

Arc Two: The Other Side of the Table

## Who Actually Owns Your Software?

**Thesis:** A founder proves she owns her software by the fact of having built it. A buyer proves it by the documents that exist when she runs the audit. Those are not different versions of acceptable proof. In an M&A transaction, only one of them is required. Software whose ownership cannot be confirmed is never priced as though it is clean.

### 01 The Repository

The deal had moved past the point where surprises were expected.

Revenue had been reconstructed and held. The customer base had passed the transfer test with one noted exception. The buyer's technical team had reviewed the architecture, asked two rounds of pointed questions about scalability, and concluded that the codebase was sound. The headline valuation was no longer provisional.

Then the buyer hands the repository to IP counsel. The instruction is narrow: review the codebase for title risk. Not functionality, not architecture, not whether the software does what the product documentation says it does. Only ownership. Only whether the company that built this software can demonstrate that it belongs to them.

The review takes eleven days. It produces three findings.

The first: a module managing subscription billing contains components drawn from an open source library carrying a copyleft license. The license permits use in internal systems. It does not permit commercial distribution without releasing the incorporating code under the same terms. The company has been commercially distributing this software for seven years. No one reviewed the license when the library was first adopted, because no one was thinking about an acquisition when a developer solved a billing problem with a library she found online.

The second: four engineers contributed materially to the original platform. Two were employees. Two were engaged as contractors in the business's second year. Their agreements specified deliverables, timelines, and payment. They did not include an IP assignment clause. Under Australian law, intellectual property created by a contractor belongs to the contractor by default. Both contractors have since moved on. One is contactable. One is not.

The third finding produces no document. It produces a gap. A principal engineer joined the company three years ago from a direct competitor. Within six months she had redesigned two of the platform's core modules. Her employment contract at the prior company contained a standard IP and non-compete clause. No one asked to review it before she was hired. No one thought to.

No one calls any of this a deal-breaker. The buyer adjusts the warranty schedule. The indemnity provisions expand. The escrow amount increases. The founder, who has spent eight years building this software, is now being asked to provide warranties about code she believed, without ever testing that belief, that she wholly owned.

### 02 The Asset Beneath the Revenue

The two previous issues in this arc examined what a buyer does with a revenue number. The first asked whether the revenue was what the founder said it was. The second asked whether it would survive after she left.

Both questions were about the income stream.

This issue asks about the asset beneath it.

# FOUNDER THESIS

In an acquisition, software is not simply a product. It is the thing whose ownership must transfer cleanly for the transaction to deliver what the buyer priced. She is not purchasing what the software does, or the years it took to make it do that. She is purchasing title to the asset that generates the business's value. That title has a chain of documents: for every significant component of the codebase, someone wrote it, under some arrangement, and either a document exists that transfers ownership of that work to the company, or it does not.

Founders remember who built the software. Buyers ask who can prove they own it.

Buyers have a name for the distance between those two positions. They call it the provenance gap. It is the difference between what a founder believes she owns and what she can demonstrate ownership of when someone external requires the proof. The IP audit is the workstream that measures it. And in a software company built at speed, with contractors who predate formal documentation, open source libraries adopted for technical reasons, and engineers who arrived carrying obligations from prior employers, the gap is rarely as small as the founder expects.

## 03 Three Sources of the Gap

The provenance gap accumulates in ordinary ways. It is not the product of negligence or bad faith. It is the natural residue of building software at the pace that early-stage companies build: shipping, hiring, solving problems with whatever is at hand. The audit simply makes visible what speed concealed.

The first source is open source adoption. Not all open source licenses are equal. A permissive license such as MIT or Apache imposes minimal conditions on commercial use. A copyleft license such as GPL or AGPL imposes a significant one: incorporate the library into software you distribute commercially, and you are required to release the incorporating code under the same terms. An engineer who uses a GPL library to solve a billing problem is making a technical decision. The acquisition process reclassifies it as a title risk. The buyer's IP counsel does not assume the license is permissive. She runs a scan.

The second source is contractor contribution. Every line of code in a commercial product was written by a person. The IP audit asks whether the company holds a written agreement transferring ownership of each significant contribution to it. For employees, an assignment clause in the employment contract usually provides the answer, though gaps are more common in early documentation than founders expect. For contractors, the Australian legal default is that IP belongs to the contractor unless a written agreement says otherwise. A founder who engaged contractors in the early years of her business may discover in diligence that the company has been licensing and monetising code for years that, in strict title terms, may never have been assigned to it.

