OpenSSL CONFERENCE

Beyond copyright: IP strategy in the age of generative AI.

Hayden Delaney | Partner, Thomsons | OpenSSL Conference 2026, Prague

§ PRIVACY AS A FUNDAMENTAL HUMAN RIGHT

20/10/2026

One missing line

No hack. No insider. The rights holder published it, and copyright was the only tool left.

20/10/2026

- **31 March 2026, morning.** Anthropic publishes Claude Code v2.1.88 to npm with a 59.8 MB source map inside: 512,000 lines of unobfuscated TypeScript. One exclusion line was missing from the packaging config.
- **Same day, within hours.** A security researcher spots the map and posts it. The source is downloaded from Anthropic's own storage, mirrored across GitHub and forked tens of thousands of times.
- **31 March, evening.** A DMCA notice cascades through the fork network: more than 8,100 repositories disabled, including legitimate forks of Anthropic's own public repository.
- **31 March, later that day.** Anthropic's statement: a release packaging issue caused by human error, not a security breach; no customer data or credentials exposed.
- **1 April. The notice is retracted to one repository and 96 forks. The same day claw-code, described by its author as a clean-room Python rewrite produced with an AI coding tool, passes 100,000 stars and overtakes Anthropic's own repository. No takedown can reach it.**

„A release packaging issue caused by human error, not a security breach.“

- Anthropic statement to CNBC, Fortune and The Register, 31 March 2026

Four threshold issues

01 Authorship and ownership

Copyright needs a human author. For AI-generated code that is now a question to be evidenced, not a given.

02 Idea/expression dichotomy

Generative AI re-implements the idea at machine speed without copying the expression. The moat copyright provided has collapsed.

03 Implications for open source

Permissive and copyleft licences bind downstream users through copyright, not contract. No copyright, no licence.

04 Protection

Copyright is a foundation, not a fortress. Choose the regime deliberately, asset by asset, and govern the rest.

OpenSSL CONFERENCE

Copyright needs a human author

Across every jurisdiction this audience ships to, copyright in code assumes a person made the creative choices.

20/10/2026

- **Australia.** IceTV v Nine Network [2009] HCA 14 and Telstra v Phone Directories [2010] FCAFC 149 (Australia): originality requires a human author's independent intellectual effort. Acohs v Ucorp (2012) 201 FCR 173 (Australia): code generated by software was not its programmers' work. Commissioner of Patents v Thaler [2022] FCAFC 62 (Australia): an inventor must be a natural person.
- **United States.** Thaler v Perlmutter (DC Cir 2025; certiorari denied March 2026) (USA): human authorship required. US Copyright Office (January 2025): prompts, however detailed, are instructions, not authorship.
- **EU and Germany.** Infopaq (C-5/08) (EU): the author's own intellectual creation. Sections 2(2) and 7 UrhG (Germany): only natural persons are authors. European Parliament resolution, March 2026 (EU): fully AI-generated output should be unprotected.
- **United Kingdom.** The outlier on its face: s 9(3) CDPA (UK) deems the arranger the author of a computer-generated work. Whether such a work can be “original” is unresolved, and the government's March 2026 report says the provision should go.
- **Closer to home.** Prague Municipal Court (2023) (Czech Republic): a bare prompt earns nothing. Beijing Internet Court (2023) (China): sustained creative control can.

The working spectrum for code

OpenSSL CONFERENCE

Three ways AI-augmented code arrives, and where each sits on the copyright line

1. Prompt and adopt
A general prompt ("write a function implementing the TLS 1.3 handshake"); output taken with cosmetic edits: renamed variables, reformatting.

2. Specify and rework
Detailed, iterative prompts fixing the architecture, the RFC and the error strategy; then substantive rewriting and integration by the engineer.

3. Author and review
The engineer writes the code; the AI reviews it and suggests targeted changes, which the engineer accepts, rejects or reworks one by one.

Almost certainly unprotected
The structure, algorithm and logic came from the model. No human creative act to attach copyright to.

Uncertain: case by case
Substantial human influence, but the foundation was machine-generated. Documentation decides it.

Protected
The creative decisions stayed with the engineer. The AI functioned like a linter or a reviewer.

