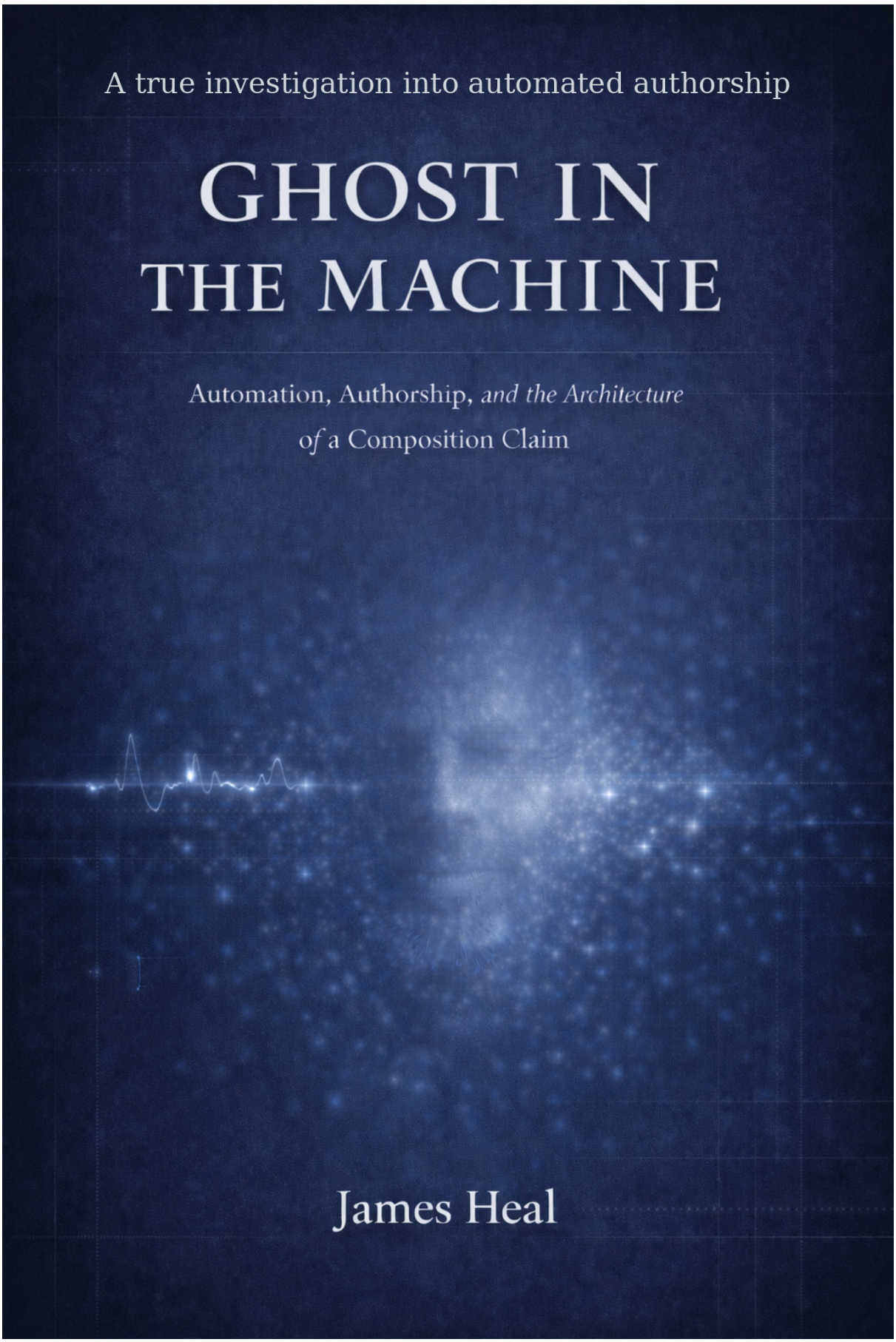

A true investigation into automated authorship
GHOST IN THE MACHINE
Automation, Authorship, and the Architecture of a Composition Claim
James Heal

# TABLE OF CONTENTS

# NOTE TO THE READER

This book examines a single composition claim that appeared on a video upload on 24 July 2017. The claim lasted thirty days. The effort required to understand the system that produced it lasted eight years.

The narrative that follows moves between personal chronology and technical explanation. At times it describes events as they occurred: the creation of a musical composition, the appearance of an automated copyright claim, and the correspondence and legal processes that followed. At other moments it pauses to explain the architecture of the systems involved.

Readers unfamiliar with digital copyright systems may find some terminology unfamiliar. Modern platforms such as YouTube operate automated rights management systems that compare uploaded audio against reference databases supplied by publishers, collecting societies, and rights administrators. When similarities are detected, the system may automatically apply a "claim" that affects how a video is monetised or distributed.

These systems do not evaluate authorship in the way a human listener would. They evaluate similarity between digital fingerprints of audio. Policy decisions attached to registered assets are then executed automatically.

Understanding that architecture is essential to understanding the events described in this book.

The account presented here is based on preserved documentation, timestamps, correspondence, and publicly available technical documentation describing how automated rights systems operate. Where legal proceedings are referenced, the descriptions reflect the procedural posture of the case at the time of writing.

The purpose of this book is not to argue that automated systems are malicious. It is to explain how they function, how registry data interacts with platform infrastructure, and how creators may encounter those systems when their work enters global distribution networks.

This book is an analytical account of those systems and events. Any discussion of system behaviour, data structure, or causation is presented as interpretation based on the documented material. Where judicial proceedings are referenced, the Court's findings are relied on only for what they determine: that there was an arguable case and that there was no reason to doubt authorship and copyright ownership. The book does not present its analysis as a finding made by the Court.

The events described are specific.

The architecture is not.

# GHOST IN THE MACHINE

*Automation, Authorship, and the Architecture of a Composition Claim*

James Heal

---

## Preface

This book was not written immediately after the events it describes.

It was written after understanding.

The composition claim that began on 24 July 2017 lasted thirty days. The effort required to understand the structure that produced it lasted eight years.

Time alters perspective.

The earliest questions concerned fairness.
Later questions concerned authorship.
Eventually the inquiry turned toward architecture.

By 20 October 2025, the Court stated that it had no reason to doubt that I wrote *Peace To The Self* and that I owned copyright in it, and accepted that there was an arguable case of copyright infringement. The procedural threshold had been met. The issues had narrowed.

That recognition was a matter of legal threshold, not a final determination of infringement. It confirmed authorship and ownership for the purposes of the proceedings, while leaving the underlying cause of the Content ID match unresolved.

At that point, the narrative no longer belonged solely to dispute.

It belonged to structure.

This book documents how automated rights systems operate at the intersection of registry, execution, and retention. It is not written in anger. It is written in literacy.

Digital platforms now mediate creative work at global scale. Every upload enters layered infrastructures of fingerprinting systems, publishing registries, monetisation engines, and compliance archives. These systems execute declared structure before they review challenge.

Understanding that ordering is no longer optional for creators.

It is structural.

The events described here are specific.

The architecture is not.

What began as a YouTube copyright notification became an eight-year investigation into the infrastructure that governs creative work online.

That investigation revealed a system in which registry precedes execution, automation precedes review, and the architecture of digital platforms shapes authorship at global scale.

# FOREWORD

## Authorship in the Age of Automation

In the twenty-first century, authorship no longer lives solely in notebooks, studios, or instruments.
It lives in databases.

Every uploaded song, video, photograph, or manuscript enters a layered infrastructure of registries, automated matching systems, monetisation engines, and compliance archives.

These systems operate at extraordinary scale.
They execute in milliseconds.
They process billions of transactions daily.
They do not understand.

Automation does not interpret authorship.
It executes registration.

*Ghost in the Machine* is not a book about outrage.
It is a book about architecture.

It traces the lifecycle of a single composition claim applied on 24 July 2017.

The claim lasted thirty days.
The investigation lasted eight years.

What began as a dashboard notification became a study of:

- Digital fingerprinting systems
- Composition asset registration
- International publishing chains
- Collecting society governance
- Monetisation routing mechanics
- Data protection law
- Enterprise data retention layers
- Judicial thresholds of proof

At its core, this book examines a structural tension of the digital age:

Registry precedes execution.

When a composition asset is registered, platforms trust it.
When a fingerprint matches, policy executes.
Revenue routes.
The burden shifts.

The system is not malicious.
It is scalable.

Scalability introduces opacity.

Interfaces display outcomes.
Archives retain objects.
Users see banners.
Systems store lifecycles.

Storage is not visibility.

This book documents:

• A composition claim applied to an original work
• Monetisation remaining active during dispute
• A thirty-day lifecycle ending in interface disappearance
• Confirmation of registry presence at SGAE in Valencia
• Publishing chain correspondence
• A regulatory Subject Access Request following judicial adjournment
• Disclosure via Legal Retrievals
• Delivery of approximately 1,899 timestamped activity records, without the underlying structured claim object
• Judicial acknowledgment that the Court had no reason to doubt authorship, together with an arguable case of infringement (HHJ Hacon, 20 October 2025)

The narrative is disciplined.
It avoids speculation.
It relies on timestamp, document, and structure.

The “ghost” in this title is not conspiracy.
It is presence beyond interface.
It is the structured data object that governs outcomes while remaining unseen.

As digital platforms increasingly mediate creative expression, understanding system design becomes essential to understanding authorship.

The question is not whether automation governs creative rights.
It already does.

The question is whether creators understand the architecture that governs them.

This work distinguishes between documented evidence, reasonable inference, and stated position. Where conclusions extend beyond disclosed material or judicial findings, they are identified as interpretation rather than established fact.

## Context

This book is written from the position of an independent composer working within contemporary digital distribution systems.

I am not employed by a rights management organisation.
I am not affiliated with a technology platform.
I am not legally trained.

When the composition claim described in this book was applied, my understanding of automated copyright systems extended only as far as the interface allowed.

I released music independently through standard commercial channels. Like many creators, I relied on distribution services, publishing registries, and digital platforms without visibility into the infrastructure beneath them.

The claim did not begin as a systems inquiry.

It began as a notification.

What followed over eight years was not activism.
It was structured investigation.

Understanding the lifecycle of that claim required learning:

- How digital fingerprinting systems operate
- How composition assets are registered and administered
- How publishing chains distribute responsibility across territories
- How collecting societies interface with platforms
- How enterprise data retention layers function
- How civil procedure defines evidential thresholds

The perspective offered in this book is neither institutional nor adversarial.

It is experiential and architectural.

The argument presented here is not that automated systems are malicious.

It is that they are layered.

Creators benefit from understanding the layers that govern their work.

# PART I

# BEFORE THE MACHINE

Creation precedes automation.

