 3 processing but needed to be applied to all When Dreams Bleed and earlier files.

This retroactive quality assurance is only possible because the pipeline maintains structured, machine-readable output. A stack of printed scripts could never be corrected this way. The digital-first approach means the entire corpus improves as knowledge grows.

3. Pipeline Architecture Summary

Step	Tool	Action	Human Role	Output
1	yt-dlp	Download MP4 from YouTube playlists	Select playlists	MP4 files
2	AssemblyAI	Transcribe + speaker diarize	None	JSON transcripts
3	Claude + Human	Propose + validate speaker map	**VALIDATE (~2 min)**	Locked map
4	Claude Opus	Classify, correct, extract, tag	None	5 deliverables
5	Claude + Human	OOC validation + exception learning	**VALIDATE (~3 min)**	Trained classifier
6	Claude + Human	Update Master Lore Bible	**ARBITRATE lore**	Bible vX.0
7	Claude + HTML	Generate interactive wiki	Review	Wiki HTML
8	Claude Opus	Write episode summaries	Review tone	Summary pages
9	Automated	Retroactive correction cascade	**Confirm spellings**	Updated corpus

Human involvement is concentrated at three gates (Steps 3, 5, 6) — approximately 5-7 minutes per episode. Everything else is automated. The pipeline is designed so that AI proposes and human validates — creative authority never leaves the creator's hands.

4. Deliverables Produced

The pipeline produces five deliverables per episode, plus four living reference documents that grow with every episode processed.

4.1 Per-Episode Deliverables (x42 episodes = 210+ files)

Animation-Ready Transcript

Every utterance timestamped, speaker-attributed, and content-tagged. Clean enough for direct use in animation production — voice actors can read from these scripts. Format: [HH:MM:SS] [TAG] | Speaker (Player) followed by indented dialogue text. Typical output: 800-1,900 usable blocks per episode. 42 completed episodes represent approximately 126 hours of production-ready dialogue.

Gold Nuggets

Curated selection of the most dramatically significant moments. Journal entries preserved verbatim. Key revelations, character-defining dialogue, and plot-critical events highlighted with timestamps. Source material for episode summaries and highlight reels.

OOC Validation Log

Every block flagged as out-of-character, with the classifier's reasoning and the human validator's decision. Both quality assurance and training data — each validation improves future accuracy.

Lore Updates

New lore discovered: characters, locations, relationships, revelations, mechanics. Cross-referenced against the Bible. Includes correction counts and retroactive fixes applied.

Speaker Map

Validated speaker-to-label mapping with evidence lines, utterance counts, and diarization anomaly notes.

4.2 Living Reference Documents

Master Lore Bible (v7.0 — 108KB)

13-section world-setting reference: characters, locations, deities, factions, mechanics, events, journals, corrections (200+ patterns), episode tracker with canonical titles for all 83 episodes, and version history. Seven major versions produced.

Interactive HTML Wiki (v7.0 — 298KB)

Single-file web application: 7 tabs, 11 PCs, 58 NPCs, 11 deities, 14 locations, 57 lore entries, 36 full-text journals, 82 episode pages (53 with full detail views). Searchable, filterable by arc, opens in any browser.

Episode Summaries (2 volumes, 45 pages)

Individual narrative summaries — one episode per page — with canonical title, characters present, narrative prose, and key lore reveals. Triple-duty: reference, marketing, novelization foundation.

Animation Scripts (42 episodes)

The Animation-Ready transcripts serve directly as production scripts. Speaker-attributed, emotionally contextualized, timestamped. 126 hours of animation-ready dialogue.

5. Efficiency Metrics

5.1 Head-to-Head

Metric	Old Process	AI Pipeline
Speaker Attribution	38%	100% (validated)
Time Per Episode	2.5-3.5 hours	~5 min active human work
Output Per Episode	~280 blocks	1,200-1,900 blocks (5 deliverables)
Lore Corrections	Manual	Auto (200+ patterns, retroactive)
Deliverables	1 (transcript)	5 per ep + 4 living docs
Self-Improving	No	Yes — compounds with every episode

5.2 Cost Analysis

	Old Process	AI Pipeline	Savings
Human Hours	200-300 hours	5-7 hours	195-293 hours
Calendar Time	6-9 months	3-4 weeks	5-8 months
API Cost	$0	$130.60	$130.60
Output Volume	~83 transcripts	401+ files (44MB)	5x more output
Net Value			**6-9 work-weeks saved**

The $130.60 total cost ($100 Claude Max + $30.60 AssemblyAI) is covered by AssemblyAI's $50 free credit tier. Effective monetary cost: zero.