Human creative control increases

Model decides structure, algorithm and logic

Engineer decides; the model assists

Copyright only. Whether the same code is patentable, or protected as a trade secret, is a separate question and is unaffected by this spectrum.

The gap, weaponised

The idea/expression dichotomy was designed to permit independent invention. It has become a template for AI-assisted replication.

20/10/2026

- Copyright protects expression, never the idea, method or function. That is sound doctrine: it is what permits independent creation and keeps copyright from becoming a monopoly on ideas.
- Generative AI now exploits that gap with precision: it absorbs material at scale and produces outputs with the same function and, potentially (this is the big “if”), no reproductions of the underlying work.
- A functional equivalent of a mature software product can now be produced in hours rather than engineer-months, and without copying a line.
- Equity markets have priced it: the 2025 to 2026 correction in listed software valuations rests on exactly this thesis (aka the SaaSpocalypse).

The law that makes it lawful

OpenSSL CONFERENCE

- **Function is not protected.** Copyright protects the form in which a program is expressed, never its function, method or algorithm: s 102(b) Copyright Act (USA); Art 1(2) Software Directive 2009/24 (EU); Computer Edge v Apple (1986) 161 CLR 171 and Data Access v Powerflex (1999) 202 CLR 1 (Australia).
- **SAS Institute v World Programming (CJEU C-406/10, 2012; [2013] EWCA Civ 1482) (EU, UK).** Functionality, the programming language and data file formats are not forms of expression. WPL studied the SAS manuals and reproduced the system's behaviour without its code, lawfully.
- **Navitaire v easyJet [2004] EWHC 1725 (Ch) (UK).** Rebuilding a booking system's commands and behaviour from observation, with no access to the source, did not infringe: no copyright in the command set or the business logic. Reaffirmed in Edozo v Valos [2026] EWHC 93 (IPEC) (UK).
- **Google v Oracle, 593 US 1 (2021) (USA).** Copying 11,500 lines of Java API declaring code, and writing fresh implementing code, so programmers could use what they already knew was fair use. Declaring code sits further from the core of copyright than other code.
- **For cryptographic code the rule is at its strongest. AES, RSA, SHA-2 and ECDSA are published methods; the RFCs are the specification. What copyright has ever protected in a crypto library is this particular implementation, and nothing else.**

Clean room, automated

The clean-room tradition

- Sega v Accolade (9th Cir 1992) and Sony v Connectix (9th Cir 2000) (USA): copying a program to discover its unprotected function is fair use.
- NEC v Intel (ND Cal 1989) (USA): NEC had an isolated team, given only the specification, write microcode from scratch; its resemblance to Intel's showed the similarities were dictated by function, and NEC's own microcode was held not to infringe.
- The method relied on separation: one team reads, writes a spec; a second team that has never seen the code implements it. Cost: engineer-months. Evidence: the wall between the rooms.

Why AI is not a clean room

- The model has read the expression, in training or in the prompt. There is no wall. It produces clean-looking output from a contaminated room.
- Cost collapses from engineer-months to hours, and the spec-writing team is the model itself.
- The only legal question left is the big if: did any of the original expression survive into the output? That is what the memorisation cases decide.
- **If none did, is the re-implementation is arguably lawful?**

OpenSSL
CONFERENCE

claw-code: the moat in one repository

The idea/expression dichotomy, exploited exactly as designed, by a machine that never needed to read the manual.

20/10/2026

- **What it is.** A Python re-implementation of the leaked Claude Code architecture, described by its author as a clean-room rewrite produced with an AI coding agent. Published within a day of the leak; 100,000 GitHub stars in 24 hours; more stars than Anthropic's own repository by 1 April 2026.
- **What it took.** Architecture, tool design, agent loop, command structure, behaviour: the ideas. In a different language, with different expression.
- **What it did not take.** The TypeScript. No DMCA notice can reach it, because there is no copy to remove. Trade secret protection was already gone: the rights holder published the source.
- **What it cost.** Hours. The moat copyright gave software was never the right itself but the engineer-months a non-infringing rewrite required. That cost is now near zero.
- **The big if, applied.** The agent had the original in front of it. If a court found the original's expression surviving in claw-code, the memorisation cases that follow would bite. If not, it is Navitaire at machine speed.