Before metadata, before distribution identifiers, before asset IDs and lifecycle states, there was sound.

*Peace To The Self* was written during a period of personal grief, composed and released under the name “A Man Barely Alive”.

My mother had been diagnosed with cancer. The composition emerged in the quiet intervals between hospital visits, consultations, and the recalibration that serious illness brings. It was not written for publication strategy. It was not conceived as a commercial release.

It was written as an act of steadiness.

The title was literal.

It was an attempt to bring calm into a situation that could not be controlled — to bring peace to the self.

In 2016, *Peace To The Self* began not as a product, but as repetition at a keyboard.

The room was small. Not a commercial studio. No isolation booth. No acoustic branding. Just a desk, a digital piano, and studio monitors positioned by instinct rather than calculation.

Music written in solitude carries a different quality.

There is no imagined audience.
No algorithm to satisfy.
No optimisation to consider.

There is only resonance.

The composition unfolded slowly. Chord progressions returning to themselves. Phrases resolving gently rather than dramatically. It was not written for competition.

It was written for alignment.

There were multiple saved versions.

Version 2.
Version 3_final.
Version 3_final_REAL.

# Chapter 1

## Before the Claim

Every independent creator understands the quiet comedy of file naming.

Creation leaves digital traces.

Each saved version carries a timestamp applied automatically by the operating system. At the time, those timestamps feel incidental — a routine by-product of digital work.

Later, they become anchors.

15 August 2016

On 15 August 2016, an early version of *Peace To The Self* was uploaded publicly.

It was not strategic.

It was instinctive.

A desire to share.

The upload did not feel consequential. There was no awareness that dates might later be examined or that chronology might one day matter.

At that moment, the internet felt open.

Neutral.

Uncontested.

## 18 January 2017 — The Break

The burglary did not announce itself loudly.

There were no shattered windows.

No cinematic disorder.

The door between the studio and the house was open.

It was never open.

That door marked a boundary. The studio was its own contained environment — cables, monitors, drafts, unfinished phrases. The house was separate.

The door stayed closed.

Always.

The room was recognisable.

But it was no longer neutral.

The door had been forced. Equipment disturbed. Storage devices moved. A passport taken.

There is something uniquely unsettling about returning to a creative space that has been physically entered by someone else.

Not because of what is missing.

But because of what may have been seen.

A keyboard is an object.
A monitor is an object.
A hard drive is an object.

But drafts are not objects.

They are unfinished thoughts.

Half-resolved harmonies.

Fragments not yet shaped for public hearing.

Standing there, I understood something that had not previously required articulation:

Creation without preservation is fragile.

After that night, preservation ceased being administrative.

It became reflex.

External drives were purchased.
Cloud storage was activated.
Folders were reorganised.
Version histories were retained deliberately.

Documentation was no longer precaution.

It was prudence.

Without knowing it, I had begun preparing for a dispute that had not yet occurred.

Creation continued.

But documentation now accompanied it.

## Public Trace

On 2 June 2017, a local newspaper referenced my music publicly.

Ink on paper carries a different solidity in the digital age.

Print does not refresh.

It does not update silently.

It does not disappear when a policy changes.

The date sat there, fixed.

Another anchor.

On 7 July 2017, *Peace To The Self* was commercially released via Ditto Music.

Distribution created identifiers.

ISRC codes.
Platform listings.
Streaming metadata.
Territorial availability.

The composition entered global systems.

Those systems distributed.

They did not contest.

Seventeen days later, YouTube's Content ID system detected a match.

In July 2017, there was no anticipation of conflict.

The composition existed.

The timestamps were secure.

The distribution was live.

The machine had not yet spoken.

Creative life felt linear.

Write.
Record.
Upload.
Release.

There was no awareness that behind the upload button existed a layered infrastructure of fingerprint databases, asset registries, policy engines, and monetisation routing systems.

The internet still felt like a platform.

Not an adjudicator.

That innocence matters.

Because when the claim appeared on 24 July 2017, it did not enter suspicion.

It entered trust.

And that is why it felt disorienting.

The machine had been invisible.

And then, one day, it spoke.

# PART II

# THE CLAIM

## Chapter 2

## 24 July 2017 — When the Machine Spoke

The upload itself was unremarkable.

Late afternoon light entered the room at an angle that flattened colour. The desk was familiar: keyboard slightly worn, studio monitors idle, cables coiled without ceremony. Uploading had become procedural long before that day.

Click **Create**.
Select file.
Watch the progress bar move from left to right.

Title.
Description.
Tags.

Peace To The Self was not new to the world. It had existed publicly since 15 August 2016. It had survived hard drives. It had survived burglary. It had survived commercial distribution metadata. It had entered streaming platforms on 7 July 2017.

Uploading it to YouTube felt less like creation and more like placement — another node in a digital footprint that already existed.

The progress bar completed.

Processing.

Thumbnail generated.

Visibility: **Public**.

There is always a brief silence after publishing any work online — a few seconds where the platform accepts the file but has not yet responded.

That silence feels neutral.

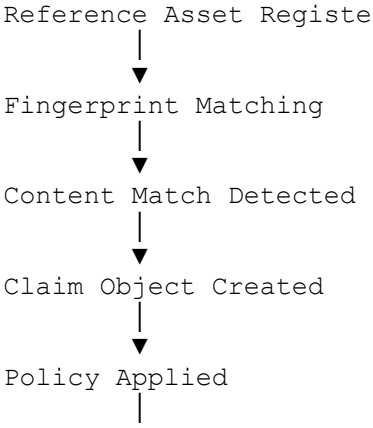
It is not.

Behind it, automated systems are already scanning. Every audio upload enters a detection environment in which waveform fingerprints are compared against registered reference files stored in enterprise databases. The process occurs in milliseconds and requires no human intervention.

The architecture behind this moment is rarely visible to creators. The simplified sequence through which automated composition claims are generated is illustrated in Figure 2.1.

---

Figure 2.1 — Simplified Automated Claim Lifecycle



Monetise / Track / Block

*Figure 2.1. Simplified representation of the automated lifecycle of a Content ID claim within a digital rights management system.*

---

At the time, none of that architecture was visible.

What appeared instead was a notification.

A grey icon.
A small alert.

Not a strike.
Not removal.

A **Content ID claim**.

The language was structured, almost clinical.

Content: *INTO THE DEEP BLUE SEA — Music composition*
Claimants: SGAE_CS; The Royalty Network (Publishing)
Matched segment: **0:50–2:42**
Policy: **Monetized by copyright owner**

The first sensation was not anger.

It was disorientation.

A composition claim does not target the recording. It targets the underlying musical work.

The recording was mine.
The composition was mine.

Yet the system had matched one minute and fifty-two seconds of my upload to a registered publishing asset bearing a different title.

The precision mattered.

Automation does not approximate.

It calculates.

I opened the claim details.

There is something uniquely unsettling about structured certainty. The system did not say *potential similarity*. It did not say *manual review pending*.

It displayed:

**Monetized by copyright owner.**

Policy had already executed.

Somewhere inside the platform's architecture, an asset object existed titled **INTO THE DEEP BLUE SEA**.

That asset contained ownership shares.
It contained a predefined policy.

And when my waveform crossed a similarity threshold against a stored fingerprint, the asset activated.

The machine had not accused.

It had applied.

That distinction would take years to understand.

Below the claim details was a button.

**Dispute.**

The wording beneath it was firm:

"I am sure that I own all rights to the audio and visual content in this video, and I want to dispute this claim."

Not *I believe*.
Not that *I think*.

**I am sure.**

Certainty is an interesting demand in automated systems.

Machines operate in binary states.

Claim exists.
Claim disputed.
Claim released.
Claim upheld.

Ambiguity is human.

States are system.

I selected the checkbox.

Typed my name.

Pressed submit.

The interface refreshed.

**Status: Dispute in progress.**

At **19:14**, the confirmation email arrived.

The header metadata read:

SPF: PASS
DKIM: PASS
DMARC: PASS

The machine had authenticated its own communication.

The email stated that the claimants had **up to thirty days** to review the information provided.

The clock began.

24 July 2017 — Claim applied
24 July 2017 — Dispute filed
23 August 2017 — Review window ends

The approved judgment later recorded this sequence in simple chronological terms. On 24 July 2017, following upload of *Peace To The Self* to YouTube, Content ID generated a play-match notification identifying an approximately 1 minute 52 second segment of *Into the Deep Blue Sea* and stating that monetisation would be applied in favour of the claimant. A dispute was filed through the YouTube system, which indicated a thirty-day review window. On the final day of that window, the visible trace of the claim disappeared from the interface, and no further outcome was communicated.

This account is relied upon here only as chronology recorded by the Court. It does not determine the cause of the match, nor does it attribute authorship or liability beyond the procedural posture described in the proceedings.

Nothing in the room had changed.

The desk was the same.
The light was the same.
The composition was unchanged.

But a structural shift had occurred.

Monetisation was active under someone else’s registered asset.

Execution had preceded review.

In that moment, I did not yet know the claim would stretch across **eight years**.

I only knew that a composition claim now existed inside an automated rights system — and that the system had spoken with confidence.

# Chapter 3

## Play-Match

The button is small.

It does not look consequential.

**“Play matched segment.”**

A neutral phrase.

No warning.
No commentary.
No implication.

Just a tool.

When pressed, the interface divides the screen.

Two audio streams.
Two waveforms.
Two timelines aligned at the matched segment.

**0:50–2:42.**

One minute and fifty-two seconds.

The system is not arguing.

It is demonstrating.

The reference audio plays.
My upload plays.

The segments overlap.

There is something uniquely unsettling about hearing your own composition mirrored back at you under a different title.

Not distorted.
Not remixed.
Not sampled.

Aligned.

Play-Match does not speculate.

It overlays.

It does not accuse.

It compares.

The effect is forensic.

You stop listening as a composer.

You start listening as an analyst.

Is that harmonic progression unique?
Is that chord resolution common?
Is the tempo similar?
Is the structure coincidental?

The mind races faster than the audio.

But the system remains indifferent.

It is not concerned with originality.

It is concerned with similarity.

Fingerprint thresholds do not understand intention.

They detect pattern.

The moment of listening becomes instructional.

The claim is no longer abstract.

It is auditory.

The overlap exists.

The policy has executed.

And the question shifts instantly from:

“Is this mine?”

to:

“How did this asset become registered?”

The machine has not explained itself.

It has presented evidence of its own logic.

Play-Match is the closest the system comes to transparency.

It shows you what it sees.

And in doing so, it reveals the gap between **authorship** and **registry**.

The composition was written in a small room.

Saved under incremental filenames.