The third source leaves no document to examine. It is the question of what arrived with engineers who came from competitors. A new hire can sign a confidentiality agreement on her first day. She cannot be audited for what she carried in her head from a prior employer. A buyer does not need courtroom proof to reprice warranty exposure. She needs proximity and uncertainty: a material rewrite of a core module in the months after a hire from a competitor, in an area where that competitor held relevant IP. The founder cannot resolve the question. She can only absorb the pricing consequence. And warranty risk, in a buyer's model, has a price.

The aggregate of these three exposures is the provenance gap. In each case, the fragility arrived not because someone made a reckless decision but because no one, at the time, had any reason to make a different one. That is what makes it so consistently present in otherwise strong software businesses. It is not a sign that something went wrong. It is a sign that something was never checked.

## 04 The Title Question

The IP audit is not a quality review. The codebase may be excellent. The architecture may be sound. The software may do exactly what it was built to do, and do it better than anything a competitor offers. None of that is what the audit is measuring.

The audit is measuring something simpler and more consequential. Whether the company that built this software can prove, document by document, that it is theirs to sell.

*A founder knows what her software does.  
A buyer needs to know who it belongs to.*

# FOUNDER THESIS

## 05 Before the Audit Runs

In a well-prepared business, the IP audit has been conducted internally before a buyer conducts it externally.

Open source dependencies have been catalogued and their licenses reviewed. Where copyleft libraries have been incorporated into commercially distributed software, they have been replaced with permissive alternatives, or legal advice obtained and documented. Contractor agreements from the early years have been located, and where IP assignment clauses are absent, assignment deeds have been executed with the original contributors while they remain findable. Employment contracts for key engineering hires have been reviewed for any obligations those engineers carried from a prior employer.

None of this is technically complex. It requires methodical review and someone who knows what a clean ownership record looks like. It requires, above all, the recognition that this is not legal hygiene. It is pricing preparation. The provenance gap, if it exists when a buyer finds it, does not sit in a legal annex. It enters the model. It widens the escrow. It expands the indemnity schedule. It lowers the amount the founder receives in the first wire.

The problem is not that founders are careless about IP ownership. The problem is that ownership, during the years of building, is not something anyone needs to test. The software ships. The customers renew. The engineers commit. The business grows. The question of whether the company can account for the authorship of every significant component of its codebase never becomes urgent until someone external decides to ask it.

By that point, one contractor is unreachable. One library has been in violation for seven years. And the buyer has adjusted the escrow amount upward to reflect risks the founder did not know she was carrying.

The audit did not introduce any of this. It introduced a number into the model.

*Software ownership is not established by the act of building. It is established by the documents that can be produced when someone asks for proof.*

The buyer is not asking what your software does.

The buyer is asking whether it is yours to sell.

By the time the audit produces its findings, the founder is reading the results of a test she did not know she was taking.



**Cube Capital** advises founders and boards of Australian software and technology companies on exit preparation and cross-border M&A transactions. The firm works exclusively on the sell side, retained by founders who want an independent view of what their business is worth and what it would take to make it worth more.

**Hani Iskander** is Founding Partner. He has founded technology companies in Australia and the United States, led the Asia Pacific region for a NASDAQ-listed software company, and spent four decades at the intersection of technology and capital markets.

web: [cube.capital](http://cube.capital) email: [hello@cube.capital](mailto:hello@cube.capital)

**Founder Thesis** began with Arc One, *The Prepared Founder Premium* (Issues 01–05).

*The Other Side of the Table* is Arc Two.

**Issue 06** *The Revenue Your Buyer Won't Pay For*

**Issue 07** *Your Customers Don't Belong to You*

**Issue 08** *Who Actually Owns Your Software?*

**Issue 09** *The Number That Decides Your Multiple*

**Issue 10** *The Risk You Are*

**Issue 11** *You're Not Selling History*

**Issue 12** *Technical Debt Is Financial Debt*

**Issue 13** *The Deal That Falls Over on the Seller's Side*

**Issue 14** *The Prepared Business Premium*