„Think of ChatGPT as a blurry JPEG of all the text on the Web.“

- Ted Chiang, The New Yorker, February 2023

OpenSSL CONFERENCE

GEMA v OpenAI

When the output infringes. The court did not move the idea/expression line; it decided, on the outputs, which side of the line the model sat on.

20/10/2026

- **LG München I (Munich Regional Court), 11 November 2025, 42 O 14139/24 (Germany).** Nine well-known German song lyrics that ChatGPT (GPT-4 and 4o) reproduced almost verbatim on request.
- **Patterns versus copies: the dichotomy as the battleground.** OpenAI's case was that the model holds only statistical patterns, the idea side of the line. The court held that training had transferred not merely abstract information about the lyrics but the lyrics themselves into the parameters.
- **Memorisation is reproduction.** Expression retained in the weights is a reproduction (s 16 UrhG (Germany); Art 2 InfoSoc Directive (EU)); outputting it is a further reproduction and a communication to the public (s 19a UrhG). The text and data mining (TDM) exception (s 44b UrhG; Art 4 DSM Directive (EU)) permits analysis, not retention.
- **The model provider is directly liable, not the user.** OpenAI chose the training data and the architecture, so it controlled the output. Injunction, disclosure and damages ordered.
- **Status:** first instance, not final. OpenAI lodged its appeal with the Munich Higher Regional Court (OLG München, 6 U 3662/25 e) on 8 December 2025; no appellate ruling as at late September 2026. The first-instance court declined the parties' joint request to refer questions to the Court of Justice of the European Union (CJEU); the CJEU will address the same issues in Like Company v Google (C-250/25) (EU), pending.

The wider picture

OpenSSL CONFERENCE

- **GEMA v Suno(LG München I, 31 July 2026) (Germany).** Six compositions memorised and recognisable in output; US law applied to the US training and fair use rejected; injunction, with disclosure and damages for three works and a chorus. Suno to appeal.
- **Getty Images v Stability AI [2025] EWHC 2863 (Ch) (UK).** The mirror image. The training claim went for want of UK jurisdiction; the outputs claim was abandoned once Stability blocked the prompts that had produced near-copies; the model itself was held to store no copy of any work (at [600]). Only watermarks survived, as trade marks.
- **Bartz v Anthropic and Kadrey v Meta (ND Cal 2025) (USA).** Training transformative because the models learned rather than reproduced; Llama could not be made to emit more than a few dozen words. Bartz turned on the pirated library: USD 1.5 billion settlement, final July 2026.
- **Thomson Reuters v Ross (D Del 2025; Third Circuit pending) (USA).** Infringement found: 2,243 Westlaw headnotes copied to train a competing tool, fair use rejected. The copied expression was the input; the functional-access cases (Sega, Connectix) did not apply.
- **In re OpenAI (SDNY MDL) (USA).** The New York Times case rests on verbatim regurgitation; OpenAI says adversarial prompting. Summary judgment pending.

What the cases say about the line

Copyright's boundary is intact. Generative AI has simply learned to live on the right side of it.

20/10/2026

- **The line has not moved.** No court has protected style, structure, function or learned patterns. Every finding of liability rested on recognisable expression in the output.
- **The battleground has moved.** Whether a model holds expression or only ideas is now decided empirically, on output evidence, prompt by prompt.
- **Two outcomes, cleanly split.** Expression survives: infringement, and in those cases the model provider is the defendant. Expression does not survive: no copyright remedy, and rights holders retreat to trade marks, input piracy or market-harm theories.
- **A well-guarded model gives the rights holder nothing.** Output filters and anti-regurgitation training are copyright compliance measures.
- **For code, this is the whole problem.** Functional re-implementation without copied expression sits exactly where these cases offer no remedy. The only foothold is regurgitation of licensed code, and the industry is engineering that out.

