YouTube

Written by Febogamedev

History

The history of YouTube is particularly important because it coincides with a broader transformation of the audiovisual system.

Before video-sharing platforms became widespread, publishing audiovisual content and reaching a substantial audience usually required access to a television broadcaster, a distributor or a producer, or independently managing a Web infrastructure.

YouTube gradually changes this model by introducing a much simpler principle: anyone can produce a video, upload it and potentially distribute it to a global audience.

The platform therefore does not emerge primarily as a content producer.
It emerges as an infrastructure that makes it easy to publish, find, play and share content produced by others.
YouTube is founded in 2005 by Chad Hurley, Steve Chen and Jawed Karim, who had previously worked at PayPal. The YouTube.com domain is registered on 14 February 2005. The founders' stated aim is to create an environment in which anyone with a video camera and an Internet connection can easily share a video with other people (Wojcicki, 2020; YouTube, 2005).

The problem may seem trivial today, but in 2005 it was not.
A user wishing to publish a video on the Web had to overcome several obstacles:

- File size.
- Different formats.
- Hosting.
- Bandwidth required for distribution.
- Player compatibility.
- Difficulty sharing content on other sites.

YouTube seeks to hide this complexity.
The model becomes:

```
Video file
↓
Upload
↓
YouTube
↓
Web page / Player
↓
Sharing
```

The user does not need to know how the underlying infrastructure works.
They simply need to upload their video.
On 23 April 2005, the platform's first piece of content is published: Me at the zoo, a short video of Jawed Karim recorded at San Diego Zoo (YouTube, 2015).
In May, the service opens in beta and gradually becomes available to the public over the following months (YouTube, 2015).
By the end of 2005, YouTube is already receiving millions of daily views (Wojcicki, 2020).

The extremely simple nature of Me at the zoo is significant.
It is not a television production, a film or a professional music video.
It is a person in front of a video camera.

This feature anticipates one of the cultural changes introduced by YouTube: online video does not necessarily need the production standards of traditional media to have value.

The platform enables a much more immediate form of audiovisual communication: person → video camera → audience.
Instead of: production → broadcaster → schedule → audience.

YouTube's first phase is therefore dominated by the concept of user-generated content, meaning content produced directly by users.

In 2005, YouTube itself described its service as a system through which people could easily upload, tag and share personal videos, not only within YouTube but also through websites, blogs and email (YouTube, 2005).
In November of the same year, Sequoia Capital invests $3.5 million in the company to support its growth (YouTube, 2005).

This ability to take videos beyond the platform easily is particularly important.
YouTube does not just build a website on which to watch content.
It also builds a player that can be embedded on the Web.
A video can be published on YouTube and then embedded in other websites while its hosting and distribution remain on the platform.
This allows YouTube to grow through websites that do not belong to YouTube.

It is a distribution model particularly well suited to the Web of the early 2000s, when blogs and social networks are becoming increasingly important.

Growth is extremely rapid and immediately creates scalability problems.
As early as the end of 2005, the company speaks publicly about the difficulties caused by the speed of its expansion. The more users upload videos and the more people watch them, the faster storage, processing and distribution capacity must grow.

One of the problems that will continue to define YouTube's architecture therefore begins to emerge:
more creators → more videos → more storage → more processing → more traffic → more users → more creators → …

In 2006, the platform's growth becomes fast enough to attract Google.
On 9 October 2006, Google announces the acquisition of YouTube for approximately $1.65 billion in stock; the transaction is completed on 13 November (Google, 2006b; Google, 2006a).
Google decides to retain YouTube as an independent brand and product, while combining its community and video catalogue with Google's expertise in search, information organisation and online advertising (Google, 2006b).

The combination is particularly important because the two companies have complementary capabilities.
YouTube has: community + video + growth + a sharing culture.
Google has: search + advertising + infrastructure + data processing.

YouTube's problem is not just distributing enormous quantities of video.
As the catalogue grows, answering a second question becomes increasingly important: How can the right video be found within an ever-growing quantity of content?
Search, recommendations and catalogue organisation therefore gradually become as important as distribution itself.

Meanwhile, an economic problem also emerges.
Users produce the content that attracts the audience, but producing quality videos requires time and resources.
If the platform wants to develop a stable creator ecosystem, it must therefore enable at least some of them to earn a financial return.

In 2007, YouTube introduces the **YouTube Partner Program (*YPP*)**.

Initially, the programme applies to a limited group of particularly popular creators and is then gradually expanded. Partners can receive a share of the revenue generated by advertisements associated with their content (Hanif, 2022).
The change is fundamental.

The platform moves from "User → publishes a video" to "Creator → publishes content → builds an audience → generates revenue"

YouTube therefore begins to create a true creator economy.
The creator is not necessarily dependent on a television broadcaster, a studio or a publisher.
They can build directly: channel → audience → monetisation → new production
This also creates a powerful economic feedback loop.

```
Larger audience
↓
More potential revenue
↓
Greater incentive to produce
↓
More content
↓
```

Larger audience

During the same period, however, another inevitable consequence of the platform's openness emerges: users can also upload content for which they do not own the rights.

A platform based on millions of uploads cannot manually check every file.
YouTube must therefore turn copyright management into a technological problem.

In October 2007, the system that will become Content ID is introduced. Rights holders can provide reference files, and YouTube automatically compares content uploaded by users against this archive. If a match is found, the rights holder can block, track or monetise the video, depending on the policies applied (Zack, 2010; YouTube, n.d.-f).
Content ID profoundly changes the platform's model.

Copyright does not necessarily produce only "unauthorised content → removal", but can also produce "identified content → claim → monetisation".

In this way, automated rights management becomes both a control system and an economic infrastructure.

YouTube must therefore begin to manage at least three major ecosystems:

- Viewers → want to watch videos.
- Creators → want to publish and monetise them.
- Rights holders → want to control and monetise their works.

Between 2007 and 2010, another change profoundly alters the nature of the platform: video begins to move from the computer to the mobile device.

YouTube introduces its first mobile experience in 2007.
In subsequent years, it expands the mobile experience and develops dedicated applications for different mobile ecosystems (Doronichev, 2010).
In July 2010, YouTube Mobile already records over 100 million plays per day, a volume comparable to YouTube's entire daily traffic when Google acquired it four years earlier (Doronichev, 2010).

This transition is particularly important from an architectural perspective.
The model is no longer simply desktop + broadband, but becomes smartphone + Wi-Fi / mobile network + battery + variable screen.

The platform must therefore adapt video, bitrate, interface and distribution to much more heterogeneous conditions.

At the same time, content quality increases.
In 2009, YouTube introduces 1080p playback, while in 2010 it adds support for sources up to 4K (Sarukkai, 2010).
The transformation is significant.

The original YouTube was strongly associated with short clips and relatively low quality. Gradually, it must become a platform capable of distributing high-resolution audiovisual content as well.
The relationship becomes:

```
better video cameras
↓
larger uploads
↓
greater encoding complexity
↓
more representations
↓
more network traffic
```

The transcoding and distribution infrastructure must therefore grow alongside the quality of capture devices.

In 2009, YouTube also exceeds one billion views per day (Hurley, 2009).
At that point, Chad Hurley describes three principles that had guided the platform from the beginning (Hurley, 2009):

- Fast video start-up.
- The importance of short content.
- The platform's openness to user production.

Meanwhile, the scale of uploads also changes.

- In 2011, 48 hours of video are uploaded every minute (YouTube, 2012).
- By 2013, the figure already exceeds 100 hours per minute (YouTube, 2013).

These figures show why YouTube must develop an architecture profoundly different from that of a platform with a controlled catalogue.

A service such as Netflix can know which films will enter its catalogue.
YouTube, by contrast, must be ready to receive new content continuously and automatically turn it into distributable content at any time:

- A video recorded on a smartphone.
- A university lecture.
- A music video.
- A match.
- A tutorial.
- A two-hour documentary.

YouTube Live is officially introduced in 2011, following earlier experiments with live-streamed events (Wilms, 2016).
Live streaming changes the platform's architecture again because signal capture, ingest, encoding, distribution and playback must take place simultaneously.

YouTube had already streamed specific live events, but the introduction of YouTube Live gradually integrates this function into the product itself.
In 2012, Felix Baumgartner's jump from the stratosphere demonstrates online video's ability to support global events in real time. The report published by YouTube on the day of the event records a peak of over 8 million simultaneous views of the live stream: this is a concurrency figure, distinct from the total audience reached (Katz, 2012).

YouTube is therefore no longer just an archive in which to watch what has already happened.
It also becomes a medium through which to witness what is happening now.

In 2013, YouTube crosses another symbolic threshold: more than one billion unique users visit the platform each month (YouTube, 2013).
The scale achieved also changes the search problem.
When hundreds of hours of video are continuously uploaded, no user can explore the catalogue directly.

The platform's value therefore increasingly depends on its ability to turn a potentially infinite catalogue into a small set of content relevant to the individual user.

Search and recommendations become central components.
The interface must no longer simply answer: What did you search for?
It must also try to predict: What might you want to watch now?

This transformation prepares the way for the growing centrality of the personalised Homepage, Up Next (the section of YouTube that suggests which video to watch after the current one) and recommendation systems based on user behaviour.

Another important change concerns the gradual transformation of the creator.
The first creators often used webcams, home video cameras and extremely simple productions.
With the ability to monetise their audience, however, more complex structures begin to form.
A channel can evolve from an individual activity into a structure consisting of creator → team → studio → production company → media company
The distinction between amateur and professional content therefore becomes increasingly blurred.
YouTube does not simply replace television.
It enables new audiovisual players to emerge, initially existing only thanks to the platform.

In 2015, this transformation is accompanied by greater product differentiation.
In February, YouTube Kids is introduced: a version designed specifically for children, with a different experience and controls (Ben-Yair, 2015).

During the same period, mobile has now overtaken desktop: in July 2015, YouTube states that more than half of views take place on mobile devices (Darby & Amarilio, 2015).
Internet video is therefore no longer an activity primarily associated with the computer.
By further reducing the distance between production and distribution, the smartphone simultaneously becomes: camera + editor + upload device + player + social interface.

Also in 2015, YouTube Red is introduced, a subscription service offering, among other features, no advertisements, background playback and the ability to save content for offline viewing (Leske, 2015).
It is an important change for the economic model.

Until then, the model was mainly free access → advertising; now subscription → ad-free experience also appears
The two models begin to coexist.

In 2017, YouTube TV launches in the United States: a subscription service that distributes live television channels over the Internet (Jeon, 2017).
It is almost a reversal of its origins.
YouTube was born as an alternative to the logic of television: anyone can publish a video without going through a broadcaster.
Now the same infrastructure is also used to distribute linear television, sport and live events.
The platform thus begins to occupy territories that had historically been separate at the same time: user-generated video + professional creators + music + live streaming + television.

In 2018, YouTube Red is renamed YouTube Premium, and a new version of YouTube Music is introduced, built as a service specifically geared towards music consumption (YouTube, 2018).
This transition makes the ecosystem's multi-product nature even clearer.

The same Google account can move between different experiences that share part of the infrastructure, catalogue, data and personalisation systems.

Another fundamental transformation takes place in 2020.
YouTube introduces the first beta of YouTube Shorts in India, designed for short vertical videos and strongly mobile-oriented production (YouTube, 2022).
Shorts does not simply introduce a shorter format.
It changes the interaction model.

- In traditional YouTube: Homepage → Thumbnail → choice → Play
- In the Shorts feed: Play → immediate evaluation → Swipe → new video

The user does not necessarily need to decide explicitly what to watch.
It is the recommendation system that continuously selects the next piece of content.