6. Creative Value — Beyond Efficiency

6.1 Narrative Discovery

Processing every word across 200+ hours surfaced narrative threads the creators had forgotten. Character arcs developing subtly across 30+ episodes became visible. Thematic connections between the ancestral prequel (400 years before the main campaign) and the present-day storyline were mapped systematically for the first time.

6.2 Animation Pipeline

The 42 Animation-Ready scripts represent 126 hours of production-ready dialogue — speaker-attributed, emotionally tagged, timestamped. This volume would be prohibitively expensive to produce through traditional script-writing from source recordings.

6.3 Journal Preservation

36 in-character journal entries — often improvised live — were extracted, attributed, and preserved as permanent literary artifacts. Several (Grip's lamentation for Figulus, Weisa's 'my childhood ends now,' Marcus's sister Lucia) are powerful standalone creative works.

6.4 The Wiki as Onboarding Tool

A new creative team member can search any character, location, or concept and immediately access canonical information. Onboarding time drops from weeks of watching recordings to minutes of browsing.

7. Broader Applicability

While this case documents a creative production, the methodology applies to any domain with large volumes of multi-speaker recorded content:

- Podcasts and interview series: Index and cross-reference knowledge across hundreds of hours.
- Corporate meeting archives: Searchable knowledge bases with attributed quotes and decision tracking.
- Legal and medical transcription: Speaker attribution plus domain-specific terminology normalization.
- Educational content: Lecture series converted to structured curricula with topic indexing.
- Oral history and documentary: Archival interviews cross-referenced into thematic databases.

8. Conclusion

As Nate observed: "This isn't about reducing jobs — it's about enhancing the efficiency and output with the team you have." This project proves that principle. AI augmentation does not replace human creativity — it amplifies it. The AI processed the volume; the human ensured the quality. The AI proposed; the human decided. The AI remembered every word across 200+ hours; the human knew which words mattered.

The result is a comprehensive creative IP system — 401 output files, 44MB of structured content — that transforms ephemeral live performance into permanent, searchable, reusable creative assets. The world of Darkeport now exists not just in the memories of its creators but in a structured form that enables animated adaptations, novelizations, companion guides, and future campaigns built on organized canonical knowledge.

The pipeline cost $130.60 total ($100 Claude Max + $30.60 AssemblyAI) and saved approximately 200 hours of human labor. It produced deliverables that would not exist at all without AI augmentation. And it continues to improve with every episode processed.

The pipeline learns. The efficiency compounds. The creative output multiplies.

Appendix A: Tool Stack

Tool	Purpose	Reference
yt-dlp	YouTube playlist downloading	github.com/yt-dlp/yt-dlp
AssemblyAI	Audio transcription + speaker diarization	assemblyai.com ($0.17/hr)
Claude Opus	Content classification, lore extraction, narrative writing, wiki generation	claude.ai (Anthropic)
docx-js	Word document generation	npm package
HTML/JS	Interactive wiki (single-file, no server)	Custom-built

Appendix B: Complete Deliverable Inventory

- **Animation-Ready Transcripts:** 74 files
- **Gold Nuggets:** 74 files
- **OOC Validation Logs:** 72 files
- **Lore Updates:** 74 files
- **Master Lore Bible:** 7 versions (v1.0-v7.0), latest: 108KB
- **Interactive Wiki:** 4 versions (v4.0-v7.0), latest: 298KB
- **Episode Summaries:** 2 volumes, 45 individual pages
- **Pipeline Efficiency Reports:** 2 versions
- **Case Study:** This document
- **Total Output Files:** 401 files, 44MB
- **Canonical Episode Titles:** 83 episodes across 9 arcs