Released under my name.

Yet here it was, aligned against a registered asset titled **Into the Deep Blue Sea**.

The moment lasted **one minute and fifty-two seconds**.

Its implications lasted **eight years**.

# Chapter 4

## Monetised During Dispute

The most revealing feature of the system did not appear in removal.

It appeared in continuity.

The video remained public.

There was no strike.
No takedown.
No warning visible to viewers.

From the outside, nothing appeared altered.

Inside the dashboard, however, a different state persisted:

**Monetized by copyright owner.**

The phrase was neutral.

It offered no timeline.
No explanation.
No indication of provisional status.

It stated an outcome.

The composition asset attached to the claim carried a predefined policy. When the fingerprint matched, that policy executed automatically.

The system did not pause.

It did not wait for review.
It did not suspend revenue pending clarification.

It executed.

---

## Execution Before Review

When a composition asset is registered in Content ID, it contains:

• Ownership percentages
• Territorial scope
• Policy configuration

If the policy is set to **Monetise**, advertising revenue generated by a matched video is allocated according to the stored percentages.

The system does not re-evaluate authorship at upload.

It performs a sequence:

Fingerprint → match → asset located → policy executed.

The dispute changes the state of the **claim object**.

It does not automatically change the policy governing the asset.

Registry authority persists unless released.

From a systems perspective, this is rational.

At global upload scale, platforms cannot suspend monetisation upon every assertion of ownership. They rely on prior registration.

Stability precedes correction.

But the inversion is structural.

Execution precedes review.

---

## The Quiet Displacement

The recording was mine.

The composition was mine.

The dispute had been filed within minutes.

Yet the economic state of the video had already shifted.

The phrase **"Monetized by copyright owner"** introduced a subtle divergence between authorship and control.

Not dramatic.
Not catastrophic.

But precise.

For thirty days, the system would treat the registered asset as authoritative.

Not because it had investigated.

Because it had been encoded.

That distinction matters.

Ownership, in automated environments, is initially a matter of **registry**, not assertion.

---

## The Economics of Silence

If no dispute is filed, monetisation continues.

There is no automatic expiry.

No periodic reassessment.
No system-generated doubt.

Revenue flows according to stored ownership data until something interrupts it.

Silence becomes economically meaningful.

In this case, the dispute had been filed immediately.

But monetisation remained active during the review window.

The policy did not suspend itself.

The clock did not display itself.

The dashboard reflected only the current state:

**Dispute in progress.**
**Monetized by copyright owner.**

Two truths occupying the same space.

---

## A System in Motion, A Room Unchanged

Nothing in the room had changed.

The desk was the same.
The monitors were still.

The composition remained what it had always been.

Yet somewhere inside a distributed infrastructure of registries, databases, and policy engines, a claim object had been created and monetisation logic was executing against it.

The system was functioning exactly as designed.

And that realisation altered perception more than the claim itself.

This was not accusation.

It was architecture.

---

## The Window Opens

The dispute confirmation email had stated:

The claimant has up to 30 days to review your dispute.

Thirty days.

The phrase did not sound significant when first read.

But now it carried structure.

Monetisation active.
Review pending.
Clock running.

The system would not move again unless acted upon — or unless the window expired.

The banner remained.
The policy remained.
The state persisted.

And time, inside automated systems, behaves differently.

---

# Chapter 5

## The Thirty-Day Window

Time behaves differently inside automated systems.

It does not negotiate.
It does not accelerate in response to urgency.
It does not pause because a human feels uncertain.

It maintains state until an input changes it.

When I filed the dispute on **24 July 2017**, confirmation arrived almost immediately. The claimant, I was told, had up to **thirty days** to review the dispute.

Thirty days.

At the time, the phrase felt procedural. Administrative. Routine.

Only later did it become clear what that window represented.

The dispute did not suspend the claim.

It altered its status.

The claim persisted.
The applied policy persisted.
Execution continued.

The system did not pause while uncertainty was examined.

It preserved continuity.

---

## Two Truths

During the thirty-day period, the dashboard displayed two simultaneous realities:

**Dispute in progress.**
**Monetized by copyright owner.**

Both were accurate.

A review process had been initiated.

Monetisation had not been interrupted.

There was no visible countdown.
No progress indicator.
No insight into internal workflow.

Externally, nothing had changed.

The video remained accessible.
Viewers saw no warning.

Internally, however, a structured state existed.

A claim had been instantiated.
It was bound to a registered asset.

That asset carried a predefined policy.

The policy continued to execute.

Filing a dispute did not dissolve that structure.

It marked it.

**Registry precedes execution.**

Until verification overrides registry, registry governs execution.

---

## Deterministic Design

Content ID does not deliberate.

It evaluates.

If a reference matches, a claim is generated.
If a claim is active, policy is applied.
If multiple policies apply, the most restrictive governs.
If a dispute is filed, the claim's status changes.

Policy does not necessarily suspend.

This is not accusation.

It is architecture.

A deterministic system preserves its last validated state until an authorised change occurs.

Waiting does not create ambiguity.

Waiting preserves execution.

---

## Silence Has Weight

In automated governance, silence carries consequence.

If no dispute is filed, the claim remains.
If no review action occurs, the policy remains.
If no release is issued, monetisation continues.

The system does not infer doubt from inactivity.

It maintains continuity.

The thirty-day window was therefore not an empty pause.

It was a bounded interval in which a separate actor could alter the claim's state.

Nothing within the system suggested it would self-correct.

Interruption required input.

---

## Human Time

Inside that window, days accumulated with a different texture.

Not panic.
Not outrage.

Awareness.

Each login presented the same configuration:

Claim active.
Dispute submitted.
Monetisation applied.

The room remained unchanged.

The composition remained unchanged.
The upload date remained constant.

Yet an alternate rights narrative existed within infrastructure.

The two realities did not collide.

They coexisted.

---

## State and Transition

What I did not yet understand in July 2017 was that automated rights governance is organised around **objects and state transitions**.

A claim is not a conversation.

It is a structured entity.

It can be **active**, **inactive**, **pending**.
It can carry policy.
It can generate revenue allocations.
It can persist beyond visibility.

Filing a dispute initiates a potential transition.

It does not erase the object.

The thirty-day window was not waiting in the abstract.

It was a defined lifecycle phase.

Execution during that period was not provisional in tone.

It was stable.

---

## Release

On **23 August 2017**, the banner disappeared.

There was no architectural explanation.

No disclosure of internal reasoning.

The claim ceased to be visible within the dashboard.

Externally, it appeared resolved.

Internally, the underlying structure remained opaque.

Whether the reference had been modified,
whether ownership data had been adjusted,
whether the claim object had been archived or transitioned —

none of this was visible.

Visibility changed.

Structure may have persisted.

The system updated its state.

---

## After the Window

The thirty days ended quietly.

The video remained.
Monetisation returned.
The interface cleared.

But something fundamental had shifted.

Before **24 July 2017**, I experienced platforms primarily as **distribution infrastructure**.

After **23 August 2017**, I understood them as **governance infrastructure**.

Distribution moves content.

Governance adjudicates rights.

The thirty-day window revealed which layer I was interacting with.

Not because the system malfunctioned.

Because it functioned precisely as designed.

---

## Architecture Revealed

Years later, examining report schemas, reference structures, and claim identifiers, the thirty-day window became legible in a new way.

It was not a period of uncertainty within the system.

It was a stable object in an active state.

A claim had been instantiated.
A dispute had been registered.
Policy continued to execute.
A state transition eventually occurred.

The human experience of that period was ambiguity.

The system experience of that period was continuity.

Understanding that distinction reframes the event entirely.

Time behaves differently inside automated systems.

It does not drift.

It waits for input.

# Chapter 6

## The Invisible Burden

Automated systems do not initiate doubt.

They execute registry.

When the composition was flagged on **24 July 2017**, the system did not pause for inquiry. It did not suspend policy pending clarification. It applied the instructions encoded within the registered asset and began routing revenue accordingly.

Execution was immediate.

Review was conditional.

The architecture assumes registry authority until formally interrupted.

This creates what might be described as an **invisible burden**.

The burden of activation rests with the creator.

If no dispute is filed, monetisation continues. There is no automatic expiry triggered by silence. There is no periodic reassessment initiated by the system.

Stability is preserved unless action alters state.

Silence, in automated environments, is not neutral.

It allows continuity.

This is not accusation.

It is design logic.

At global upload scale, platforms cannot suspend execution while awaiting certainty. They rely on prior declarations encoded within asset objects.

Registry establishes presumption.

Policy enforces it.

Interruption requires input.

The inversion is subtle.

Traditionally, authorship begins with the creator.

In automated rights infrastructure, operational authority begins with registration.

Registry asserts.

Automation executes.

The creator must intervene.

The dispute mechanism exists.

But it is reactive.

The system does not activate review on its own initiative.

Understanding this alters perception.

The claim did not persist because it was believed.

It persisted because it had been registered.

Correction required deliberate activation.

The burden was not emotional.

It was structural.

---

## The Thirty-Day Blind Spot

When the dispute was filed, the claim entered a review window.

The dashboard displayed:

Dispute in progress.
Monetized by copyright owner.

Two states coexisted.

Execution continued during review.

Thirty days later, the banner disappeared.

No detailed explanation accompanied the release.

No breakdown of interim revenue routing appeared at interface level.

No structured claim object was exposed for inspection.

From the user's perspective, the matter had concluded.

From an architectural perspective, a lifecycle transition had occurred.

Years later, when account data was exported, the visible activity history began on the date the claim was released. The claim itself was absent from the user-accessible record.

This revealed a boundary.

Operational systems generate and retain structured objects.

Interface systems render simplified states.

Export systems provide curated history.

When a claim disappears from a dashboard, the underlying object does not necessarily cease to exist. It may migrate from an active system to an archival or compliance layer.

Visibility is not retention.

Display is not storage.

An automated event may govern revenue for a defined period, then vanish from the interface once resolved.

The system retains memory.

The user sees closure.

This is not concealment by default.

It is layered architecture.

But for the individual creator, the effect can be disorienting.

An event that altered monetisation, triggered formal dispute, and required intervention may leave no visible trace within ordinary export tools.

The machinery persists beyond the surface.

Understanding that distinction is essential.

In automated environments, what is unseen is not necessarily absent.

It is often simply deeper.

# Chapter 7

## Registry Authority

The claim did not originate inside the video.

It originated inside a registry.

That distinction matters.

Digital platforms do not invent ownership.

They ingest it.

Before a claim is applied to a video, a reference must exist.

Before a reference exists, an asset must be registered.

Before an asset is registered, ownership must be asserted somewhere upstream.