The platform therefore moves from a model primarily based on selecting a video to one in which, in some contexts, algorithmic selection of the stream is central.

The same platform must thus support profoundly different experiences:

- Long-form → relatively deliberate choice.
- Shorts → continuous feed.
- Live → real-time event.
- Music → music listening.
- TV → consumption on the big screen.

These experiences use the same YouTube ecosystem but place different demands on recommendations, interface, advertising and infrastructure.
Meanwhile, the quantity of content continues to grow.

In 2025, on the platform's twentieth anniversary, YouTube states that it has exceeded 20 billion uploaded videos and receives an average of over 20 million new videos every day (Mahesh, 2025).

This scale helps explain a fundamental feature of YouTube's architecture.
The problem is not simply storing a large library.
It is managing a catalogue that never stops being built.
Every day, operations such as the following must therefore take place millions of times:

```
Upload
↓
Analysis
↓
Transcoding
↓
Rights checking
↓
Indexing
↓
Distribution
↓
Recommendation
```

In parallel, the way YouTube is watched continues to change.
The service, which originated mainly within the desktop browser, has also become increasingly important on television sets.

In January 2026, YouTube stated that it had ranked first for share of streaming watch time in the United States for almost three years according to Nielsen, referring to television viewing measured by The Gauge (Mohan, 2026).
In the same communication, Shorts recorded an average of 200 billion daily views. Views do not equal unique users and must not be confused with the engaged views used to allocate revenue (Mohan, 2026).

This produces an apparently paradoxical situation.
YouTube begins as an alternative to the traditional television model, based on short videos uploaded by users through the Web.

Twenty years later, it is simultaneously:

```
a creator platform
+
a music service
+
```

```
a live platform
+
a short-form feed
+
a television service
+
an advertising system
+
a subscription platform.
```

The historical transformation can therefore be summarised as follows:

```
2005 -- Video Sharing
↓
2006 -- Google / growth on a global scale
↓
2007 -- Creator Monetization + Content ID
↓
2007-2010 -- Mobile + HD + 4K
↓
2011 -- Live Streaming
↓
2013 -- Mass Platform + Recommendation
↓
2015 -- Kids + Subscription + Mobile First
↓
2017 -- YouTube TV
↓
2018 -- Premium + Music
↓
2020 -- Shorts
↓
2025-2026 -- Multi-format Global Media Platform
```

The history of YouTube can therefore be read as a continuous expansion of the initial problem.
In 2005, the main question was: How can a person be enabled to publish a video on the Web easily?

As the platform grows, new questions continually become necessary:

- How can millions of uploads be processed?
- How can they be distributed globally?
- How can relevant content be found within an enormous catalogue?
- How can creators be paid?
- How can copyrighted material be recognised automatically?
- How can events be broadcast live?
- How can video be adapted to smartphones, televisions and different connections?

- How can both free users and subscribers be monetised?
- How can the next video to show be decided automatically?

Each historical transformation therefore introduces a new layer into the architecture.

It is precisely this relationship between product evolution and technological evolution that makes YouTube particularly interesting to study: from a simple video-sharing service, it has become a global infrastructure capable of ingesting, processing, classifying, protecting, recommending, monetising and distributing a quantity of content that no traditional editorial team or catalogue could manage manually.

Business model

YouTube operates as a multisided platform: it connects different participants who derive value from one another's presence.
The main actors are:

- Creators: who produce and publish content.
- Viewers: who watch content, generate attention and interactions and, in some cases, pay for a subscription.
- Advertisers: who pay to reach particular audiences through advertisements.
- Rights holders: who control and monetise copyrighted works.
- Brands and merchants: who can use creators and content to promote or sell products.
- YouTube: which provides infrastructure, distribution, discovery, recommendations, advertising, measurement and monetisation systems.

The system can be represented in simplified form as follows:

- Creator → Content → YouTube → Viewer
- Advertiser → Money → YouTube → Advertising → Viewer.
- Viewer → Attention + Data + Subscription + Impressions → YouTube.

YouTube therefore sits at the centre of several economic flows.
The platform must simultaneously persuade:

- Creators to produce content.
- Viewers to use the service.
- Advertisers to invest money to reach that audience.

This structure generates an important network effect.

```
More creators
↓
More content
↓
More opportunities to find something interesting
↓
More viewers
```

```
↓
Greater value for advertisers
↓
More monetisation opportunities for creators
↓
Greater incentive to produce content
```

Content therefore represents the element that sets the whole system in motion.

The production investment is borne mainly by creators, publishers, companies, broadcasters and other participants that publish on the platform.
YouTube, in turn, provides the infrastructure needed to turn this content into a distributable and monetisable product.
This allows the platform to build a potentially enormous catalogue without having to buy every single piece of content directly.

However, the model requires a fundamental mechanism: part of the economic value generated must return to those who produce the content.

This is the basis of the **YouTube Partner Program**.
In 2025, YouTube stated that over three million creators earned money through its programmes and, in September of the same year, that it had distributed a total of over $100 billion to creators, artists and media companies over the previous four years (YouTube, 2025; Voolich, 2025).

Revenue sharing is therefore not merely a form of payment.
It is one of the mechanisms through which YouTube keeps its content supply active.
Conceptually:

```
Revenue
↓
Creator’s share
↓
Creator’s investment
↓
New content
↓
Audience
↓
New revenue
```

But where does this revenue come from?
Historically, YouTube’s most important source of revenue is advertising.
Many viewers can use the service without paying directly.
The economic price of free access is the possibility that their experience will include advertisements.

The fundamental relationship is:

- Viewer → Attention: the viewer brings the platform primarily time and attention: watching videos, remaining on the platform and interacting with content. This attention matters because it creates advertising opportunities and makes YouTube attractive to advertisers.
- Advertiser → Money: advertisers pay YouTube to show advertisements to particular audiences. In return, they gain access to advertising space, targeting and campaign measurement.
- YouTube → Audience + Advertising Inventory: YouTube offers advertisers two things: an audience to reach and a certain quantity of advertising inventory, meaning concrete opportunities in which an advertisement can be shown, for example before a video, during a long video or between two Shorts.

These opportunities are normally found:

- At the beginning of the video → Pre-roll Opportunity
- In the middle of a long video → Mid-roll Opportunity
- Between Shorts in the feed: Short → Short → Ad → Short → …

Advertisers do not simply buy “a YouTube video”.

They use Google’s advertising infrastructure to define campaigns, budgets, objectives, audiences and other conditions.
For each advertising opportunity, the platform must therefore determine:

- which campaign is eligible?
- which advertisement can be shown?
- how much is the advertiser willing to pay?
- which targeting and brand safety constraints must be respected?

Advertisers can use YouTube with different objectives.

- Performance-oriented campaigns seek a concrete, measurable action, such as a click, a website visit, an app installation or a purchase. In this case, the advertisement’s value depends primarily on its ability to produce a response from the user.
- Brand-oriented campaigns, by contrast, mainly seek to increase brand awareness and perception. An advertisement can therefore have value even when the user does not immediately take action: it may be enough for it to actually be shown and seen.

This distinction directly affects how advertising works on YouTube. When an advertising opportunity arises, the system must not only identify which campaigns are available, but also consider their objectives and the characteristics of the viewer and the context.

The same view can therefore have different values for different advertisers. A company interested in brand awareness may prioritise the ability to reach many people, while a performance-oriented advertiser may be more interested in the likelihood that the user will click, install an app or make a purchase.

YouTube must therefore turn every advertising opportunity into the most effective possible match between viewer, content, context and the advertiser’s objective.

YouTube does not necessarily retain all the advertising revenue generated by creators' content. Partners who accept the Watch Page Monetization Module can share in revenue generated by advertisements shown before, during, after or around their long-form videos and live streams. For these forms of monetisation, YouTube pays the creator 55% of applicable net revenue, retaining the remainder. However, this does not mean that the creator receives 55% of everything an advertiser spends: the percentage is applied to the net revenue that actually enters the revenue-sharing process (YouTube, n.d.-s).

Example
For example, greatly simplifying, if certain advertisements associated with a creator's videos produced €100 of net revenue to be shared, the model would be:

100 € net revenue
- 55 € creator.
- 45 € YouTube.

However, this does not mean that every €100 spent by the advertiser automatically becomes €55 for the creator. A distinction must be made between what the advertiser pays and what actually enters the revenue-sharing calculation.
Moreover, not all views necessarily generate advertising. A user might not receive an advertisement because no eligible campaign exists, because they use Premium, because of the content's characteristics or because of other session conditions. Therefore:

1 view ≠ 1 ad impression ≠ fixed revenue.

The path can therefore be outlined as follows:

```
Advertiser
↓
Advertising expenditure
↓
YouTube
↓
Advertisement shown alongside the creator’s content
↓
Revenue sharing
↓
Creator
```

In 2024, Alphabet reported $36.1 billion in revenue from YouTube advertisements, compared with $31.5 billion in 2023. However, this figure relates specifically to advertising and does not represent the entire revenue generated by all YouTube services (Alphabet Inc., 2025).

The Shorts advertising model works differently.
In long-form video, an advertisement can be associated with the watch page of a specific piece of content.
In the Shorts feed, by contrast, advertisements are shown between pieces of content.
For example:

```
Short A
↓
Short B
↓
Advertisement
↓
Short C
```

That advertisement therefore cannot simply be attributed solely to the preceding or following Short.
For this reason, YouTube uses a system based on a **Creator Pool**.

- Revenue generated by advertisements shown in the feed is aggregated.
- Part of the system must also account for music licensing costs.
- The Creator Pool is then allocated among monetising creators based on their share of eligible engaged Shorts views in each country; a 45% revenue share is applied to the amount allocated to each individual creator (YouTube, n.d.-u).

Conceptually:

```
Advertisements in the Shorts feed
↓
Advertising revenue pool
↓
Music licensing + Allocation to creators
↓
Creator’s share
↓
45% revenue share
```

The economic model therefore adapts to the structure of the interface.

- In long-form video, the content constitutes a relatively independent unit.
- In Shorts, the experience is instead a continuous feed, and monetisation is also built at feed level.

YouTube does not depend exclusively on advertising.
Subscriptions represent a second major component of the model.
The most obvious case is **YouTube Premium**, in which the user takes out a monthly subscription.
In return, the user receives an experience that includes features such as no advertising, background playback (video and audio playback in the background on mobile) and offline downloads (YouTube, 2018).
The creator can also be paid when the Premium viewer sees no advertising.
YouTube distributes part of subscription revenue to creators based on consumption of their content: the more a Premium subscriber watches a particular creator, the greater the share associated with that consumption may be (YouTube, n.d.-r).

The model becomes:

```
Premium subscription
↓
YouTube
↓
Viewing activity
↓
Revenue allocation to creators
```

This maintains the economic incentive to produce even in the absence of advertisements.

YouTube Music, in particular, is the music streaming service that uses YouTube's infrastructure
YouTube Music combines two modes:

- Free access → advertising.
- YouTube Music Premium → subscription.

YouTube Music Premium offers ad-free music listening, background playback and downloads within the YouTube Music experience, which also includes music videos. It does not automatically extend these benefits to all ordinary YouTube videos; furthermore, podcasts may contain advertisements (YouTube, n.d.-e).

In 2025, YouTube reported over 125 million YouTube Music and Premium subscribers globally, including free trials (Greenberg, 2025).