Why open source is so exposed

- MIT, Apache 2.0 and the GPL family are granted by the copyright owners (the contributors) to whoever receives the code. A breach of the open source licence which results in termination relies on copyright to enforce unlicensed use.
- Enforcement therefore rests on the underlying copyright and every open source contract, in theory, presupposes something to license.
- If AI-generated contributions carry no copyright, the licence has nothing to attach to for that code.
- Copyleft terms reach only what copyright reaches: copies and derivatives of the expression.
- A model trained on a GPL codebase that re-implements its function in fresh code produces neither, so no share-alike, source or notice obligation attaches.
- Where it instead reproduces the licensed expression, the GPL applies, the contributor's CLA representations are false without their knowing, and GPL-2.0 code cannot lawfully sit inside an Apache 2.0 project.

An AI-aware CLA

Disclosure

Notify at submission: which parts used AI, which tool, whether third-party code went into the prompt. A field in the pull-request template; enhanced review for disclosed contributions.

Human effort

Where tools were used, the contributor has reviewed, understood and materially modified the output so that the contribution reflects their own independent intellectual effort. This is the originality warranty, made honest.

Tool terms

Compliance with the tool's terms, including output-ownership provisions; a written AI tools policy naming approved tools and prohibiting tools whose terms retain rights in outputs.

Provenance

No third-party reproduction, notices preserved, tool and prompt records retained. The project scans disclosed contributions against public code, so inadvertent reproduction is caught and negligence refuted.

Patents: built for methods

What patents do that copyright cannot

- Protect the method, the system architecture and the workflow, which is precisely what generative AI reproduces most easily.
- Enforceable against independent development and reverse engineering. A clean-room rewrite is no defence.
- Apache 2.0 clause 3 grants a patent licence from every contributor, and the grant attaches to a contribution whether or not copyright subsists in it. AI output that carries no copyright still carries the patent licence.

The trade-offs, stated plainly

- Expensive and slow. Subject-matter, novelty and inventive-step thresholds apply, and software patents remain contested in several jurisdictions.
- Thaler (Australia, and the same result in the USA, UK and EU): there must be a human inventor. The same evidence problem as copyright, moved upstream.
- File before disclosure. Shipping features publicly first forfeits protection outside grace-period jurisdictions.
- This community's long unease with software patents is legitimate.
- Defensive tools exist: patent pledges, Open Invention Network, licence-embedded grants. E.g. note the breadth of Apache's defensive termination: it is triggered by a counterclaim, not just a suit, and it ends the patent licences from every contributor, not only the one sued.

Trade secrets, redefined

For anything that genuinely stays inside the organisation, trade secret protection is now the stronger regime. But only if the AI terms are right.

20/10/2026

- Trade secret and confidential information protection has always required two things: information that is genuinely confidential, and reasonable steps to keep it that way.
- Generative AI has changed the content of reasonable steps. Submitting confidential code, system prompts or business logic to a public AI service whose terms permit retention or training is the fastest way to destroy the claim.
- The threshold question is no longer only who holds the information, but into which models it has been ingested.
- Confidentiality frameworks that do not address AI use, internally and in vendor contracts, are not doing their job.

A layered strategy

Copyright

Direct republishers, bad-faith mirrors, identifiable infringers, and the foundation of every licence. Preserve it with evidence of human authorship.

Patents

Methods, architectures and workflows that AI can replicate without copying. File before disclosure. Use defensive structures.

Trade secrets

Everything that genuinely stays inside. AI-aware vendor terms, internal use policy and the discipline to keep it there.

Governance

Build controls, contribution policy, AI-use policy and contract. The layer that would have caught the missing line.

OpenSSL CONFERENCE

The lesson

A layered IP strategy is required. Reliance on copyright alone is no longer a strategy; it is a default that nobody chose.

- Copyright remains indispensable: it is the foundation the entire open source licensing model stands on.
- It has never protected function, and generative AI now re-implements function at machine speed without copying expression. The moat was the cost of the rewrite, and the cost is gone.
- So choose regimes deliberately. Patents for methods. Trade secrets for what stays inside, with AI-aware terms. Governance underneath all of it.
- For open source: decide what AI-generated contributions mean for your licence, and record the decision before someone else's lawyer asks.

OpenSSL CONFERENCE

Thank you. Questions?

Hayden Delaney | Partner, Thomsons | hdelaney@thomsons.com.au | thomsons.com.au/insights