By the time a creator sees a banner, the system is already acting on data that predates the upload.

Registry precedes execution.
Execution precedes review.
Presumption precedes correction.

This sequence describes the logic of scale.

Registry establishes a structured source of authority.

Execution applies that authority automatically when conditions are met.

Presumption maintains stability until a formal challenge interrupts it.

Each step reduces friction at scale.

Each step prioritises continuity over hesitation.

This ordering is not accidental.

It is structural.

---

## Upstream Trust

At scale, platforms cannot independently verify authorship each time a match occurs.

Millions of hours of content are uploaded daily.

Billions of data points circulate through publishing and collecting societies.

Fingerprints are generated automatically.

Claims are applied deterministically.

The system must trust something.

It trusts registry data.

When a publisher, administrator, or rights holder registers a composition, that registration becomes authoritative within the ecosystem in which it operates.

Authority here does not imply moral certainty.

It implies operational priority.

The platform does not decide who composed a work.

It executes based on declared ownership transmitted through structured channels.

Registry data enters the system as presumption.

Dispute becomes the mechanism for challenge.

---

## Presumption at Scale

Presumption is efficient.

Without presumption, every match would require manual investigation before policy could be applied. That would collapse automation.

Instead, the system assumes registry validity until contested.

This principle is not unique to copyright infrastructure.

Financial systems operate on declared balances until reconciled.

Identity systems operate on stored credentials until revoked.

Logistics systems route packages based on encoded address data until corrected.

Scale demands presumption.

Presumption demands hierarchy.

Hierarchy places registry above individual assertion at the moment of execution.

That is why the claim in July 2017 did not arrive as a question.

It arrived as a state.

---

## The Asymmetry

There is an asymmetry between registration and dispute.

Registration is proactive.

Dispute is reactive.

Registration may occur long before a conflict surfaces.

Dispute occurs after execution has begun.

The burden of interruption rests with the party affected by the applied policy.

This inversion can feel unjust at an individual level.

But at scale, it is predictable.

If platforms waited for perfect certainty before applying policy, enforcement would stall.

If they refused to rely on registry declarations, rights holders would withdraw participation.

Automation requires structured input.

Structured input requires hierarchy.

Hierarchy produces asymmetry.

Understanding this does not dissolve frustration.

It contextualises it.

---

## Why Verification Follows Execution

Verification is resource-intensive.

It requires documentation, correspondence, and sometimes legal process.

Execution, by contrast, is computational.

Match.
Instantiate claim.
Apply policy.

Milliseconds.

Verification unfolds in days, weeks, months, sometimes years.

The order reflects cost.

Computation is inexpensive.

Investigation is not.

The thirty-day window described earlier was not a courtesy.

It was a buffer between computational certainty and human verification.

Within that window, the system preserved its initial presumption.

Registry remained authoritative.

Until altered.

---

## Distributed Ownership

Modern compositions rarely exist within a single ownership layer.

A composer may assign publishing rights.

A publisher may appoint sub-publishers in specific territories.

A collecting society may administer performance rights.

An administrator may ingest reference files.

A platform may rely on that ingestion.

By the time a claim appears, multiple entities may have contributed metadata.

No single node necessarily sees the entire chain.

Each node sees its layer.

This diffusion reinforces registry authority.

If multiple structured inputs converge on the same declared ownership, the presumption strengthens automatically.

Dispute becomes the only mechanism that interrupts convergence.

---

## Architecture, Not Intention

It is tempting to interpret presumption as bias.

But architecture often explains what emotion misattributes.

The system is not designed to doubt registry at the moment of execution.

It is designed to act on it.

Doubt enters later.

Review enters later.

Documentation enters later.

That sequence is mechanical.

Understanding that sequence reframes the original event.

The banner was not an accusation.

It was the visible endpoint of an upstream registry assertion moving through automated infrastructure.

Once seen this way, the event shifts category:

From confrontation
to architecture.

---

## Authority and Stability

Registry authority is not infallible.

It is foundational.

Systems require foundation.

Without a default source of truth, automation fragments.

Registry provides that foundation.

Dispute provides correction.

The tension between the two is structural, not personal.

When the claim in July 2017 was released, registry authority had been interrupted by dispute.

But the architecture remained intact.

That architecture continues to govern millions of works daily.

The system did not change because of one challenge.

It functioned as designed.

Understanding that does not remove responsibility from participants within the chain.

It clarifies the environment in which they operate.

Registry precedes execution.
Execution precedes review.
Presumption precedes correction.

In automated ecosystems, that ordering is the price of scale.

# PART III — INVESTIGATION

## Chapter 8
### 23 March 2018 — Valencia: Registry Confirmed

By March 2018, the dashboard phase had ended.

The claim had appeared.
It had been disputed.
It had been released.

The interface was quiet.

But silence on a dashboard does not resolve a registry question.

On 23 March 2018, I visited SGAE in Valencia.

The office did not resemble a technology company.

It resembled administration.

Paper signage.
Reception desk.
Fluorescent lighting.
Institutional furniture.

The architecture of a collecting society is not designed for spectacle.

It is designed for continuity.

SGAE — Sociedad General de Autores y Editores — administers composition rights in Spain**.**

Not recordings.
Not performances.

Compositions.

It records authorship.
It tracks usage.
It distributes royalties.

Its authority is not algorithmic.

It is registry-based.

Standing inside that office, the distinction between platform and registry became tangible.

YouTube executes.

SGAE records.

A platform reacts to fingerprints.

A collecting society maintains titles, ownership shares, and territorial representations.

I was not there to accuse.

I was there to confirm existence.

Did the title Into the Deep Blue Sea exist within the publishing registry?

The answer was yes.

It was registered.

It was associated with named publishing entities.

It existed within a formal rights management structure.

That confirmation shifted scale.

The claim applied to my upload had not emerged from algorithmic hallucination.

It had flowed from registry data.

Automation had trusted a title that existed.

---

## Registry Weight

A registry entry is heavier than a dashboard notification.

It implies:

- Ownership percentages
- Administrative control
- Territorial scope
- Sub-publishing arrangements

Publishing chains are layered.

Original composer.
Primary publisher.
Territorial sub-publisher.
Collecting society.
Digital reporting ingestion.

By the time a composition asset reaches a platform like YouTube, it has often passed through multiple databases.

The platform does not re-adjudicate authorship.

It trusts upstream data.

In Valencia, the claim ceased being a platform anomaly.

It became a publishing chain question.

---

## Governance Context

Collecting societies operate on trust.

They act as intermediaries between creators and the systems that monetise creative work.

They aggregate rights.
They administer shares.
They collect revenue.
They distribute royalties.

Their authority is institutional.

Platforms rely on that authority.

Automation scales that reliance.

If registry data is flawed, automation will still execute it.

This is not negligence.

It is dependency.

At the time of my visit in March 2018, SGAE was publicly navigating scrutiny relating to governance issues connected to late-night broadcasting and royalty distribution patterns — widely referred to as “La Rueda.”

This context is not evidence of misconduct in my case.

It is evidence of complexity.

When institutions managing vast repertoires face scrutiny, it highlights something essential:

Registry integrity is foundational to automated execution.

Automation does not monitor governance debates.

It executes data entries.

Standing inside SGAE's office, the atmosphere was administrative, not dramatic.

Files.
Desks.
Personnel moving through routine tasks.

The architecture of rights management is mundane.

But its consequences are not.

When registry data flows into automated systems, it acquires scale.

A single entry can propagate across platforms, territories, and monetisation systems.

And once encoded, it operates without intention.

---

## Structural Shift

After Valencia, the question was no longer:

"Why did YouTube do this?"

The question became:

"How did this composition asset enter the registry?"

That question required correspondence.

Not with a platform.

With publishers.

The claim had moved from **interface to infrastructure**.

And infrastructure is rarely visible from a dashboard.

---

# Chapter 9

## Correspondence and the Publishing Chain

The Valencia visit confirmed existence.

But existence is not explanation.

A registry entry records a title.

It does not narrate origin.

It does not explain how ownership shares were entered.

It does not describe how a composition asset became linked to a fingerprint.

To understand the pathway, correspondence became necessary.

This phase was not adversarial.

It was architectural.

---

## Fragmentation as Structure

Music publishing appears simple from a distance.

Composer.
Publisher.
Royalty distribution.

In practice, it is layered.

An original composer may assign rights to a primary publisher.

That publisher may appoint sub-publishers for specific territories.

Collecting societies administer performance and mechanical rights.

Digital platforms ingest ownership data from registry feeds.

The platform that applies a claim may not be the entity that registered the composition.

The collecting society that records the title may not be the publisher that originally entered it.

The administrator responding to correspondence may not control the asset at platform level.

Responsibility is distributed.

Understanding that distribution was essential.

The claim was not the action of a single visible hand.

It was the output of a chain.

## The Royalty Network and Administrative Boundaries

Correspondence directed toward representatives associated with **The Royalty Network** sought clarity on publishing administration.

The responses reflected professional compartmentalisation.

Recording rights were distinguished from composition rights.

Territorial administration was separated from global ownership.

Platform-level claims were described as automated reflections of registry data rather than manual decisions made at upload.

There was no overt hostility.

No dramatic denial.

There was structure.

Each entity operated within defined parameters.

Each pointed toward upstream or downstream layers.

It became clear that publishing chains are not linear.

They are networked.

Ownership data flows through databases across jurisdictions.

When it reaches a platform like YouTube, it arrives already structured.

The platform does not adjudicate originality.

It executes based on received metadata.

## Territorial Layers

Further correspondence reinforced the same principle: publishing administration is segmented by territory and function.

Territorial sub-publishers may administer rights within specific regions.

Ownership percentages may differ across territories.

Data may pass through multiple systems before reaching digital platforms.

This complexity does not imply error.

It implies infrastructure.

The composition asset associated with the claim existed within that infrastructure.

The match was not accidental.

It was registry-driven.

The correspondence did not yield a dramatic admission.

It yielded confirmation of fragmentation.

No single entity controlled the entire chain.

No single email could resolve it.

---

## Major Publisher Compartmentalisation

Following the **March 2018** registry confirmation in Valencia, the registration details identified **Sony/ATV Music Publishing Holdings (Spain)** and **Ediciones Musicales Clippers S.L.** as publishing entities associated with the composition title registered at SGAE.

This introduced another layer to the publishing chain.

At SGAE's suggestion, I attended **Sony/ATV's Madrid office**.

The meeting was professional and measured.

I was informed that Sony and Clippers were listed as joint publishing entities in relation to the registered title.

The matter, I was told, would be reviewed internally.

No confrontation occurred.

No accusation was exchanged.