A different model again applies to paid films and TV programmes.
In some markets, including Italy for films, YouTube allows individual titles to be purchased or rented. This involves neither advertising nor a general subscription to the platform: the user makes a transaction to access a specific piece of content (YouTube, n.d.-a).
This model belongs to *TVOD* (Transactional Video on Demand):

```
User
↓
Rental / Purchase
↓
YouTube
↓
Film
↓
Revenue sharing with rights holders
```

There is a third economic relationship: fan funding, meaning forms of monetisation in which the viewer pays directly to support a creator or obtain particular benefits.
In this case, the money does not come primarily from an advertiser or a general platform subscription, but directly from the viewer who chooses to support a specific creator.

YouTube offers tools such as:

- Channel memberships.

- Super Chat.
- Super Stickers.
- Super Thanks.

Channel memberships involve recurring monthly payments in exchange for members-only benefits.

Super Chat and Super Stickers allow viewers to buy highlighted messages or graphical elements during live streams and Premieres, a feature that allows content creators to show a prerecorded video to an audience in real time, while Super Thanks allows financial support for the creator through a contribution associated with a video or a Short.

These features are enabled through the **Commerce Product Module**, one of the contractual modules of the YouTube Partner Program dedicated to forms of fan funding.
For these features, YouTube pays the creator 70% of applicable net revenue (YouTube, n.d.-s).

This form of monetisation changes the economic nature of the audience.
A viewer is not valuable only because they can see advertising.
They can also become a direct customer of the creator.
A relatively small channel with a highly engaged community could therefore generate significant economic value even without reaching the scale needed to produce enormous advertising volumes.

Another form of monetisation is commerce.
YouTube allows eligible creators to link products to their content through **YouTube Shopping**.
In this case, the path is:

```
Video / Short / Live stream
↓
Product tag
↓
Viewer’s click
↓
Seller
↓
Purchase
↓
Commission for the creator
```

In the YouTube Shopping affiliate programme, creators can associate their content with products sold by participating sellers. The seller sets a commission: a percentage of the sale value they are willing to pay when the creator helps generate that purchase (YouTube, n.d.-n).

The video thus performs two functions simultaneously: content, because it informs or entertains the viewer, and a commercial discovery tool, because it can guide them towards products they might subsequently buy.
For example, a photography tutorial can entertain or inform the viewer while also allowing them to buy the camera shown in the video.

It is important to distinguish this model from advertising sold directly by YouTube.
A creator can independently enter into a commercial agreement with a brand.
In this case, the creator can integrate a sponsorship directly into their video.

This transaction does not necessarily constitute advertising revenue for YouTube.
It is therefore necessary to distinguish:

- YouTube Ads → Google's advertising platform.
- Creator Sponsorship → commercial agreement between brand and creator.

YouTube is developing tools that also facilitate brand partnerships, but economically the two mechanisms remain different.
This matters because economic activity connected to the YouTube ecosystem also includes revenue earned by participants outside the platform. Its scope is broader than YouTube's revenue, and it is not a quantity that can be obtained by indiscriminately adding together revenue, sales and economic impact.
A creator can use the audience built on YouTube to generate revenue through:
Advertising + Premium + Memberships + Super Chat/Stickers/Thanks + Shopping + Sponsorships + External businesses.

Rights management also performs an economic function.
However, **Content ID** should not simply be considered a separate revenue source.
Content ID associates audio and video matches with rights holders' claims and policies. It does not establish legal ownership of a work on its own: claims can be incorrect and disputed (YouTube, n.d.-f).
When a match is found between an upload and a work registered in the system, the rights holder can, depending on their policies:

- Block: the video is blocked.
- Track: the video remains available, while the rights holder can monitor viewing statistics without blocking it or monetising it directly.
- Monetise: a view that initially belonged to a video uploaded by a user can generate revenue for the rights holder.

Content ID therefore makes it possible to transform copyright management from a simple removal system into a system for the economic allocation of rights.
This is particularly important for music, television and other professional content.
Another important point is that policies can be geographical: the same video may, for example, be monetised in Italy, tracked in France and blocked in another country, depending on the rights holder's settings (YouTube, n.d.-f).

Recommendation does not directly produce money.
However, it is fundamentally important to the platform's economic model.
YouTube has an enormously larger quantity of content than any user could explore.

The economic problem is therefore not just having content, but successfully matching each piece of content with a potentially interested audience.

If a user continually finds relevant content, their engagement will probably increase and they will be more likely to return to the platform.

```
Relevant content
↓
Viewing
↓
Satisfaction / Engagement
↓
Return to the platform
```

This can produce different consequences.

- For a free user: More relevant views → More advertising opportunities.
- For a Premium user: Greater perceived value → Greater subscription retention.
- For the creator: Greater discovery → Larger audience → More monetisation opportunities.

However, this does not mean that the system simply seeks to maximise watch time at any cost.
YouTube uses different signals of satisfaction, quality and behaviour to build its recommendations.
From an economic perspective, however, effective search can increase the platform's ability to turn its enormous catalogue into actual usage.

We can therefore describe YouTube's overall model as a cycle.

```
Creators / Rights holders
↓
Content
↓
YouTube infrastructure + Discovery
↓
Audience
↓
Attention + Subscriptions + Purchases
↓
Revenue from advertising + Subscriptions + Commerce
↓
Revenue for YouTube + Creators / Rights holders
↓
New content
↺
```

The model's most important feature is therefore that YouTube does not simply sell videos.

It sells and coordinates different forms of access to the value generated by the encounter between content, audiences and commercial activities.

- It offers viewers access to content.
- It offers creators distribution, audiences and monetisation tools.
- It offers advertisers access to measurable audiences.
- It offers rights holders tools for control and monetisation.
- It offers brands and merchants access to creators' economic influence.

In return, it retains part of the economic value produced by these relationships.
The overall model can therefore be summarised as:
Content → Audience → Monetisation → Revenue sharing → More content

This cycle is probably the heart of YouTube's business model.

Discovery System

One of YouTube's main problems, as we have already seen, is the quantity of content available.

The platform contains too many videos for a user to explore directly. The problem is therefore not only storing and distributing content, but successfully matching each piece of content with a potentially interested audience.

YouTube must continually answer a question:
Which video, among millions of possible candidates, should be shown to this particular user, at this particular moment and in this particular position in the interface?

This creates a fundamental relationship between platform and creator.
The creator produces and publishes the content, but YouTube's discovery systems help connect it with an audience.
The process can be represented as follows:

```
Creator
↓
Video
↓
YouTube discovery systems
↓
Potential audience
↓
Viewer response
↓
New signals
↺
Future recommendations
```

To understand this system, it is important to change perspective.
The system is less concerned with “promoting a video to an audience” than with finding the most suitable video to suggest to a viewer when they use YouTube.

The system is therefore designed around the viewer.
The creator tends to ask: “How can I persuade the algorithm to promote my video?”
A more appropriate question would be: “For which viewers might my video be a good recommendation?”

YouTube describes this principle by essentially saying that the recommendation system follows the audience: it observes what viewers choose, watch and appreciate, and tries to identify other content potentially suited to them (YouTube, n.d.-g).

All this reveals a fundamental aspect of how YouTube works: publishing a video and making it visible are not the same thing.
Uploading makes the content available on the platform, but its actual ability to reach an audience depends on the various discovery systems.
The creator cannot directly decide to whom a video will be shown, but can design the content to make it easier to understand and more relevant to its intended audience.
YouTube continually observes viewers’ responses and uses these signals to update its predictions:
Content → Audience → Viewer response → New signals → New recommendations

YouTube is therefore not just a platform on which videos are uploaded and played. It is also a system for the algorithmic distribution of attention, continually seeking to connect content and viewers.

But how does YouTube recommend new content?
To describe the process, the two-stage model documented by YouTube in 2016 is useful: candidate generation and ranking. It is a historical and educational reference, not a complete specification of the current architecture. The first problem is to reduce the enormous catalogue to a much smaller set of candidate videos (Covington et al., 2016).
To do this, YouTube uses the available information about the viewer, such as watch and search history, channels followed, previous interactions, preferred topics and formats, and the behaviour of other users with similar interests (YouTube, n.d.-g).

If, for example, someone frequently watches videos about Blender (graphics software) and animation, the system can consider new videos from the channels they follow, content on the same topics and videos appreciated by other viewers with similar behaviour.

At this initial stage, YouTube does not yet need to establish which is the best video to show. It mainly needs to identify a set of content with a plausible reason to interest that particular viewer.

The process can be summarised as follows:
Billions of videos → Viewer profile → Candidate retrieval → Small set of videos

Only afterwards does ranking come into play.

Once a relatively small set of plausible content has been obtained, YouTube must determine the order in which to present it.
Ranking therefore seeks to estimate, for each candidate, how well that video suits that particular viewer at that particular moment.
The video is not simply assigned an absolute value.
Rather, the question is:
“How likely is this user to choose this video, continue watching it and be satisfied with it?”

YouTube groups the main signals relating to content performance into three broad categories (YouTube, n.d.-p):
Appeal → Engagement → Satisfaction

Appeal concerns the content’s ability to be chosen when it is offered to the viewer.
It depends primarily on how interesting the topic is to that particular audience and on how the video communicates its value through elements such as the title and thumbnail (the preview image used to convey the content quickly).
Improving Appeal does not mean resorting to clickbait, but making the content’s promise immediately understandable and interesting.

We can summarise it with a question: “Does this video interest me enough to start watching it?” (YouTube, n.d.-i)
A video can increase its Appeal through:

- A topic that genuinely interests the audience to whom it is offered.
- A clear title that communicates the content’s value.
- An easily understandable thumbnail.
- An interesting promise that is consistent with what the video will actually deliver.

Once the viewer has chosen the video, Engagement comes into play: the video’s ability to hold attention and generate involvement.
The main question becomes: “Once I have started, do I keep watching?”
The central Engagement signal therefore concerns viewing behaviour: how long the viewer stays with the content, what percentage of the video they watch and at which points they tend to leave, skip or rewatch a section.
A video can improve its Engagement by immediately delivering on the promise communicated by its title and thumbnail, removing redundant parts, creating a narrative or informational progression and delivering value throughout its duration.
Alongside continued viewing, there are complementary forms of interaction, such as comments, likes and shares. These actions help creators describe engagement, but do not constitute an automatic score: the presence of a metric in Analytics does not imply that it is a direct ranking factor, nor does it reveal its weight.

The creator can observe these behaviours through audience retention curves (a graph showing, along the timeline, what percentage of viewers are still present at each point in the video), identifying where viewers continue watching, leave the video, skip parts or go back (YouTube, n.d.-d).

Two particularly useful metrics are average view duration and average percentage viewed.

Average view duration and average percentage viewed describe different aspects of consumption. For Shorts, YouTube states that it uses both among its ranking signals, together with the choice to watch the content; neither constitutes a universal formula on its own (YouTube, n.d.-l).

To improve Engagement, it is therefore useful to:

- Immediately deliver on the promise of the title and thumbnail.
- Reduce unnecessarily long introductions and preambles.
- Create a progression that gives the viewer a reason to continue.
- Deliver value throughout the video.
- Remove redundant sections or mere filler.
- Analyse retention to identify drop-off points, skips and particularly interesting sections.

The third element is **Satisfaction**: how positive the viewer's experience was.
The question becomes: "At the end, am I glad I watched this video?"

A video can hold the viewer's attention for a long time without necessarily being satisfying. Watch time alone is therefore not enough to describe the quality of the experience. YouTube therefore tries to estimate satisfaction using different signals, such as likes, dislikes, shares, negative or positive feedback and satisfaction surveys (Goodrow, 2021).
A video can improve Satisfaction by:

- Delivering on its initial promise.
- Offering a useful, interesting, moving or rewarding outcome.
- Avoiding clickbait and expectations that the content cannot fulfil.
- Reaching a conclusion consistent with what was promised.
- Respecting its audience's expectations and the level of detail they require.
- Spontaneously generating positive signals such as likes and shares.

We can therefore summarise the three concepts as follows:

- Appeal → Does it persuade me to start?
- Engagement → Does it persuade me to continue?
- Satisfaction → At the end, am I glad I watched it?

However, these signals are not evaluated in isolation.
They are combined with viewer personalisation and the context in which the recommendation occurs.
The context can include elements such as the device used and the user's previous habits; the 2016 paper documents the use of contextual features. The following examples illustrate the principle of personalisation, not scores or weights published by YouTube (Covington et al., 2016).

Let us return to the example of a viewer interested in Blender and animation.
After the selection stage, four candidates might remain:

- Video A: advanced Blender tutorial.
- Video B: GPU review.
- Video C: animation created with Blender.
- Video D: Photoshop tutorial.

For a user who often watches Blender videos, tends to complete technical tutorials and has recently searched for “Blender rigging”, the system might assess:

- Video A → high probability of being chosen and providing satisfying viewing.
- Video C → good thematic affinity.
- Video B → possible but less direct interest.
- Video D → weaker affinity.

The ranking might therefore produce:
A → C → B → D

For another viewer, the same four pieces of content might be ordered completely differently.

Ranking therefore does not simply measure the video’s absolute quality, but its predicted relevance to a specific viewer and a specific context.

YouTube can suggest new content in different situations:

- Home → “What might interest you now?”
- Up Next → “What might you want to watch after this video?”
- Search → “Which content best answers your search?”
- Shorts Feed → “Which Short should I show you immediately afterwards?”

These are different problems and therefore do not necessarily use the same signals in the same way.

Home
YouTube describes Home as a highly personalised surface.
It can contain videos from the user’s subscriptions, new content and videos appreciated by viewers with similar interests.
Signals that may contribute to selection include watch and search history, the video’s performance among viewers with similar tastes, how frequently the user watches a particular channel or topic, and how many times a particular piece of content has already been shown to them (YouTube, n.d.-g).

This has a fundamental consequence: two users can open YouTube at the same time and see completely different Home pages.
There is therefore no single YouTube Home page.
There are personalised Home pages for individual viewers.

Up Next
One of the most important signals in this context is the video currently playing.
YouTube looks for content that the viewer might want to watch immediately afterwards, often related to the current video but also personalised according to the user’s history and interests (YouTube, n.d.-g).

A video therefore does not necessarily compete with all the content on YouTube. It may primarily compete to become a good continuation of a particular viewing experience.

Search
When the user uses Search, the situation changes again because they directly express an intention through a query.

YouTube describes three broad categories of factors:
Relevance + Engagement + Quality

Relevance indicates how closely a piece of content matches the search performed.
YouTube can estimate it by considering the match between the query and different elements of the video, such as its title, description, tags and actual content (YouTube, n.d.-h).

A video therefore increases its relevance when there is strong consistency between what the user is looking for and what it actually offers.

The process can be represented as follows:

```
Query
↓
Consistent title
↓
Consistent description
↓
Genuinely relevant content
↓
User behaviour consistent with the query
↓
Greater Relevance
```

YouTube does not consider metadata alone.
The video's actual content must also be consistent with the search.

- If a video genuinely addresses the topic of the query, it has strong semantic consistency.
- If, instead, it only uses those words superficially, the consistency is much weaker.

Tags can contribute to relevance, but today their importance is relatively limited, and they are particularly useful, for example, for handling spelling variants or terms that are easily misspelled (YouTube, n.d.-p).

Relevance must therefore not be confused with the mere presence of keywords.
The real question is: "Does the content answer the intent that led the user to perform that search?"

YouTube also observes the behaviour of users arriving from a particular query.
If, for example, users searching for "Blender character rigging" frequently choose a particular video and watch it for a long time, that behaviour can provide further confirmation that the video is a good result for that search.

Relevance and Engagement are therefore not completely independent (YouTube, n.d.-h).

User behaviour can also become evidence of how well the Query and Video match

Engagement in Search therefore indicates how much users arriving from a particular search actually interact with that content.
It is very similar to the concept of Engagement discussed earlier, but the context changes: here, behaviour is also observed in relation to a specific query.

We can therefore have:

```
Query
↓
Video shown in results
↓
Click and viewing
↓
Engagement in relation to the query
↓
Further confirmation of Relevance
```

The third element is **Quality**.
In this case, the term does not simply mean how technically well made the video is.
Rather, the question is: “Is this source credible when discussing this topic?”

YouTube therefore seeks signals that help identify the channel’s expertise, authoritativeness and trustworthiness in relation to the subject area covered.
This factor is particularly important for topics such as news, politics, medicine and science, where the source’s credibility is especially relevant (YouTube, n.d.-h).

A creator can build greater authority over time through:

- Subject expertise and content accuracy.
- Consistency in covering a specific domain.
- Reliability of the information provided.
- Use of trustworthy sources.
- A recognisable reputation of the source or author.

Authority should not be considered an absolute characteristic of the channel, but a property connected to a specific area of expertise.
For example, a creator may be considered highly authoritative when discussing Blender but lack the same credibility when addressing a medical topic.
It is therefore more accurate to think of authority as a relationship between:
Source + Domain → Authority on the topic

Popularity must also be distinguished from authority. A channel with a large following, many subscribers and many views is not automatically a competent or reliable source on every topic. Popularity and authority can coincide, but they describe different properties.

For music, entertainment and topical subjects, popularity and timeliness can also help interpret audience interest.

- **Freshness indicates how recent and timely content is in relation to the search performed. Its usefulness depends on the viewer's needs: it can be crucial for an update on an event, whereas an evergreen tutorial can retain value for a long time. This distinction is an interpretative criterion, not a table of algorithm weights.**
- Popularity describes how widely known and followed a piece of content or an author is. Views and subscribers can help describe it, but they are not a public ranking formula. YouTube's documentation instead distinguishes relevance, engagement and quality, whose importance varies according to the search (YouTube, n.d.-h).

Popularity and freshness are not the same:
- Popularity → How widespread is the content, and how much does it interest the audience?
- Freshness → How recent and current is the content?

Content can therefore be very popular but not very recent, or very recent but not yet very popular.

The weight of the different factors therefore changes according to the query.
For example:
- Medical search → authority and trustworthiness become highly important.
- Breaking news → authority and freshness can become fundamental.
- New film trailer → the official source and freshness can be particularly relevant.
- Blender tutorial → relevance, Engagement and subject expertise may carry greater weight.

Shorts Feed
In Shorts, the relationship between the viewer and the recommendation system changes again.
With traditional video, we often have: Thumbnail → Click → Video
With Shorts, instead: Video → Viewer reaction → Swipe or continued watching → Next video

The content is therefore shown before the viewer even has to explicitly choose to click on it.

This also changes the meaning of Appeal.
In the traditional format, the title and thumbnail must persuade the user to start watching.
With Shorts, instead, the system can directly observe the viewer's reaction in the very first moments through signals such as:
- Watching the video.
- Skipping the video.
- Rewatching the video.
- Liking the video.
- Sharing the video.
- Subscribing to the channel.

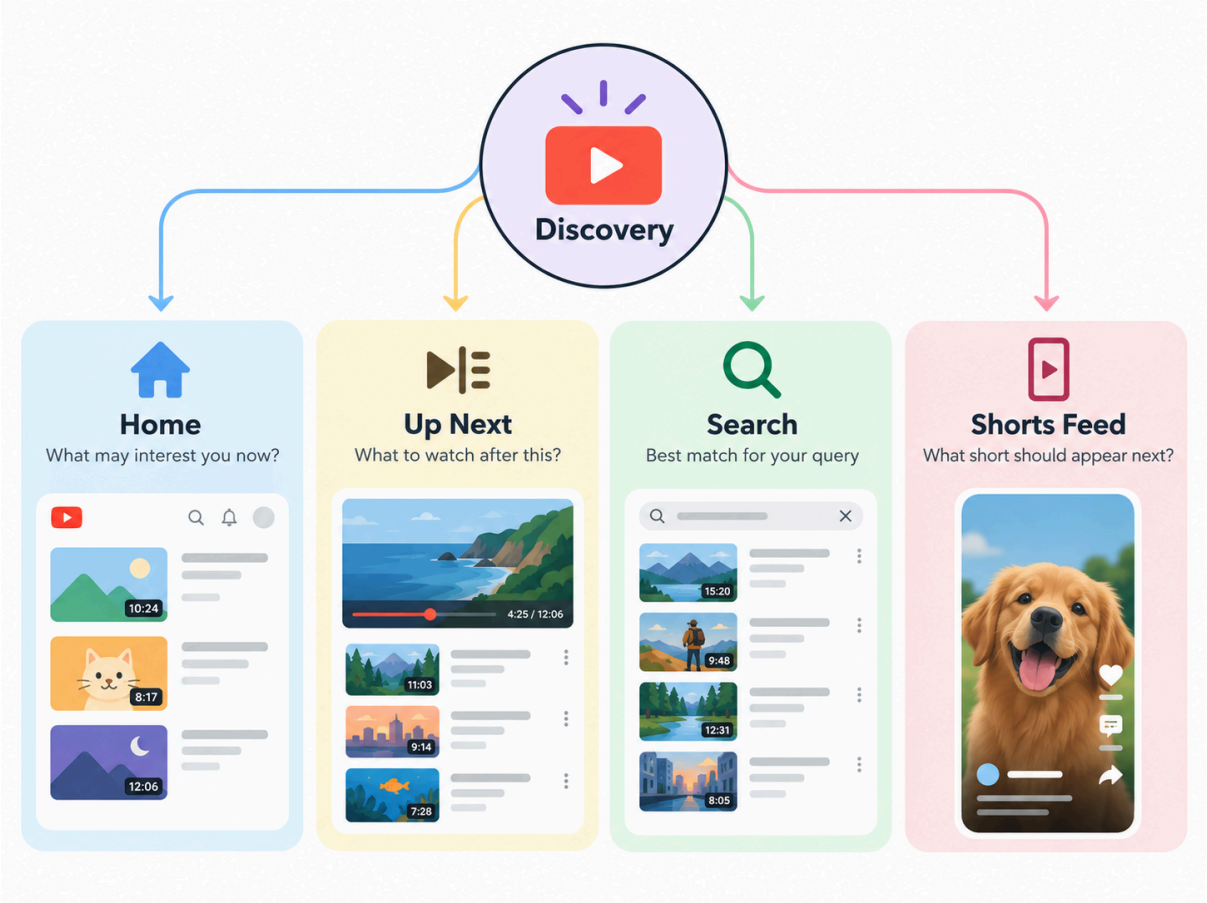
The documentation on Shorts ranking specifically mentions the choice to watch rather than ignore the content, average view duration and percentage viewed, likes and post-watch

surveys. The interactions listed above describe the user's experience without implying that every action has a declared ranking weight (YouTube, n.d.-l).

The cycle therefore becomes:

```
Candidate Short
↓
Viewing
↓
Watch / Swipe / Rewatch / Interaction
↓
New signals
↓
New recommendation
↺
```

The Shorts Feed therefore represents an example of continuous feedback: viewers' responses can contribute to subsequent recommendations. The diagram is educational and does not imply that every interaction immediately updates the system's models.



The operation of the Discovery System therefore reveals a fundamental feature of YouTube: publishing a video and making it visible are not the same thing.

Uploading makes the content available on the platform, but its actual ability to reach an audience depends on the different systems through which YouTube organises and distributes attention: Home, Up Next, Search and Shorts Feed.