The interaction reflected administrative process.

Subsequent correspondence escalated to **Sony Music Entertainment International** at legal level.

The response clarified corporate separation.

Sony Music Entertainment distinguished **recording rights** from **composition administration** and indicated that the matter likely fell within publishing structures rather than recording ownership.

Further communication revealed additional segmentation:

• Spain → United Kingdom
• Publishing division → legal division
• Legal → advertising and games divisions
• Territorial publisher → sub-publisher
• Sub-publisher → administrative representative

At one stage, it was explained that relevant divisions operated independently and that the business was structurally compartmentalised.

**Clippers Music Publishing** later confirmed its role as sub-publisher in Spain for catalogue administered by **The Royalty Network**.

The Royalty Network, in turn, had been identified within the original **Content ID claim** as the publishing administrator linked to the reference asset.

No single entity asserted complete authorship control over the registered composition.

Instead, responsibility appeared distributed across:

• A collecting society (SGAE)
• A territorial publisher (Sony/ATV Spain)
• A sub-publisher (Clippers)
• An international administrator (The Royalty Network)
• Corporate divisions operating within distinct legal frameworks

Each correspondence clarified its own limited remit.

None presented the entire chain.

What emerged was not contradiction.

It was structure.

Modern publishing architecture is distributed by design.

Ownership data flows across jurisdictions, subsidiaries, and administrative layers before it reaches digital platforms.

When that data is ingested into automated systems such as Content ID, it acquires scale.

The claim applied on **24 July 2017** was therefore not merely a platform event.

It reflected a registered composition asset moving through a fragmented rights administration network.

Compartmentalisation did not negate persistence.

It explained it.

---

## The Realisation

By the end of this phase, the claim no longer felt like a singular event.

It felt like an **emergent property of a distributed system**.

Registry entry.
Ownership shares.
Territorial representation.
Digital ingestion.
Fingerprint matching.
Policy execution.

Each layer was functioning according to its own logic.

The issue was not malice.

It was alignment.

Somewhere within that publishing chain, a composition asset had been registered under a title that triggered a match against my work.

The system had trusted it.

The correspondence did not provide a simple correction.

It clarified that the claim's origin lay **upstream of the platform**.

The question had shifted again.

Not:

"Why did YouTube apply the claim?"

But:

"How did the publishing data that YouTube relied upon come into being?"

The inquiry had moved beyond interface and beyond platform.

It now lived within the architecture of rights administration itself.

---

# Chapter 10

## May 2021 — The First Data Boundary

By May 2021, the dispute had matured into something quieter.

The claim had long since disappeared from the dashboard.

Correspondence had clarified publishing fragmentation.

The Valencia visit had confirmed registry presence.

But one question remained unresolved:

**Where is the record?**

Not the banner.
Not the email.
Not recollection.

The structured object.

If a fingerprint match generated a claim on **24 July 2017**, then a claim object must have been created within YouTube's rights management system.

That object would have contained:

• Video ID
• Asset ID
• Matched segment timestamps
• Policy configuration
• Lifecycle state transitions
• Dispute flags
• Release status

Enterprise systems do not execute policy without structured entities.

So where was it?

---

## Google Takeout

In **June 2021**, I initiated a **Google Takeout** export.

The interface presents itself as comprehensive.

Download your data.
Your videos.
Your comments.
Your account history.

The archive arrived as a compressed file.

**12.7 MB.**

It felt small.

That perception matters.

Years of platform activity condensed into a file smaller than a high-resolution photograph.

Extracted.

Opened.

HTML activity logs.
Video metadata.
Channel settings.
Upload records.

At first glance, it appeared complete.

But then something emerged.

The archive's activity history began on **23 August 2017**.

The exact date the claim disappeared from the dashboard.

Before that date, no activity entries were displayed in the exported log.

The boundary was precise.

**23 August 2017.**

The thirtieth day.
The release date.

That precision altered the tone of inquiry.

This was not random truncation.

It was segmentation.

---

## Segmentation Revealed

User export tools do not mirror backend systems.

They provide curated data layers.

The Takeout archive demonstrated this clearly.

Interface-level history was accessible.

Certain logs were exportable.

But deeper claim lifecycle objects were absent.

There was no structured **Content ID claim object** in the export.

No asset identifier.
No dispute state transition log.
No monetisation routing record.

The claim had existed.

It had governed monetisation.

It had transitioned states.

Yet the user-level export did not contain the object that governed those actions.

The archive revealed a boundary between:

**User-accessible data**

and

**Enterprise-level operational data.**

The ghost had not disappeared.

It had descended.

---

## Storage vs Display

This was the first moment architectural clarity overtook suspicion.

Enterprise systems are layered.

Operational databases.
User interface layers.
Export systems.
Compliance archives.

The disappearance of the claim from the dashboard in **August 2017** was a display decision.

The absence of the claim object in the **2021 Takeout export** was a segmentation decision.

Neither necessarily implied deletion.

They implied layering.

Storage does not equal display.

Display does not equal retention.

Retention does not equal disclosure.

The system had depth.

The claim had existed.
It had been processed.
It had been released.

But the structured entity governing that lifecycle was not present in the user-export layer.

The ghost inside the machine was not mystical.

It was architectural.

---

## The First Clear Line

May and June **2021** marked a shift.

Before this, the issue was **publishing chain complexity**.

After this, the issue was **data architecture**.

The question sharpened:

If the object exists in operational storage but is absent from export layers, what mechanisms govern access?

That question would not be answered through dashboard navigation.

It required legal framing.

And that framing would come **three years later**, in **October 2024**.

---

# Chapter 11

## Hundreds of Nights

The claim lasted thirty days.
The investigation lasted years.

What began as a notification became a pattern of evenings spent reconstructing the system behind it.

Browser tabs accumulated.
Registry entries were cross-checked.
Asset identifiers were traced.
Corporate filings were read.
Correspondence was catalogued.
Screenshots were preserved.
Statutory provisions were printed and annotated.

The work was incremental.

There was no single breakthrough.
Understanding emerged through accumulation.

One evening clarified how fingerprint thresholds operate.
Another illuminated the publishing chain behind a registered composition.
Another revealed the distinction between interface display and backend object state.

Each discovery was small.
Together they formed architecture.

Time altered scale.

A thirty-day dispute window expanded into an eight-year inquiry — not because of obsession, but because each answer generated another structural question.

If the claim had been released, what triggered the release?
If the claim object disappeared from the interface, where did it migrate?
If registry authority initiated execution, what governed registry priority?
If retention exceeded visibility, what layer preserved the object?

The matter no longer concerned revenue.
It concerned coherence.

Friends asked why it still mattered.
Colleagues suggested that the release of the claim should have been sufficient.

It would have been easier to accept that view.

But something shifts when authorship is formally contested by automation.

The issue ceases to be financial.
It becomes structural.

If registry can overwrite chronology without scrutiny, silence becomes precedent.

Understanding became necessary.

Not only to clarify authorship, but to restore internal equilibrium — to bring peace to the self.

Clarity required persistence.
Persistence required time.

There were evenings when the scale of the system was difficult to ignore.

Digital platforms operate across billions of uploads.
Fingerprint systems scan continuously.
Policies execute automatically.

Against that scale, one composition appears negligible.

But scale does not nullify authorship.
It contextualises it.

The architecture that governs billions governs the individual equally.

Mapping that architecture required more than correspondence.

It required literacy.

The documentation was not hidden.

Public developer resources described bulk reporting systems, system-managed asset and claim reports, policy application logic, and the structured fields that govern lifecycle transitions — claim identifiers, asset identifiers, match origins, applied policies, territorial ownership flags.

The language was technical rather than explanatory.

But it revealed something essential.

The system was object-driven.

Claims had primary keys.
Assets had state.
Policies resolved according to rule hierarchies.

What appeared opaque at dashboard level was structured at reporting level.

The architecture was documented.

It required literacy.

With each layer clarified, emotional intensity receded.

Structure replaced speculation.
Architecture replaced assumption.

The matter slowed.

Not because it weakened.
Because it matured.

By the time the issue reached court, the groundwork had already been laid in those evenings.

The evidence assembled.
The chronology preserved.
The architecture mapped.

The claim lasted thirty days.

The literacy required to understand it lasted far longer.

And once acquired, that literacy remained.

# PART IV — LEGAL PHASE

## Chapter 12

### October 2024 — Adjournment and Judicial Threshold

By **October 2024**, the dispute had moved beyond correspondence and private inquiry.

It was before the Court.

Litigation alters tempo.

In private investigation, questions can expand without constraint. In court, questions must narrow. They must be framed within rule, evidence, and threshold.

The application before the **Intellectual Property Enterprise Court** sought **Norwich Pharmacal relief**: disclosure to identify a potential wrongdoer and to obtain information necessary to bring proceedings.

The structure of such relief is strict.

A wrong must be arguable.
The order must be necessary.
The respondent must be mixed up in the alleged wrongdoing.
Disclosure must be proportionate.

These are not rhetorical standards.

They are jurisdictional limits.

---

## The First Hearing

The application first came before the Court on **10 October 2024**.

At that stage, I was represented by counsel.

The evidential basis was primarily inferential: the Content ID match, the registry confirmation, the absence of structured claim disclosure, and the need to identify the creator of *Into the Deep Blue Sea*.

The Court was not satisfied, on the material then before it, that the requirements for relief had been met.

That conclusion was not dramatic.

It was procedural.

Threshold had not yet been crossed.

But something unusual followed.

In the approved judgment, **HHJ Hacon** records:

“Exceptionally, because Mr Heal had for the most part been acting without legal assistance, I agreed to an adjournment to allow him to find and to file further evidence.”

The word **“exceptionally”** carries weight.

Adjournments of this nature are not automatic.

Where the evidential threshold is not met, dismissal is ordinarily the outcome.

The Court exercised discretion.

Not to decide the issue.

To allow further evidence.

That distinction matters.

The adjournment did not imply endorsement.

It did not imply success.

It did not imply that the claim would ultimately satisfy the requirements.

It indicated that the evidential question warranted further opportunity.

---

## Threshold Clarified

Norwich Pharmacal relief requires more than suspicion.

It requires an **arguable wrong** — defined in jurisprudence as a case more than barely capable of serious argument.

The alleged wrongdoer in this instance was the creator of *Into the Deep Blue Sea*.

The inference relied upon was straightforward:

If Content ID correctly identified similarity,
and if the composition predated the registered asset,
then there may be infringement in the opposite direction.

But inference is not proof.

The Court identified alternative possibilities:

• The algorithm may have erred.
• Similarity may have arisen independently.
• There may have been confusion between titles.

The existence of possibilities weakens certainty.