In each of these contexts, the platform tries to determine which content may best suit a specific viewer. The creator cannot directly decide to whom a video will be shown, but can design the content to make it easier to understand, relevant and satisfying for its intended audience.
This structure also creates a need for creators to optimise how their content is presented and described. This set of practices is commonly known as **YouTube *SEO*** (Search Engine Optimization).
YouTube *SEO* primarily concerns making a video easy for the system to understand and relevant to users' searches. It can include, for example:

- Choosing a clear title consistent with search intent.
- Writing a relevant description.
- Using keywords appropriately and tags when useful.
- Choosing topics that people actually search for or that are relevant to the intended audience.
- Building content that is genuinely consistent with what the title and description promise.

However, *SEO* must not be interpreted as a set of tricks that can automatically guarantee a good ranking.
Optimising the title, description and metadata can help YouTube understand what the video is about and for which searches it might be relevant, but viewers' actual behaviour remains fundamental.
A video that is perfectly optimised in terms of keywords, but ignored by users or quickly abandoned, is unlikely to be considered a good result (YouTube, n.d.-h; YouTube, n.d.-p).
This is why *SEO* represents only part of the problem.

- In Search, the creator must work primarily on Relevance: the match between what the user is looking for and what the video offers.
- In recommendations, the following become particularly important:
    - Appeal → persuading the viewer to start watching.
    - Engagement → persuading them to continue.
    - Satisfaction → ensuring that they consider the experience positive.

We can therefore summarise the creator's work through four complementary objectives:

- SEO → Being found.
- Appeal → Being chosen.
- Engagement → Being watched.
- Satisfaction → Being appreciated.

Visibility on YouTube therefore does not depend on a single parameter, but on the combination of the platform's understanding of the content, audience interest and viewers' actual responses.

The creator’s task is therefore not to “persuade the algorithm”, but to build content that the system can understand and that, above all, is genuinely relevant to the people to whom it is offered.

Architecture

YouTube’s architecture starts from a condition profoundly different from that of a platform with a controlled catalogue.
Netflix, for example, normally knows in advance which content must enter its catalogue and can prepare it through a relatively controlled pipeline.
YouTube, by contrast, must be ready to receive content continuously from millions of different participants:
Creator → Upload → YouTube

The file may be a short video recorded on a smartphone, a two-hour tutorial, a professional music video, a podcast, a vertical Short or a recording of an event.
The platform therefore cannot manually plan the preparation of every piece of content.
It must build a pipeline in which most operations happen automatically.
In simplified form, the architecture can be represented as follows:

```
Content ingest
↓
Content processing and transcoding
↓
Rights management and content analysis
↓
Metadata and indexing
↓
Control and discovery
↓
Playback and monetisation orchestration
↓
Media content distribution
↓
Client-side playback
↓
Data and feedback
↺
```

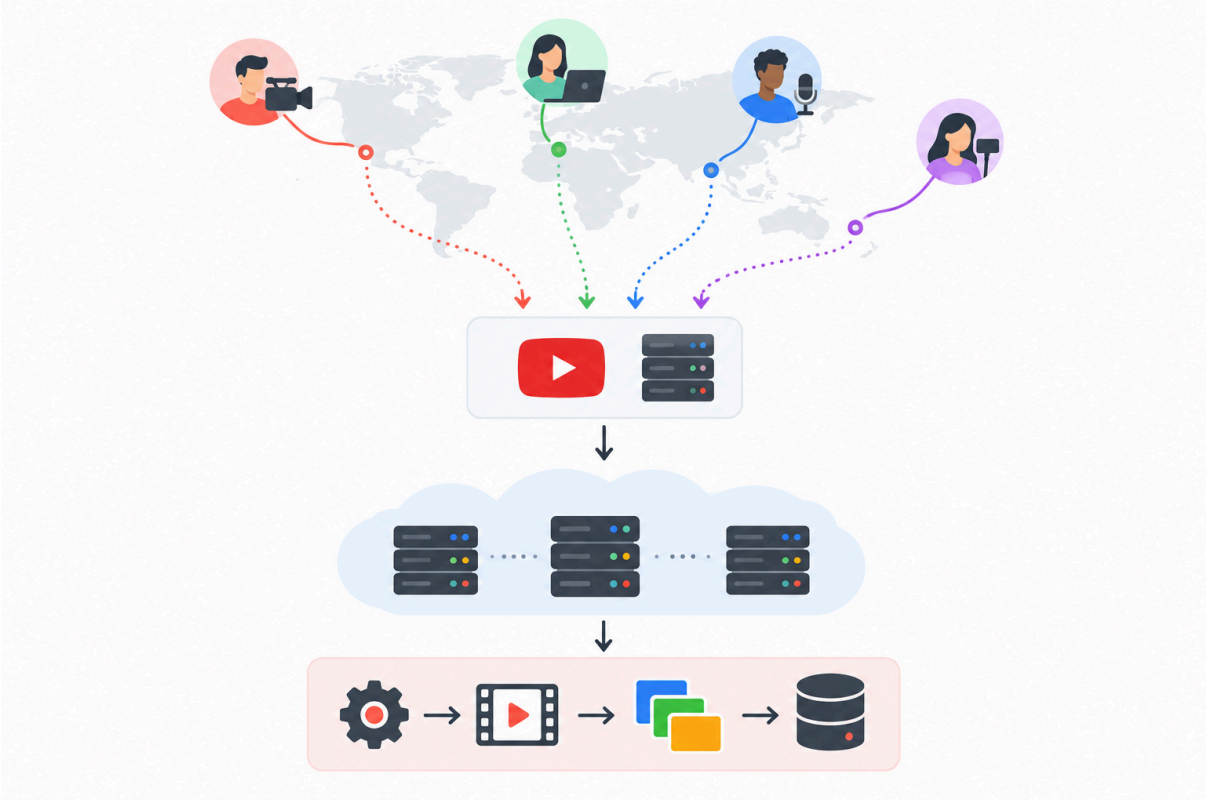


These blocks should not necessarily be imagined as a rigid sequence.
Many operations can take place in parallel.
While new versions of the video are being generated, for example, other systems can analyse its audio, check rights, process metadata or generate elements needed for publication.
While new versions of the video are being generated, other systems can perform analyses and prepare information for publication. The order of the blocks is a functional simplification: public sources describe individual components, but do not publish the complete topology of the internal pipeline (YouTube, 2021).

The first stage is ingestion, meaning the entry of content into YouTube's infrastructure.
When the creator uploads a video, the platform receives a source file accompanied by information such as the title, description, visibility settings and other metadata.
From the user's perspective, uploading takes place through a logical YouTube endpoint.
Behind this access point, Google uses a distributed infrastructure capable of receiving and managing large quantities of uploads from around the world.
The received file is not necessarily ready to be distributed to viewers yet. Once the transfer is complete, the content enters YouTube's processing pipeline, where subsequent processing and transcoding operations are performed.
The process can be represented as follows:

```
Creator worldwide
↓
YouTube Upload Endpoint
↓
```

```
Distributed Google Infrastructure
↓
YouTube Processing Pipeline
```



YouTube explicitly distinguishes between uploading and processing.
The YouTube Data API distinguishes different fields: status.uploadStatus can take values such as uploaded, processed, failed, rejected and deleted; processingDetails.processingStatus instead describes processing with processing, succeeded, failed or terminated. These are not states in a single enumeration; processing details are accessible to the video's owner (Google, n.d.-c).
This distinction is important architecturally because it separates two different operations:

- The first stage consists of correctly transferring and registering the file within YouTube's infrastructure.
- The second consists of processing that file and turning it into content that can actually be distributed to viewers.

The transcoding stage also takes place progressively.
YouTube can initially make a lower-quality version available and then continue processing the higher-resolution representations.
For this reason, after a 4K video has been uploaded, only lower resolutions may initially be available (YouTube, n.d.-j).
The process can be represented as follows:

```
Upload complete
```

```
↓
Initial processing
↓
Low-quality playback available
↓
Processing of higher qualities continues
↓
Additional representations become available
```

This strategy allows content to become playable without necessarily waiting for the entire processing pipeline to finish.

But why does YouTube generate different versions of the same video?
Viewers may use smartphones, tablets, computers and Smart TVs with very different screens, decoding capabilities and connections.
An unstable mobile connection, for example, cannot necessarily sustain the same stream as a Smart TV connected to a high-speed connection.
For this reason, YouTube uses transcoding: it generates different representations of the same content.
The same video can therefore be made available through different combinations of resolution, bitrate and codec, producing representations suited to different playback conditions (YouTube, 2021).

The file provided by the creator and the streams subsequently used for playback are therefore two different problems.
The creator may, for example, upload a video encoded in H.264, a codec recommended by YouTube for uploading, but this does not mean that the same file is simply delivered to all viewers (YouTube, n.d.-t).

During content preparation, YouTube can generate representations using different codecs according to the characteristics of the video and the playback devices. For example, playback of 4K content requires devices or browsers compatible with compression technologies such as VP9 (YouTube, n.d.-b).

The amount of computation needed to transcode millions of pieces of content per day becomes enormous.
Video encoding is therefore not simply a function of the player: it is an important distributed computing problem in data centres.
Google has also documented the development of specialised hardware called the Video Coding Unit (VCU), designed to accelerate video transcoding workloads at data-centre scale (YouTube, 2021).

In the educational model of adaptive streaming, after transcoding, the representations are prepared for progressive requests from the player: this is the function of packaging.
Representations can be organised into separate segments or portions of resources retrievable through byte-range requests.

A manifest can describe the representations and how to access the segments. This is the general model of adaptive HTTP streaming: it does not imply that every YouTube client always receives a separate manifest or the same combination of protocols.

For the purposes of the diagram, we call the infrastructure that stores and makes the prepared representations available the origin. Google documents storage in data centres and replication of content between locations; the exact organisation of YouTube's internal objects and services is not entirely public.

- Distributed means that content does not depend on a single server or a single data centre. Representations and segments can be managed by multiple systems located at different points in the network. This makes it possible to spread the load, reduce the distance from delivery systems and serve an enormous number of requests simultaneously.
- Redundant, by contrast, means that the infrastructure does not depend on a single copy or component. If a server, a storage system or an entire node becomes unavailable, another replica or another path can continue to provide the content. Redundancy therefore serves primarily to ensure reliability and continuity of service (Google, n.d.-b).

However, the technical preparation of the video is not enough.
YouTube must also understand what has been uploaded so that it can be indexed, and which rules must be applied.

One fundamental rule to apply is copyright.
As we have already seen, YouTube cannot manually check every uploaded video.
For this reason, it uses **Content *ID***.
Rights holders can provide reference audio or video files. When a new video is uploaded, Content ID automatically compares it against the database of registered works (YouTube, n.d.-f).

- If no match is found, the process continues normally.
- If a match is found, however, the policy can determine, as we saw in the previous chapter:
    - That the content is blocked.
    - That the content is tracked.
    - That the content is monetised.

But automated analysis is not limited to copyright.
YouTube can apply various processing systems to content. An easily observable example is the speech recognition used to generate captions automatically.
Another feature is automatic dubbing (auto dubbing). YouTube can analyse the original speech, translate it into another language and generate a new dubbed audio track that the viewer can select instead of the original audio (YouTube, n.d.-q).
The process can be represented in simplified form as follows:

```
Original audio
↓
Speech recognition
↓
```

```
Transcription
↓
Translation
↓
Speech synthesis
↓
New audio track
```

In some languages, YouTube also uses Expressive Speech systems, which seek to preserve features such as the tone and intonation of the original voice, making the dubbing more natural (YouTube, n.d.-q).