And certainty is not required — but arguability must be demonstrated.

The adjournment allowed refinement of that demonstration.

---

## From Narrative to Record

Leaving the courtroom after the **October 2024 hearing**, something shifted internally.

The issue was no longer explanatory.

It was evidential.

Dashboards do not persuade courts.

Emails do not substitute for disclosure.

Suspicion does not satisfy threshold.

The structured object — the **Content ID claim lifecycle entity** — became central.

If it existed, it would anchor inference.

If it did not, its absence would narrow possibility.

The adjournment created a defined window.

Not a delay.

An evidential opportunity.

---

## Exceptional Discretion

The significance of the adjournment was not emotional.

It was structural.

Courts guard jurisdiction carefully.

Relief compelling disclosure from third parties is exceptional. It intrudes into privacy and commercial systems. It is granted only when necessary and proportionate.

To adjourn rather than dismiss is to recognise that the matter is not frivolous, but incomplete.

The Court did not relax the test.

It preserved it.

But it allowed further material to be obtained.

That procedural posture altered the dispute.

The question was no longer:

“Why did this happen?”

It became:

“What evidence satisfies threshold?”

The answer to that question required a statutory mechanism.

And that mechanism was the **Subject Access Request submitted in November 2024**.

---

# Chapter 13

## November 2024 — The Subject Access Request

The **October 2024 adjournment** narrowed the issue.

The question was no longer explanatory.

It was evidential.

If a **Content ID claim object** had been instantiated on **24 July 2017**, then structured data must have existed within YouTube’s systems reflecting:

- The creation of the claim
- The matched segment
- The applied policy
- The dispute submission
- The lifecycle transition
- Any monetisation routing during the review window

Without that structure, inference remained incomplete.

The mechanism available was statutory.

---

## The Legal Basis

Under **Article 15 of the UK GDPR** and **section 45 of the Data Protection Act 2018**, an individual is entitled to obtain confirmation as to whether personal data concerning them is being processed, and, if so, access to that data.

The language is deliberate.

“Personal data.”
“Concerning.”
“Identifiable individual.”

The law does not compel narrative explanation.

It compels disclosure of personal data.

The distinction is critical.

The request was therefore framed not as accusation, but as classification:

Does the **Content ID claim object** — created in connection with my account, my video, my dispute submission, and my monetisation state — constitute personal data?

If it did, it was disclosable.

If it did not, that boundary would define the limits of access.

---

## Drafting with Precision

The instinct in such moments is to request everything.

But breadth weakens clarity.

The request was targeted.

It sought:

• The structured Content ID claim object associated with the **24 July 2017 notification**
• Lifecycle state transitions linked to that object
• Monetisation routing data during the **thirty-day dispute window**
• Asset identifiers connected to the matched reference
• Internal logs referencing my account during that period

The wording avoided speculation.

It did not ask:

“Why did this happen?”

It asked:

“Provide the data relating to this object.”

Architecture responds better to precision than to emotion.

---

## Interpretation and Boundary

A data controller must assess what constitutes personal data.

Enterprise systems contain:

• User-level account data
• Operational metadata
• Internal system identifiers
• Integrity logs
• Privileged internal analysis

Not every system object automatically qualifies as personal data.

The claim object, however, if it contains:

• My channel identifier
• My video ID
• My dispute submission
• Timestamps of my account activity
• Monetisation flags affecting my upload

may arguably relate to an identifiable individual.

The boundary is interpretive.

Disclosure depends on classification.

---

## January 2025 — Legal Retrievals

In **January 2025**, a response was received indicating that the requested data had been supplied via **"Legal Retrievals."**

The phrase revealed architecture.

"Legal Retrievals" implies:

• Archived environments
• Compliance-layer storage
• Data segregated from ordinary interface export

This was distinct from the **2021 Google Takeout export**.

The Takeout archive had demonstrated segmentation.

The reference to Legal Retrievals demonstrated retention.

The claim lifecycle had not necessarily vanished.

It had likely transitioned.

The remaining question was scope.

Would the structured claim object be disclosed?

Or would disclosure be limited to extracted logs deemed personal?

---

## Classification, Not Drama

The SAR process is not adversarial by default.

It is classificatory.

A controller determines what falls within the statutory definition of personal data.

It extracts what it considers responsive.

It may withhold what it classifies as operational system architecture.

This does not automatically imply concealment.

It reflects legal interpretation.

But interpretation can be tested.

The statutory clock began.

---

## Structural Maturity

By this stage, something had shifted internally.

The early phase of the dispute had been reactive.

The present phase was analytical.

The layers were mapped:

Fingerprint.
Asset.
Claim object.
Policy.
Lifecycle.
Retention.

The **Subject Access Request** was not an appeal for explanation.

It was a structural inquiry.

If the claim object related to my identifiable account activity, it should be disclosable.

If it did not, that boundary would define the architecture more clearly.

Either outcome narrowed ambiguity.

And narrowing ambiguity, at this stage, was progress.

---

# Chapter 14

## February 2025 — The USB and the 1,899 Timestamps

In February 2025, a USB device was delivered by courier.

Not a dashboard link.
Not a downloadable archive.
Physical media.

The method mattered.

A USB implies extraction.
Selection.
Deliberate transfer from one environment to another.

For years, the claim had existed in fragments — notification, correspondence, registry confirmation, statutory request.

Now there was tangible output.

The device was inserted.
The files opened.

The structure was orderly:
Folders.
CSV files.
Account-level logs.
Copies of uploaded content.

The volume appeared substantial.

Approximately 1,899 timestamped activity records were present within the dataset. These records reflect logged system and account-level activity associated with the disclosed materials, rather than the native Content ID claim object.

Rows of:
• Account activity
• System acknowledgements
• Upload events
• Modifications

Retention was evident.
History had not been erased.

But volume is not architecture.

---

# Logs and Objects

A timestamp records that something occurred.
An object governs how and why it occurred.

Enterprise systems do not execute monetisation without structured entities.
They do not process disputes without lifecycle states.
They do not release claims without state transitions.

A Content ID claim object, if created on 24 July 2017, would have contained:
• Video ID
• Asset ID
• Matched segment markers
• Policy configuration
• Dispute status
• Lifecycle timestamps

The disclosed material contained activity logs. It did not include the native Content ID claim object, nor associated monetisation or complaint-level metadata relating to the original claim.

There was no document labelled:
“Content ID Claim — 24 July 2017.”

Instead, there were entries indicating that events had occurred.

Logs are derivative.
Objects are generative.

This distinction is structural, not semantic.

---

# Retention Confirmed

The USB demonstrated something important.

Data from 2017 had been preserved.
The 2021 Takeout export had revealed segmentation.
The 2025 USB disclosure revealed retention.

Neither demonstrated deletion.

The claim lifecycle had not simply evaporated after 23 August 2017.
It had likely migrated.

Enterprise systems prioritise auditability.
High-risk objects are rarely destroyed casually.

They transition:
Active → Archived
Interface-visible → Compliance-retained

The USB confirmed preservation at some layer.
It did not collapse the layers.

---

# Metadata and Placeholder Dates

The files within the USB disclosure were date-stamped 30/12/1899. This date does not correspond to any plausible creation or activity date in the context of the materials.

As set out in my witness evidence, I interpret this date as a default placeholder value, consistent with known system behaviour where metadata is missing, not preserved, or not exported with the file. The presence of a uniform placeholder date across the dataset is therefore consistent with the absence of original timestamp metadata.

Accordingly, the 30/12/1899 date is not treated here as a genuine record of creation or activity, but as an indicator that the underlying temporal metadata associated with the files was not included in the disclosure.

This interpretation is based on the structure and presentation of the disclosed materials. The disclosure itself consisted of extracted files and activity records, rather than a full replication of the underlying system environment.

The absence of original timestamp metadata does not, of itself, establish the reason for that absence. It indicates a limitation in the disclosed dataset rather than providing a definitive explanation of system-level handling.

---

# Expectation and Boundary

It would have been easy to interpret the absence of the native claim object as obstruction.

But enterprise disclosure is governed by classification.

A controller must determine what constitutes personal data.

It may extract:
• Account-linked logs
• Activity records
• User-facing metadata

It may not extract:
• Full operational object structures
• Internal system identifiers
• Proprietary architecture

The disclosed material reflected interpretation.
It did not necessarily reflect the full architecture.

This is where the boundary becomes visible.

Dashboard display is not archive.
Archive is not export.
Export is not object-level replication.

The approximately 1,899 timestamped activity records demonstrate activity.
They do not demonstrate mechanism.

---

# Structural Clarity

By February 2025, the shape of the system was clearer than at any previous point.

The claim had been:
Applied.
Disputed.
Released.

Each stage required object-level state change.

If the object existed — and the architecture suggests it must have — it resided within a deeper layer than user export.

The ghost inside the machine is not disappearance.
It is the structured entity that governs outcomes while residing beneath controlled layers of visibility.

The USB did not conclude the inquiry.
It clarified its depth.

The materials provided therefore represent a partial dataset of recorded activity, rather than a full replication of the underlying system architecture.

# Chapter 15

## Legal Retrieval Architecture

By early **2025**, the dispute was no longer about whether something had happened.

It was about **where it resided**.

The claim had existed.
It had governed monetisation.
It had transitioned state.
It had disappeared from interface.

The **Takeout export** demonstrated segmentation.

The **USB disclosure** demonstrated retention.

The remaining question concerned architecture.

## Layered Systems

Large-scale digital platforms do not operate as single databases.

They operate in layers.

**Operational Layer**

The live environment.

Uploads are scanned.
Fingerprints are compared.
Assets are matched.
Claim objects are instantiated.
Policies execute.

Speed is prioritised.

Scale is constant.

Explanation is secondary.

---

**Interface Layer**

Dashboards render simplified states:

“Claim active.”
“Dispute in progress.”
“Claim released.”

These are **representations**, not native objects.

Visibility is curated.

---

**Export Layer**

User tools such as **Google Takeout** provide account-level history and metadata.

They do not replicate backend object structures.

Export is filtered architecture.

---

**Compliance Layer**

Data preserved for audit, litigation, and regulatory purposes resides within controlled environments.

Access occurs through defined legal channels.

The reference to **“Legal Retrievals”** indicated such a layer existed.

---

## Object Migration

The disappearance of a claim from a dashboard is not evidence of deletion.

In layered systems, disappearance typically reflects **transition**.

Active → Archived
Visible → Segmented
Operational → Compliance

Deletion removes existence.

Migration alters location.

High-risk audit environments favour retention.

The architecture suggested that the claim object — if instantiated — would persist somewhere within structured storage.

Whether it was disclosed in full was a separate legal question.