We can therefore think of a broader Content Intelligence layer, in which different systems can extract information from the content:

```
Audio / Video / Text / Metadata
↓
Automated analysis
↓
Derived information
```

However, we should not imagine that every possible analysis is necessarily performed in the same way on every piece of content.
The architectural point is that YouTube does not merely hold the video file: it also builds a set of derived information that other systems can use.

A video must then become something that a discovery system can find.
The audiovisual file alone is not enough.
YouTube has two types of information:

- Provided by the creator, such as title, description, tags, language, channel and visibility.
- Produced or observed by the platform, such as information about content, rights, viewer behaviour and other signals.

When a user performs a search, YouTube therefore does not need to analyse billions of video files from scratch.
It must be able to query structures that allow plausibly relevant content to be retrieved quickly.
The processing and indexing architecture therefore enables the search and recommendation systems described in the previous chapter to operate.
At this point, it is useful to distinguish two broad categories of functions.

- Control & Discovery: on one side are the systems that must decide what to show and under which conditions => The control plane.
- Media Delivery: on the other are the systems that must actually transport audiovisual data to the viewer => The media plane.

The Control & Discovery layer manages information such as account management, user profile management, subscription management, search and recommendation management, rights management, the availability of content for a specific user in a specific context, and monetisation.

When the user opens Home, selecting the content to show is logically distinct from transferring video data; automatic previews and preloading can nevertheless initiate media requests.
The platform must determine “Which content should be shown?”

The Discovery system retrieves candidates and ranks them based on the viewer’s profile and context.
When a search is performed, when the user is on a video page for Up Next, or when a Short is viewed in the feed, YouTube makes an automated, personalised editorial decision.

When the viewer chooses a video, the problem changes.
Knowing which content they want to see is not enough.
YouTube must build a playback session consistent with numerous conditions.

For example, the system must be able to distinguish between:

- Free Viewer: the experience may include advertising opportunities.
- Premium Viewer: the experience may be ad-free.

Similarly, a Content ID policy might allow a video in one territory and block it in another.

The playback request must therefore be interpreted within a specific context consisting of these parameters:

- Viewer.
- Video.
- Rights.
- Region.
- Account type.
- Playback device.
- Monetisation and advertising conditions.

For a free user, the advertising system may also come into play.
The audiovisual content and the advertisement are conceptually two different assets.
The advertising system must therefore coordinate with playback without being confused with the video file itself.
This distinction becomes even clearer in Shorts, where the advertisement belongs to the flow of the feed and not necessarily to an individual Short.

Once what should be played has been determined, the audiovisual data must be delivered to the viewer.
Serving every request directly from a single data centre would be inefficient.
Geographical distance would increase latency, network backbones would have to carry enormous amounts of repeated data, and a few central systems would become bottlenecks.
Video distribution therefore uses a *CDN*.

Google documents a network made up of data centres, points of presence and Google Global Cache edge nodes, also installed within operators' networks. These nodes temporarily store popular content, including YouTube videos, close to users (Google, n.d.-b).

Conceptually, we can represent distribution as:

```
Distributed Media Origin
↓
Google Global Network
↓
Edge / Cache Infrastructure
↓
Viewer
```

A cache temporarily stores copies of content close to users.
If millions of people are watching the same video, the same data does not need to be retrieved from the more distant source every time.

We can have a request to the edge cache:

- If the content is present: Cache Hit → delivery
- If it is not present: Cache Miss → request to the origin → Cache → delivery

This behaviour statistically reduces latency and increases the capacity to serve large numbers of viewers simultaneously.

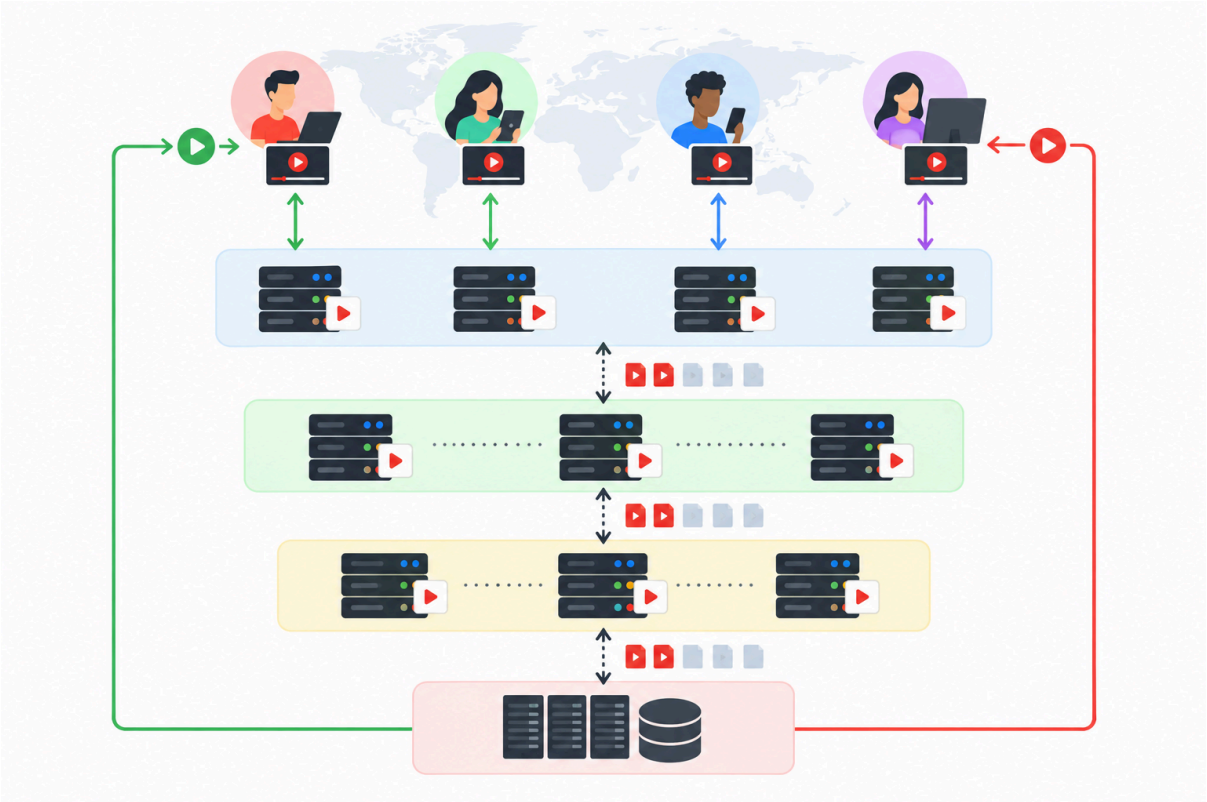
Google Cloud documents a multilevel caching hierarchy for Media CDN. We use it as an example of the principle, not as evidence that every YouTube request follows exactly the same topology as the Cloud product. The levels described are (Google Cloud, n.d.):

- Deep edge cache.
- Peering edge.
- Long-tail caches, also used as origin shields (origin shields are intermediate caches placed in front of the origin, primarily to protect it from excessive requests).

The principle is that a cache that does not have a segment does not necessarily have to request it directly from the origin.
The request can progressively move up through cache levels with increasing capacity.

```
Viewer
↓
Edge Cache
↓ cache miss
Peering / Mid-tier Cache
↓ cache miss
Long-tail Cache / Origin Shield
↓ cache miss
Origin
```

If one of the intermediate levels already has the segment, it returns it without involving the origin. Only when the content is unavailable throughout the entire hierarchy does the request reach the origin infrastructure.
The retrieved segment can then be stored in the caches, allowing subsequent requests to be served by nodes progressively closer to viewers.
The multilevel structure therefore makes it possible to reduce latency, network traffic and load on the origin simultaneously.



The final element in the chain is the client.
It can be a browser, a smartphone, a tablet, a smart TV, a console, etc…
The client does not necessarily receive a single unchanging file.
It has access to different representations of the same content and can choose the one most appropriate to current conditions.
The principle is *ABR*.
The player continuously observes variables such as available bandwidth, buffer, device capabilities and current representation, and can change quality during playback (YouTube, n.d.-b).

For example:

- Good connection: 1080p
- If bandwidth decreases: Bandwidth drop → Switch to 720p
- Bandwidth recovery: -> switch to a higher-quality representation.

The goal is not always to use the highest possible quality.
The player must find a balance between quality and stability.

- Quality that is too high for the available bandwidth can cause rebuffering.

- Quality that is too low, on the other hand, wastes network capacity and unnecessarily reduces perceived quality.

The client must also:
- Receive the data.
- Maintain the buffer.
- Decode audio and video.
- Synchronise them.
- Render them on the device.

YouTube uses DRM systems, particularly Widevine, the DRM system developed by Google and also used by Netflix, Disney+ and other streaming platforms (Google, n.d.-d).
However, it does not necessarily use this system on all videos. In particular:
- Ordinary videos uploaded by users: generally without DRM, but there are exceptions.
- Paid films and premium content: may be protected through DRM.
- Music videos and copyrighted content: not necessarily; copyright does not imply the use of DRM.

Distribution therefore does not end when the data reaches the viewer's smartphone or device.
It ends when the system succeeds in transforming that data into stable, continuous audiovisual playback.
During playback, YouTube also continuously collects new data. This can concern at least three broad dimensions.
- The first concerns viewer behaviour: how the viewer interacts with the platform. For example: starting or pausing playback, continuing to watch, skipping content, liking, commenting, sharing, performing a search or moving to another video.
- The second concerns Quality of Experience: the technical quality of playback. Start-up time, rebuffering, errors, quality and stability are examples of useful parameters for describing it. YouTube makes diagnostic information observable through Stats for nerds, including bandwidth and buffer data; however, it does not make its entire internal telemetry system public (YouTube, n.d.-m; YouTube, n.d.-o).
- The third concerns the economic component, through data such as advertising impressions, interactions with advertisements, subscription status and monetisation events: events with economic relevance, such as a purchase through YouTube Shopping, a Super Thanks or other transactions associated with the platform.

This data feeds different functions: Analytics makes indicators available to creators, while viewers' responses contribute to recommendation systems. The cycle illustrated below is a functional summary: it does not imply that every event collected feeds into the same model or is processed immediately (YouTube, n.d.-d; Goodrow, 2021).
A continuous cycle is therefore created:
Playback → Data → Analysis and Machine Learning → New decisions → New playback

In terms of basic processing, a Short shares many components with other videos.
The main difference emerges primarily at the levels of discovery, user interaction and monetisation.
The client does not necessarily wait for the user to choose a video through a thumbnail.

The system selects the next piece of content and the player presents it immediately.
The cycle therefore becomes particularly rapid:

```
Candidate Short
↓
Playback
↓
Watch / Swipe / Rewatch / Interaction
↓
New signals
↓
Next candidate
```

In this case, the viewer's swipe is not merely an interface command.
It is also a new signal that can influence the next selection.

Live streaming requires a different pipeline from Video on Demand.

- In Video on Demand, YouTube first receives a complete file and can therefore process it before playback.
- In live streaming, by contrast, future content does not yet exist: the pipeline must work while the signal is being produced.

The creator or production system therefore sends a continuous audiovisual stream to YouTube's infrastructure.

For live ingest, YouTube supports several protocols, including RTMP, RTMPS, HLS and DASH (Google, n.d.-e).

- *RTMP* and *RTMPS* are particularly common in standard encoder workflows.
- *HLS* and *DASH* can also be used for high-quality, high-resolution streams.

Before broadcasting, the creator configures the stream through **YouTube Live Control Room**: YouTube's graphical interface dedicated to creating, configuring, monitoring and managing live broadcasts.
From this interface, the creator can, for example, set the broadcast's title, description, visibility, scheduling, latency and other options, as well as obtain the parameters needed to connect their encoder to YouTube's ingest infrastructure (YouTube, n.d.-k).
YouTube therefore provides the encoder with at least two fundamental pieces of information:

- Ingestion URL: the address of the ingest infrastructure to which the stream should be sent.
- Stream Key: a key that allows YouTube to identify and accept the stream sent by the encoder.

With *RTMP* or *RTMPS*, the encoder uses the server address and stream key to determine the destination to which audio and video should be sent continuously.
In simplified form:
Server *URL* + Stream Key → Identified ingest stream

The **YouTube Live Streaming API** is the software interface that allows external applications and systems to create, configure and manage a YouTube live broadcast programmatically, without manually using Live Control Room.
Through the API, it is possible, for example, to create a broadcast, configure its parameters, associate the stream from the encoder and obtain the information needed for ingest.
In the Live Streaming API, YouTube exposes equivalent concepts through an ingestion address, the address to which the stream should be sent, and a stream name, which identifies the configured stream (Google, n.d.-a).
Depending on the encoder, these values can be entered separately or combined in the form: STREAM_URL / STREAM_NAME

However, the stream key should not necessarily be interpreted as the identifier of a single live broadcast: YouTube allows the same key to be reused for subsequent broadcasts. Its main function is to allow the infrastructure to recognise and accept the stream sent by the encoder (YouTube, n.d.-k).

When the stream reaches the ingest infrastructure, it must be processed in real time.
YouTube therefore transcodes the live stream to produce different output formats and qualities, so that viewers with different devices and connections can follow the same broadcast (YouTube, n.d.-c).
Unlike *VOD*, however, it is not possible to analyse all the content in advance: future frames have not yet been produced.

This difference introduces one of the main problems of live streaming: latency.
Reducing it requires reducing the amount of content accumulated before playback.
But this also reduces the margin with which the player can absorb network variations.
We therefore have a trade-off:

- Larger buffer → Greater stability → Greater latency.
- Smaller buffer → Lower latency → Greater risk of rebuffering.

YouTube therefore offers different latency settings.

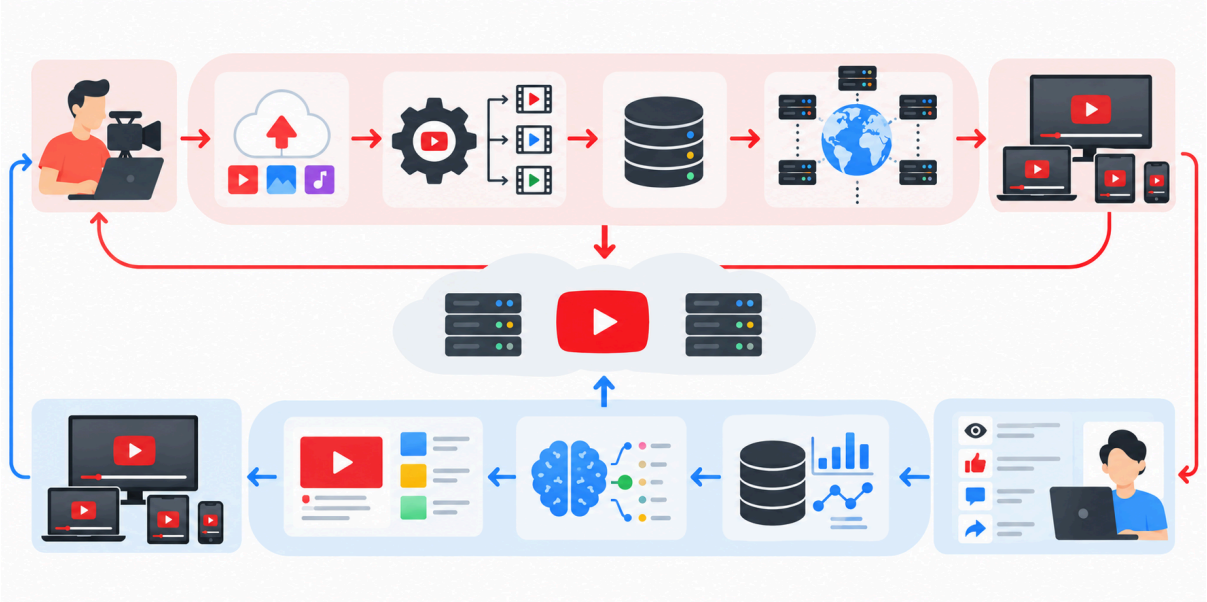
- Normal mode places greater emphasis on stability and quality.
- Low-latency mode reduces delay; ultra-low-latency mode is designed for highly interactive experiences but increases the risk of buffering (YouTube, n.d.-o).

Low-latency mode is particularly important, for example, for live chats or sporting events, because interaction loses value if the viewer sees the event with too much delay.
Live streaming must also integrate with the same systems present throughout the platform: the discovery system, rights, advertising, analytics and account management.
A live stream can appear on Home, be recommended, be monetised and produce data while it is being broadcast.

Live streaming is therefore not a system completely separate from YouTube.
Functionally, the live pipeline integrates with the platform's distribution, discovery, rights and monetisation. This convergence does not imply that all internal components are identical to those of VOD.

YouTube's overall architecture can therefore be summarised as a system in which two major flows meet.

- The first is the content flow: Creator → Upload → Processing → Storage → Distribution → Viewer.
- The second is the information flow: Viewer → Behaviour and telemetry → Data → Models and decisions → Discovery → Viewer.



The two flows continuously influence each other.

- Content produces behaviour.
- Behaviour produces data.
- Data influences recommendations.
- Recommendations determine which content is shown or suggested to the viewer.

YouTube must simultaneously receive, process, understand, classify, protect, monetise, recommend and distribute a continuously growing quantity of content.
The platform's fundamental problem is therefore not simply: "How can a video be transmitted?" but:
"How can millions of new pieces of content continually be transformed into distributable audiovisual experiences relevant to billions of potential viewers?"

Supporting material

Bibliography

Alphabet Inc. (2025). Annual report 2024 (Form 10-K). https://abc.xyz/assets/59/ff/66de11a3bfa7db6cb929ba12a01b/6b3f31ad08de72f11d32f2b4f712b918.pdf

Ben-Yair, S. (2015). Introducing the newest member of our family, the YouTube Kids app—available on Google Play and the App Store. https://blog.youtube/news-and-events/youtube-kids/

Covington, P., Adams, J., & Sargin, E. (2016). Deep neural networks for YouTube recommendations. Proceedings of the 10th ACM Conference on Recommender Systems. https://research.google/pubs/deep-neural-networks-for-youtube-recommendations/

Darby, M., & Amarilio, O. (2015). Say hello to the redesigned YouTube mobile app. https://blog.youtube/news-and-events/say-hello-to-redesigned-youtube-mobile/

Doronichev, A. (2010). YouTube Mobile gets a kick start. https://blog.youtube/news-and-events/youtube-mobile-gets-kick-start/

Goodrow, C. (2021). On YouTube's recommendation system. https://blog.youtube/inside-youtube/on-youtubes-recommendation-system/

Google (2006a). Google closes acquisition of YouTube. https://googlepress.blogspot.com/2006/11/google-closes-acquisition-of-youtube_13.html

Google (2006b). Google to acquire YouTube for $1.65 billion in stock. https://googlepress.blogspot.com/2006/10/google-to-acquire-youtube-for-165_09.html

Google (n.d.-a). LiveStreams: YouTube Live Streaming API. Retrieved October 29, 2026, from https://developers.google.com/youtube/data/live/reference/rest/v3/liveStreams

Google (n.d.-b). Network infrastructure. Retrieved October 29, 2026, from https://peering.google.com/static/js/site/modules/pages/infrastructure.html

Google (n.d.-c). Videos: YouTube Data API. Retrieved October 29, 2026, from https://developers.google.com/youtube/data/reference/rest/v3/videos

Google (n.d.-d). Widevine: Overview. Retrieved October 29, 2026, from https://developers.google.com/widevine/drm/overview

Google (n.d.-e). YouTube Live Streaming ingestion protocol comparison. Retrieved October 29, 2026, from https://developers.google.com/youtube/data/live/guides/ingestion-protocol-comparison

Google Cloud (n.d.). Origins overview: Media CDN. Retrieved October 29, 2026, from https://docs.cloud.google.com/media-cdn/docs/origins

Greenberg, J. (2025). Introducing Premium Lite: Watch your favorite creators ad-free. https://blog.youtube/news-and-events/introducing-premium-lite/

Hanif, A. (2022). Made on YouTube: Supporting the next wave of creative entrepreneurs. https://blog.youtube/news-and-events/supporting-the-next-wave-of-creative-entrepreneurs/

Hurley, C. (2009). Y,000,000,000uTube. https://blog.youtube/news-and-events/y000000000utube/

Jeon, A. (2017). YouTube TV is now live. https://blog.youtube/news-and-events/youtube-tv-is-now-live/

Katz, T. (2012). Mission complete: Red Bull Stratos lands safely back on Earth. https://blog.youtube/news-and-events/mission-complete-red-bull-stratos-lands/

Leske, M. (2015). Meet YouTube Red, the ultimate YouTube experience. https://blog.youtube/news-and-events/red/

Mahesh, V. (2025). 20 ways we're celebrating two decades of YouTube. https://blog.youtube/news-and-events/happy-birthday-youtube-20/

Mohan, N. (2026). From the CEO: What's coming to YouTube in 2026. https://blog.youtube/inside-youtube/the-future-of-youtube-2026/

Sarukkai, R. (2010). What's bigger than 1080p? 4K video comes to YouTube. https://blog.youtube/news-and-events/whats-bigger-than-1080p-4k-video-comes/

Voolich, J. (2025). The next 20: Powering the future of entertainment together at Made on YouTube. https://blog.youtube/news-and-events/made-on-youtube-2025/

Wilms, K. (2016). We'll do it live—a new chapter in YouTube's live stream. https://blog.youtube/news-and-events/well-do-it-livea-new-chapter-in/

Wojcicki, S. (2020). YouTube at 15: My personal journey and the road ahead. https://blog.youtube/news-and-events/youtube-at-15-my-personal-journey/

YouTube (2005). YouTube receives $3.5M in funding from Sequoia Capital. https://blog.youtube/news-and-events/youtube-receives-35m-in-funding-from/

YouTube (2012). It's YouTube's 7th birthday... and you've outdone yourselves, again. https://blog.youtube/news-and-events/its-youtubes-7th-birthday-and-youve/

YouTube (2013). Here's to eight great years. https://blog.youtube/culture-and-trends/heres-to-eight-great-years/

YouTube (2015). Celebrating 10 Years of YouTube. https://blog.youtube/news-and-events/celebrating-10-years-of-youtube/

YouTube (2018). Introducing YouTube Premium. https://blog.youtube/news-and-events/introducing-youtube-premium/

YouTube (2021). Reimagining video infrastructure to empower YouTube. https://blog.youtube/inside-youtube/new-era-video-infrastructure/

YouTube (2022). 15 years of YouTube in APAC. https://blog.youtube/news-and-events/15-years-of-youtube-in-apac/

YouTube (2025). Cleo Abram & Amjad Hanif's hot takes on AI, ownership & your channel's future. https://blog.youtube/creator-and-artist-stories/cleo-abram-amjad-hanifs-hot-takes/