Architecturally, its prior existence was unavoidable.

Monetisation cannot route without an asset.

A dispute cannot be processed without a claim object.

A release cannot occur without state change.

Structure precedes interface.

---

## Architecture Without Intention

It is tempting to personify systems.

But architecture is not motive.

Systems are designed to:

Scale.
Execute.
Retain.
Defend.

They are not designed to narrate their own transitions to users.

Opacity is often a by-product of layering, not concealment.

When understood structurally, the ghost dissolves into design.

What once appeared mysterious becomes procedural.

The machine did not accuse.

It executed registry.

It did not erase.

It migrated.

It did not deliberate.

It processed.

---

## Threshold Approaches Recognition

By the time the matter returned to court in **October 2025**, the architecture had been mapped.

The issue was no longer confusion about how the system worked.

It was whether authorship had been infringed.

Automation had acted in milliseconds.

The legal system would now apply deliberation.

Between those two tempos lay **eight years of literacy**.

And literacy alters posture.

The machine had executed.

The Court would decide.

# PART V — CONCLUSION

## Chapter 16

20 October 2025 — Recognition

This chapter addresses the approved judgment as a matter of legal recognition at threshold, not a final determination of infringement. The Court's findings are limited to the existence of an arguable case and the absence of doubt as to authorship and copyright ownership. The judgment does not determine the cause of the Content ID match or the internal operation of any system. Any broader discussion in this book is presented as analysis and not as a finding made by the Court.

The objective that remains is narrow and specific.

It is not to determine liability.
It is not to attribute fault.

It is to understand the structure of the Content ID claim that operated between 24 July 2017 and 23 August 2017.

That includes:

• Whether one or more claim objects were instantiated
• The asset or assets to which those claims were linked
• The manner in which monetisation was routed during the dispute period
• The lifecycle transitions associated with dispute and release
• The location and form of any retained records relating to that lifecycle

The interface presented a simplified state.

The subsequent export did not include the underlying claim object.

Later disclosure demonstrated retained activity without revealing the governing structure.

The question that remains is therefore structural:

What was the configuration of the claim that governed the composition during that thirty-day period?

This is a matter of record reconciliation.

It does not extend the findings of the Court.

It seeks only to understand the system-level structure that produced the outcome observed.

On 20 October 2025, the matter returned to the Intellectual Property Enterprise Court.

The sequence was no longer fragmented.

24 July 2017 — Claim applied.
24 July 2017 — Dispute filed.
23 August 2017 — Claim released.
23 March 2018 — Registry confirmed in Valencia.
May–June 2021 — Export boundary observed.
October 2024 — Adjournment granted.
November 2024 — Subject Access Request submitted.
February 2025 — USB delivered.

What began as a dashboard notification had become an evidential pathway.

The courtroom atmosphere was measured.

Digital systems operate in milliseconds.
Courts operate in deliberation.

The issue before the Court was narrow:

Whether there was an arguable case that copyright in Peace To The Self had been infringed.

An arguable case is not a final determination.
It does not assign liability.
It does not award damages.
It does not conclude proceedings.

It recognises that the evidential threshold has been met sufficiently for the claim to proceed.

That distinction is central.

In the approved judgment, HHJ Hacon stated that the similarities identified by Content ID were sufficient for him to conclude that there was an arguable case that there had been infringement of my copyright in Peace To The Self.

He further recorded that he had no reason to doubt that I wrote the composition and that I owned copyright in it.

This represents a threshold assessment and not a final determination of infringement.

The Court left open the possibility that the Content ID process may have produced an incorrect outcome, alongside other potential explanations. Content ID operates through audio fingerprint matching rather than title or name comparison, and the Play-Match function demonstrated approximately 1 minute and 52 seconds of overlapping audio within a composition of approximately five minutes in duration.

The judgment does not determine the cause of that match or prefer one explanation over another. The interpretation of how such a match may arise — whether through error, independent creation, or issues within registration, ownership attribution, or reference-data alignment — is not resolved by the Court and is addressed in this book as analysis rather than judicial finding.

Recognition did not alter the music.

The composition had not changed.
The chronology had not changed.
The timestamps had not changed.

Its legal posture had.

For eight years, authorship had existed as documentation and conviction:

Saved drafts.
Version histories.
Public uploads.
Distribution records.
Preserved correspondence.

Now it existed within judicial acknowledgment.

There was no surge of triumph.

There was alignment.

The early years of the dispute contained ambiguity:

Was the match erroneous?
Was the registry authoritative?
Was the similarity coincidental?

By October 2025, that ambiguity had narrowed.

The Court did not determine final infringement.
It determined that the case was properly arguable.

Automation had acted instantly in 2017.
Recognition required eight years.

The machine executed registry.
The Court applied threshold.

Between those two moments lay structure.

And structure, once understood, alters perception permanently.

---

# Chapter 17

## The Ghost and the Author

The claim lasted thirty days.
The architecture lasted eight years.

On 20 October 2025, the Court stated that it had no reason to doubt that I wrote *Peace To The Self* and owned copyright in it, and accepted that there was an arguable case of copyright infringement.

An arguable case is not a final determination.
It does not conclude proceedings.
It does not assign liability.

It recognises that the evidential threshold has been met sufficiently for the matter to proceed.

That distinction matters.

For eight years, authorship had existed as timestamp, version history, public trace, distribution record, preservation discipline, and procedural persistence.

Now it existed within judicial acknowledgment.

The composition had not changed.
The chronology had not changed.
The sound had not changed.

Its legal posture had.

Recognition aligned creative fact with procedural structure.

But recognition did not explain the machine.

Understanding had already done that.

---

# The Ghost Defined

The concept of the “ghost” is used here as an analytical model to describe how claims, references, and attribution may persist or transition within automated systems. It is not a finding made by the Court, but a framework developed to interpret the documented sequence of events.

The ghost inside the machine is not a person.
It is not conspiracy.
It is not absence.

It is structured persistence beyond interface visibility.

On 24 July 2017, a fingerprint threshold was met.
A claim object was instantiated.

That object referenced a registered composition asset containing declared ownership, territorial scope, and policy configuration.

When the match triggered, the attached policy executed.

No deliberation occurred at upload.
No inquiry preceded application.

The system acted on what had already been encoded.

When I filed a dispute, the object did not dissolve.
It transitioned state.

When the review window elapsed, it transitioned again.
When the banner disappeared, visibility changed.

Structure did not necessarily disappear with it.

At no stage did the system require belief.
It required registration, policy, and state.

---

## Authority and Inversion

Automated rights systems operate on prior declaration.

Before any upload is scanned, reference files have been deposited.
Assets have been instantiated.
Ownership shares have been asserted.
Policies have been configured.

When similarity is detected, the platform does not adjudicate originality in real time.
It evaluates against registered reference data and applies the stored instruction set attached to the matched asset.

Registration is not philosophical proof.
It is operational priority.

Execution is computational.
Review is procedural.

One occurs in milliseconds.
The other unfolds in days, sometimes years.

Hierarchy is therefore inevitable.

The uploader must activate correction.
The system does not initiate doubt.

This inversion is structural.

If no dispute is filed, monetisation continues.
There is no automatic expiry triggered by silence.

Continuity is preserved unless formally interrupted.

The claim did not persist because it was believed.
It persisted because it had been registered.

---

## Visibility and Retention

The dashboard displayed the claim.
The dashboard later removed the claim.

The 2021 export demonstrated segmentation.
The 2025 Legal Retrieval confirmed retention.

Enterprise systems are layered.

Operational environments generate structured objects.
Interface layers render simplified states.
Export tools provide curated history.
Compliance environments preserve archival records.

When a claim disappears from interface, it ceases to be displayed.
That does not mean the governing object has ceased to exist.

Deletion removes existence.
Segmentation removes visibility.

In high-risk environments, deletion is not necessarily the default outcome.
Migration between system layers may occur.

Objects transition from active state to archived state.
From interface-visible to compliance-retained.

Visibility is presentation.
Retention is preservation.

The ghost is not erasure.
It is depth.

---

## Human Narrative, System State

Human experience is narrative.
System operation is stateful.

The human waits.
The system records timestamps.

The human experiences disappearance.
The system records transition.

The human seeks explanation.
The system executes stored policy.

Tension arises when interface presentation is mistaken for full architecture.

Once that distinction becomes clear, ambiguity reduces.

What appeared arbitrary becomes procedural.

The system is not theatrical.
It is deterministic.

---

## After Recognition

Recognition did not create authorship.
It confirmed it.

What changed in October 2025 was not the composition.
What changed was posture.

Before recognition, authorship was internally certain and externally contested.
After recognition, it was formally acknowledged.

That shift did not produce triumph.
It produced alignment.

Eight years earlier, the machine had executed registry authority without deliberation.
Now the Court had applied judicial threshold with deliberation.

Automation acted in milliseconds.
Recognition required years.

Between those two moments lay literacy.

---

## Literacy

Something irreversible happens when you understand how systems operate.

Dashboards cease to feel neutral.
Disappearance ceases to imply deletion.
Claims cease to feel accusatory.

Registry becomes visible.
Layering becomes legible.

The machine does not become smaller.
It becomes clearer.

The claim lasted thirty days.
The education lasted eight years.

The ghost inside the machine is not malevolent.
It is the structured persistence of data within layered systems operating at scale.

When understood, it loses mystique.

And when mystique dissolves, literacy remains.

The machine executes.
The author endures.

---

# POSTSCRIPT

Following the conclusion of this manuscript and the Court's recognition on 20 October 2025 that there was no reason to doubt authorship, together with its acceptance that there was an arguable case of copyright infringement, the investigation has entered a final phase of formal record reconciliation.

This phase concerns the alignment of digital infrastructure with legal reality. Where automated systems once executed registry-based presumption, the objective now is to ensure that those systems reflect the authorship of the Work as recognised for the purposes of the proceedings.

The matter is no longer one of uncertainty.

It is one of correction.

The structural question identified in Chapter 16 — concerning the configuration of the Content ID claim and its lifecycle between 24 July 2017 and 23 August 2017 — now moves from analysis toward reconciliation.

The architecture examined throughout this book does not conclude with recognition. It continues through the processes by which records are updated, data is reconciled, and system states are brought into alignment with legal outcome.

This phase is procedural.

It is precise.

This phase ensures that digital architecture reflects the legal position of the Work.

The process continues.

For live updates, technical glossaries, and continued documentation of this process, visit:

mechanicalpublishing.com

# APPENDIX I

## How Content ID Actually Works

This appendix exists to separate **feeling from mechanism**.

Content ID is not a metaphor.

It is an engineered system.