YouTube (n.d.-a). Buy or rent movies & TV shows on YouTube. Retrieved October 29, 2026, from https://support.google.com/youtube/answer/9674101?hl=en

YouTube (n.d.-b). Change the quality of your video. Retrieved October 29, 2026, from https://support.google.com/youtube/answer/91449?hl=en

YouTube (n.d.-c). Choose live encoder settings, bitrates, and resolutions. Retrieved October 29, 2026, from https://support.google.com/youtube/answer/2853702?hl=en

YouTube (n.d.-d). Content tab analytics tips (Video). Retrieved October 29, 2026, from https://support.google.com/youtube/answer/12942217?hl=en

YouTube (n.d.-e). Explore YouTube Music Premium benefits. Retrieved October 29, 2026, from https://support.google.com/youtubemusic/answer/9266556?hl=en

YouTube (n.d.-f). How Content ID works. Retrieved October 29, 2026, from https://support.google.com/youtube/answer/2797370?hl=en

YouTube (n.d.-g). How YouTube recommendations work. Retrieved October 29, 2026, from https://support.google.com/youtube/answer/16089387?hl=en

YouTube (n.d.-h). How YouTube search works. Retrieved October 29, 2026, from https://support.google.com/youtube/answer/16090438?hl=en

YouTube (n.d.-i). Impressions & click-through-rate FAQs. Retrieved October 29, 2026, from https://support.google.com/youtube/answer/7628154?hl=en

YouTube (n.d.-j). Low video quality after upload. Retrieved October 29, 2026, from https://support.google.com/youtube/answer/71674?hl=en

YouTube (n.d.-k). Manage live stream settings. Retrieved October 29, 2026, from https://support.google.com/youtube/answer/9854503?hl=en

YouTube (n.d.-l). Search & discovery tips (Shorts). Retrieved October 29, 2026, from https://support.google.com/youtube/answer/11914225?hl=en&co=YOUTUBE._YTVideoType%3Dshorts

YouTube (n.d.-m). Send YouTube debug info. Retrieved October 29, 2026, from https://support.google.com/youtube/answer/7519898?hl=en

YouTube (n.d.-n). Understand commission rates & earnings in the YouTube Shopping affiliate program. Retrieved October 29, 2026, from https://support.google.com/youtube/answer/17263096?hl=en

YouTube (n.d.-o). Understand live streaming latency. Retrieved October 29, 2026, from https://support.google.com/youtube/answer/7444635?hl=en

YouTube (n.d.-p). Understand your content performance for YouTube's recommendation system. Retrieved October 29, 2026, from https://support.google.com/youtube/answer/16559650?hl=en

YouTube (n.d.-q). Use automatic dubbing. Retrieved October 29, 2026, from https://support.google.com/youtube/answer/15569972?hl=en

YouTube (n.d.-r). Your content & YouTube Premium. Retrieved October 29, 2026, from https://support.google.com/youtube/answer/6306276?hl=en

YouTube (n.d.-s). YouTube partner earnings overview. Retrieved October 29, 2026, from https://support.google.com/youtube/answer/72902?hl=en

YouTube (n.d.-t). YouTube recommended upload encoding settings. Retrieved October 29, 2026, from https://support.google.com/youtube/answer/1722171?hl=en

YouTube (n.d.-u). YouTube Shorts monetization policies. Retrieved October 29, 2026, from https://support.google.com/youtube/answer/12504220?hl=en

Zack, S. (2010). Content ID and fair use. https://blog.youtube/news-and-events/content-id-and-fair-use/

Further reading

- Paul Covington, Jay Adams and Emre Sargin, *Deep Neural Networks for YouTube Recommendations*, Proceedings of the 10th ACM Conference on Recommender Systems, 2016. — A foundational research paper examining the application of deep neural networks to large-scale video recommendation. The authors describe a two-stage architecture consisting of candidate generation, which selects potentially relevant videos from an enormous catalogue, and ranking, which orders those candidates according to predicted user behaviour. The paper is particularly relevant to the handout's explanation of YouTube's Discovery System because it provides the

original technical basis for distinguishing content retrieval from personalised ranking. It also examines how user histories, contextual information and implicit feedback can contribute to prediction. The research provides an important foundation for understanding how a platform can transform billions of available videos into a manageable selection for each viewer. Since the paper describes a system published in 2016, it should be regarded as a historical architectural reference rather than a complete specification of YouTube's current recommendation algorithms.

- Cristos Goodrow, *On YouTube's Recommendation System*, YouTube Official Blog, 2021. — A firsthand explanation of the principles guiding YouTube's recommendation systems and their evolution from popularity-based suggestions towards personalised content discovery. Goodrow examines the different roles of clicks, viewing duration, survey responses, likes, dislikes and other signals in estimating whether viewers find a video valuable. The article is particularly relevant to the handout's distinction between Engagement and Satisfaction because it explains why watch time alone cannot adequately represent the value of a viewing experience. It also discusses the importance of authoritative information and the platform's efforts to reduce recommendations of potentially harmful or misleading content. The resource offers a valuable conceptual perspective on the difference between maximising immediate interactions and recommending videos that viewers are likely to appreciate.
- YouTube, *Understand Your Content Performance for YouTube's Recommendation System*, YouTube Help. — An official guide examining how YouTube interprets viewer responses through three broad categories: Appeal, Engagement and Satisfaction. The document explains how the decision to watch, continued viewing and the perceived value of the experience contribute to evaluating content performance. It is particularly relevant to the handout because it directly develops the three concepts used to explain the relationship between creators, content and audiences. The guide also provides practical advice on choosing topics, designing thumbnails, writing titles, improving introductions and analysing audience retention. It clarifies why a creator should not treat the recommendation system as a collection of isolated ranking tricks, but instead seek to produce content that is understandable, relevant and satisfying for its intended audience.
- YouTube, *How YouTube Search Works*, YouTube Help. — An official explanation of the mechanisms used to organise video search results according to relevance, engagement and quality. The documentation describes how YouTube considers the relationship between a search query and video metadata, including titles, descriptions and tags, together with information derived from the audiovisual content and viewer behaviour. It is particularly relevant to the handout's distinction between Search and recommendation surfaces such as Home and Up Next. The guide explains why search optimisation cannot be reduced to keyword repetition and why the usefulness and credibility of the source may become particularly important for informational queries. It provides a theoretical foundation for understanding YouTube SEO as the process of making content discoverable and genuinely relevant to users' search intentions rather than manipulating a fixed ranking formula.
- YouTube, *YouTube Partner Earnings Overview*, YouTube Help. — An official guide describing how creators participating in the YouTube Partner Program can earn revenue through advertising, subscriptions and other monetisation features. The document explains the different contractual modules associated with Watch Page advertising, Shorts monetisation and fan funding, including the applicable

revenue-sharing arrangements. It is particularly relevant to the handout's analysis of YouTube as a multisided platform because it demonstrates how the same infrastructure supports different economic relationships between creators, viewers, advertisers and YouTube itself. It also clarifies why creator earnings cannot be calculated simply by multiplying video views by a fixed payment rate. The resource provides an essential foundation for understanding revenue sharing, the role of net advertising revenue and the different commercial mechanisms that sustain the creator economy.

- YouTube, *YouTube Shorts Monetization Policies*, YouTube Help. — An official explanation of the economic model used to distribute advertising revenue generated within the YouTube Shorts feed. The documentation describes how advertising revenue is pooled, how music licensing costs are considered and how the Creator Pool is allocated among eligible monetising creators according to their share of qualifying engaged views. It is particularly relevant to the handout's comparison between long-form videos and Shorts because it demonstrates how different interaction models require different monetisation mechanisms. Unlike Watch Page advertising, where advertisements can be associated with a specific viewing context, Shorts advertisements appear between videos in a continuous feed. The document provides a concrete example of how interface design, user behaviour, rights management and economic distribution become interdependent components of a digital platform.
- YouTube, *How Content ID Works*, YouTube Help. — An official technical introduction to YouTube's automated copyright identification system. The document explains how eligible rights holders supply reference files that are compared with uploaded audiovisual content to identify possible matches. When a match is detected, the rights holder may apply policies that block, track or monetise the video, depending on the relevant rights and territorial conditions. It is particularly relevant to the handout's distinction between content processing, rights management and monetisation because it demonstrates how copyright enforcement can become part of an automated distribution infrastructure. The guide also helps clarify why identifying a match does not independently establish legal ownership and why claims may be disputed. It provides an important foundation for understanding how user-generated content platforms coordinate access to copyrighted works at scale.
- YouTube, *Reimagining Video Infrastructure to Empower YouTube*, YouTube Official Blog, 2021. — A technical account of the development of specialised hardware for large-scale video transcoding. YouTube describes how its engineering teams created Video Coding Units (VCUs) to accelerate the computationally intensive process of transforming uploaded videos into representations suitable for different devices, codecs and resolutions. The article is particularly relevant to the handout's Content Processing and Transcoding section because it demonstrates why processing audiovisual content at YouTube's scale requires extensive distributed computing resources. It also explains how specialised hardware can improve computational efficiency compared with conventional CPU-based encoding systems. The resource offers a concrete example of the relationship between video compression, data-centre architecture, operational costs and the need to process continuously growing quantities of user-generated content.
- Google, *Network Infrastructure*, Google Peering and Network Infrastructure Documentation. — An official description of Google's global network infrastructure,

including data centres, edge Points of Presence and Google Global Cache nodes. The document explains how Google uses geographically distributed systems and interconnections with Internet Service Providers to reduce network distance and deliver content efficiently. It is particularly relevant to the handout's distinction between the Control Plane and the Media Plane because it provides a concrete architectural perspective on the infrastructure responsible for delivering video to viewers. Google Global Cache is especially important for understanding how popular YouTube content can be temporarily stored within operators' networks instead of being retrieved repeatedly from distant data centres. The resource also helps distinguish the functions of centralised processing, network interconnection and edge caching, while avoiding the assumption that every YouTube request necessarily follows an identical internal delivery path.

- Google, *YouTube Live Streaming Ingestion Protocol Comparison*, Google for Developers. — An official technical comparison of the protocols supported by YouTube for receiving live audiovisual streams from external encoders. The documentation examines RTMP, RTMPS, HLS and MPEG-DASH, distinguishing their encryption capabilities, supported video codecs, latency characteristics and suitability for high-resolution broadcasting. It is particularly relevant to the handout's Live Streaming Architecture section because it demonstrates how the platform can receive streams through different ingest technologies before processing and distributing them to viewers. The guide also explains why RTMPS is commonly used for conventional live encoding workflows, while HLS and DASH ingest can support higher-resolution content and additional codec options at the cost of different latency characteristics. It provides an applied foundation for understanding the relationship between encoder configuration, contribution protocols and YouTube's real-time processing pipeline.

Editorial notes

In producing these materials, I have used generative artificial intelligence tools to support the writing process, particularly to improve the form of the text, reorganize content, refine wording, and speed up certain editorial tasks.

As I work independently on the production of these materials, I try to automate any activities that can reasonably be automated, so that I can devote more time to research, design, and content development.

However, artificial intelligence does not determine the content of this work: the choice of topics, structure, ideas, interpretations, examples, and teaching approach are developed by me. AI is therefore used primarily as a tool to support production and formal revision, while authorship and responsibility for the design and content remain mine.

Unless otherwise indicated, this material is distributed under the Creative Commons Attribution–NonCommercial–ShareAlike 4.0 International license (CC BY-NC-SA 4.0).

You are therefore free to share, redistribute, adapt, and create derivative works based on this material, provided that appropriate credit is given, the material is not used for commercial purposes, and any modified or derivative versions are distributed under the same license.