Understanding its architecture is essential to understanding how a composition claim is **created, executed, and retained**.

---

## 1. Reference Files and Fingerprints

## Content ID operates on digital fingerprinting**.**

Rights holders upload reference files — audio recordings or audio-visual content — to YouTube's rights management system. These files are processed to generate mathematical signatures derived from frequency patterns, timing intervals, and waveform characteristics.

The fingerprint is **not a copy of the song**.

It is an abstraction.

When a user uploads a video, YouTube scans the audio track and compares its fingerprint signature against registered reference files.

If overlap exceeds a defined threshold, a match is generated.

The system does not evaluate context.

It does not assess authorship.

It evaluates **similarity between fingerprints**.

Match detection precedes policy execution.

---

## 2. The Asset Object

When a reference file is uploaded, it is tied to an **Asset**.

An asset is a structured database object containing:

• Asset ID
• Title metadata
• Rights holders
• Ownership percentages
• Territory information
• Policy settings (Monetise, Block, Track)

The asset governs what happens when a fingerprint match occurs.

The fingerprint triggers the asset.

The asset triggers the policy.

The system does not negotiate at upload.

It executes predefined instructions.

---

## 3. Claim Object Creation

When a match is detected, the system generates a **Claim Object**.

This object typically contains:

• Video ID
• Asset ID
• Matched segment start and end times

• Claimant identity
• Policy status
• Dispute state
• Lifecycle timestamps

This object governs the lifecycle of the claim.

When a dispute is filed, the object changes state.

When a claim is released, the object changes state again.

The dashboard reflects these states.

It does not contain them.

---

## 4. Policy Execution and Monetisation Routing

Each asset within Content ID carries a predefined policy.

Common policy types include:

• Monetise
• Block (worldwide or territory-specific)
• Track

When a fingerprint match triggers a claim object, the associated asset policy executes automatically.

If the policy is set to **Monetise**, the system enables advertising on the video. Revenue generated is allocated according to the ownership percentages stored within the asset object.

This allocation is not calculated dynamically at the moment of upload.

It is governed by stored share splits entered during asset registration.

For example:

If an asset reflects multiple publishing entities with defined percentages, revenue distribution follows those proportions.

The system does not evaluate the uploader’s claim of ownership at this stage.

It executes based on **registry data**.

Execution is deterministic.

---

## 5. Monetisation During Dispute

When an uploader files a dispute, the claim object changes state to **“Under Review.”**

However, filing a dispute does not automatically suspend policy execution.

The asset remains attached to the claim object.

The monetisation policy remains active.

Revenue generated during the review period may be:

• Held pending resolution
• Allocated based on final determination

But the system does not assume uploader ownership simply because a dispute has been filed.

Registry authority persists unless released.

This design reflects scalability requirements.

With millions of daily uploads, automated systems must prioritise registry consistency over individual assertion.

The burden of activation lies with the uploader.

Absent dispute, the claim persists indefinitely under the governing policy.

---

## 6. Dispute Lifecycle Architecture

A typical dispute lifecycle follows defined stages:

1. Claim applied
2. Uploader files dispute
3. Claim enters review window (commonly 30 days)
4. Claimant may:
   • Release claim
   • Reject dispute
   • Allow review window to lapse
5. Claim object transitions to final state

Each stage is timestamped within the claim object.

The interface reflects these states.

The database stores them.

Lifecycle architecture ensures that claims are not manually re-evaluated at each stage.

They follow pre-coded conditional pathways.

If no action is taken within the defined review period, the system may automatically release the claim depending on policy configuration.

Automation governs state transitions.

---

## 7. Storage, Archiving, and Retention

Enterprise platforms separate data layers for operational efficiency and compliance.

Common layers include:

• Active operational databases
• User-facing interface layers
• Exportable account data layers
• Compliance and litigation archives

When a claim is released, it may no longer appear in the dashboard.

This reflects an **interface decision**.

It does not necessarily indicate deletion of the claim object.

Backend retention may persist for:

• Audit purposes
• Legal compliance
• Regulatory inquiries
• Internal integrity checks

These retention environments are often segregated from user-accessible systems.

The term **“Legal Retrievals”** suggests data preserved within such compliance-oriented storage layers.

Retention exceeds visibility.

---

## 8. Extraction vs Object-Level Disclosure

When data is disclosed in response to regulatory or legal request, it may be provided as:

• Extracted activity logs
• CSV reports
• Timestamp lists

These are **derivative outputs**.

They do not necessarily replicate the internal claim object in its native database form.

Object-level disclosure would involve providing the structured entity as stored, including all associated fields and state transitions.

Extraction demonstrates activity.

Object disclosure demonstrates architecture.

The distinction is material in complex rights disputes.

---

## 9. Architectural Summary

Content ID operates on four foundational principles:

1. Fingerprints trigger matches.
2. Assets govern policy.
3. Claim objects manage lifecycle.
4. Retention layers preserve records beyond interface display.

The system is not interpretive.

It is procedural.

Understanding these mechanics clarifies how a composition claim can:

• Be applied instantly
• Remain monetised during dispute
• Disappear from the dashboard after release
• Persist in backend retention layers

The ghost inside the machine is not absence.

It is **structured persistence beyond surface visibility**.

---

# APPENDIX II

This appendix clarifies the sources consulted in understanding the structural operation of the automated rights systems described in this book.

The analysis presented in the preceding chapters was **not derived from speculation or proprietary access**. It was informed by publicly available documentation published by platform operators and developer resources.

These materials are accessible to content owners, administrators, and technical partners. They describe system architecture at **reporting and API level**, rather than at user interface level.

The documentation reviewed included, among other materials:

---

## YouTube Reporting and Analytics API Documentation

• Content Owner Reporting API structures
• System-managed Asset Reports
• System-managed Claim Reports
• System-managed Reference Reports

Field definitions describing:

• Claim identifiers
• Asset identifiers
• Policy identifiers
• Lifecycle timestamps
• Match origin classifications
• Territorial ownership indicators
• Status flags

---

## Policy Application Documentation

Materials explaining:

• Monetisation, tracking, blocking, and takedown policy hierarchies
• How policies are applied when fingerprint thresholds are met
• Clarification that applied policies are attached to registered assets and execute automatically upon match

---

## Asset and Reference Eligibility Guidance

Documentation describing:

• Requirements for reference file submission
• Ownership declaration standards
• Publishing and administrator ingestion processes
• Distinctions between **composition assets** and **sound recording assets**

---

## Binding and Object Relationships

Materials describing how **assets, references, claims, and videos** are bound within structured object relationships.

These resources define:

• Claim lifecycle state transitions
• Object-level identifiers used in reporting systems
• The separation between interface display and backend reporting objects

---

## Architectural Implications

These materials are technical in language.

They do not narrate events.

They describe **schema**.

What they reveal is structural:

Content ID operates through **object instantiation**.

Claims possess identifiers.

Assets possess state.

Policies execute according to predefined rule hierarchies.

Lifecycle transitions are recorded within reporting environments.

The documentation does not adjudicate authorship.

It defines architecture.

---

## Purpose of Consultation

The purpose of consulting these materials was not technical development or reverse engineering.

It was **architectural literacy** — to understand how a composition claim could be created, executed, transitioned, and retained within layered systems.

The distinction between **dashboard presentation** and **backend object structure** is fundamental.

Interface displays simplified states.

Reporting environments describe structured entities.

Compliance layers preserve retained data.

Understanding this layered design clarified that automated outcomes are not conversational decisions.

They are **deterministic executions of stored registration data interacting with fingerprint analysis**.

The architecture is public.

It requires reading.

---

# APPENDIX III

Monetisation Economics and Escrow Architecture

This appendix explains the **financial mechanics underlying composition claims**.

Its purpose is to describe structure rather than allegation.

---

## 1. Advertising Revenue Model

YouTube monetisation generally operates through:

• Advertising impressions
• Cost per thousand views (CPM)
• Revenue-share agreements between platforms and rights holders

When monetisation is enabled, advertising revenue is generated based on viewer engagement and advertiser bidding structures.

Revenue is then allocated according to **policy rules attached to registered assets**.

---

## 2. Composition Claims and Revenue Allocation

When a composition claim is applied:

The matched segment becomes associated with a **publishing asset**.

If the asset's policy is set to **"Monetise,"** revenue attributable to the video may be allocated according to the ownership shares stored within that asset.

Those shares are entered at the **registration level**.

The system does not dynamically renegotiate allocation during upload.

It executes **stored percentages**.

---

## 3. Dispute Period Revenue Handling

During a dispute window, revenue generated from the video may be:

• Held pending the outcome of the dispute
• Allocated after the claim is resolved
• Distributed according to the final state of the claim

The key structural point is that **monetisation is not automatically suspended when a dispute is filed**.

The asset remains attached to the claim object.

The policy remains active unless modified.

This design prioritises **continuity at scale**.

---

## 4. Escrow Logic

Escrow in automated rights systems refers to the **provisional holding of revenue pending claim resolution**.

Escrow does not imply neutrality.

It operates within the framework of the existing asset's authority.

If the claim is released, revenue may revert to the uploader.

If the dispute is rejected, revenue may be distributed to the claimant.

The mechanism is **rule-based rather than discretionary**.

---

## 5. Indefinite Monetisation Risk

If no dispute is filed:

The claim remains active.

Monetisation continues under the governing policy.

There is **no automatic expiry** absent a lifecycle trigger.

This is a structural feature of automated governance systems.

The burden of activation lies with the uploader.

---

## 6. Economic Architecture Summary

The monetisation system operates through a simple structural chain:

**Registry → Asset → Policy → Revenue Routing**

Automation ensures scalability.

But scalability shifts responsibility toward creators to **monitor claims and activate dispute mechanisms when necessary**.

Financial execution precedes human review.

Understanding that inversion is essential to understanding **digital rights economics**.

What began as a YouTube copyright *notification* became *an eight-year investigation into the* infrastructure that governs creative work online.

In *Ghost in the Machine*, composer James Heal reconstructs the hidden architecture of automated rights enforcement after a Content ID claim was applied to his original composition *Peace To The Self*. Through forensic analysis, registry investigation, and legal proceedings in the Intellectual Property Enterprise Court, Heal reveals how modern digital platforms execute copyright decisions through layered systems of fingerprinting, registry authority, and monetisation routing.

Part investigative narrative, part technical examination of digital governance, the book exposes the structural ordering that defines authorship in the twenty-first century: registry precedes execution, automation precedes review, and platforms now operate as global systems influencing the determination of creative ownership.

*Ghost in the Machine* offers a rare inside account of how automated copyright systems actually function—and what it takes for individual authorship to be recognised within them.