AI Transformation: A Journey — Part One

Ali Pichvai
Co-Founder of Quod Financial

21 May 2026

This paper builds upon my previous essay on how AI is impacting the software business[1]. It details our company's initiative to implement agentic engineering, which is part of our broader AI transformation.

Decades ago, I read George Soros' *The Alchemy of Finance*, which offers a practical, "real-time" account of his trading experiments and thought processes. I was fascinated to see that even a legendary investor goes through a standard process of discovery, complete with hesitations, mistakes, and reversals. It highlights that successful trailblazers rely on method and mindset, rather than prescience or mysterious foresight.

Furthermore, because AI development is exponential rather than linear, you must constantly re-evaluate your decisions and anticipate the major leaps that will soon make the impossible feasible.

Therefore, rather than writing a retrospective "know-it-all" account, I decided to compile the weekly emails we have sent to our engineering team over the last four months. In addition, I have added two of the chats from our own chat system, which sometimes provide a more candid discussion of our issues. These document our adoption phase, which paves the way for our fully agentic phase. This represents just one possible journey, but as the adage goes, 'All roads lead to Rome.'

Act 1. Scene 2.

Internal Inbox

A **Ali Pichvai** <a...@quodfinancial.com>
to @engineering ▾

Feb 27th, 2026

Dear all,

As we discussed in our company kick-off meeting in January 2026, AI transformation is now the top priority for the company.

Since January, there has been a remarkable acceleration in AI capabilities across coding, testing, and knowledge-based industries in general. This not only reinforces the business case for our AI transformation, but also triggers a race against time, a race for relevance, if not survival.

What do we mean by AI Engineering? Simply put, it is the emerging and extremely powerful practice of using AI and AI agents to automatically write, test, document, and deploy our software. This is fundamentally different from humans merely using AI as an assistant. This new paradigm of automated coding introduces a completely new set of best practices and radically changes our engineering roles and processes. Ultimately, we expect a massive increase in productivity, conservatively a 3x to 5x increase, but eventually far beyond what we can currently imagine.

Most of you have already participated in the January and February bootcamps designed to introduce these new concepts. Since then, we have formed several working groups to experiment with and implement AI Engineering. While adoption rates naturally vary, those who have embraced it are making remarkably fast progress. We are going to accelerate this pace, especially as we are seeing a strong appetite and demand from the teams themselves.

We know there will be some level of skepticism and fear. Skepticism arises not only from the negative stories you read or hear, but also because it is deeply counterintuitive to think that skills you have spent years developing can be automated. Fear stems from horror stories about job losses and the prospect of our roles becoming irrelevant.

We have also discussed the immediate consequences of this transformation. Two of the most critical are the need to expand our product portfolio to increase our addressable market, and the necessity to significantly scale our product management capabilities. Expanding our product portfolio is both an offensive and defensive strategy: it drives revenue through new offerings while increasing the pressure on competitors who are slower to adopt AI Engineering. Scaling product management is the logical next step to successfully harness our (we hope dramatically) increased productivity and manage these new product categories.

To accelerate our AI transformation, we have decided to implement some key organizational changes. First, I will step into the role of AI Transformation Officer, effective early March. My primary objective will be to drive this engineering

transformation forward, working directly alongside you to ensure constant feedback and rapid decision-making. In addition, B., CTO/CPO will put far focus on Product Management. He will reshape the Product Management team to capitalize on our increased productivity and, more importantly, lead the exploration of new product launches.

We believe this represents the biggest opportunity, and simultaneously the biggest threat, we have faced since the launch of our company. We have a tremendous advantage in our size and know-how, both of which are critical in this new AI-driven landscape. Our transformation hinges on making these cultural, business, and technological changes as swiftly and profoundly as possible.

As we enter the most challenging phase of our transformation, we will be significantly increasing our collaboration and communication across the group. I look forward to working with all of you toward our ambitious goals.

Sincerely,
Ali

PS: You can congratulate me on being promoted at last, and finding an actual job at Quod!

Resistance versus persistence Internal

A **Ali Pichvai** March 24th, 2026

The meeting with the tech leaders and advocates was somehow mostly either their skepticism or fear. Quoting articles about vibe coding isn't reality, we are not doing vibe coding. We are revamping the core of our engineering!

E **Engineering Lead** March 24th, 2026

Ali, you have to understand them, they are bright and performing. They read a lot of failures about the fact that "real software" is not addressable by AI. The statement about 'in 6 months I will forget coding' was particularly telling.

A

Ali Pichvai March 24th, 2026

I didn't say that I don't understand them. But they should at least read positive cases, not just cherry-pick negative stuff. Re: forgetfulness, I see the issue with juniors never learning the basics, but he is already senior and his brain will not forget the best practices of coding, will he?

For the juniors, we must keep recruiting. If we stop, the evolutionary algorithm to find the next generation of talent breaks. We must continue bringing bright young people.

E

Engineering Lead March 24th, 2026

I think that their critical thinking is a product of work, not true resistance. They will do it, they know this is the future and not imposed by us. The whole world is changing.

I overheard at coffee, [Engineer] saying to the team the benefits of Agentic and that this is the only way forward. Real experiences carry more than our views.

I have noticed that being junior or senior is not the reason someone is successful. It is more if you were good before AI, you are now even better as the technology amplifies your 'talent'.

A

Ali Pichvai March 24th, 2026

Agree!

Rome wasn't built in a day!

Internal Inbox

A

Ali Pichvai <a...@quodfinancial.com>
to @engineering ▾

March 27th, 2026

Dear all,

A few updates for the week:

1. **Working Groups:** We have started creating Working Groups aimed at setting up Agentic coding. These practical sessions will allow all of you to start working hands-on and concretely.
2. **QA Automation:** We have started QA work on the new Agent C. version and the upcoming Agent P. The first Agent is a test case management tool: generating test cases directly from Product requirements in Jira (using similarity models). At this stage, you will "curate" all the test cases, but behind the scenes, a reinforcement learning loop will use your feedback to increase precision. The second Agent is generating Python code for our QA test harness. QA has long been the limiting factor of our productivity, making this a revolutionary step (though, as an Iranian, "Revolution" never brings back good memories).
3. **Tool Usage:** [AI Provider 1] being rolled out rapidly, and we are already seeing initial usage. At this stage, don't worry about token utilization; the focus is entirely on adoption and experimentation. The economics will be addressed at a later stage. We are also taking out [AI Provider 2], they provide an agent which was liked. We intend to only use open source projects for agents (anyway they are all forks of VSCode).
4. **Metrics Tracking:** We have started implementing our initial metrics. These include the number of tokens used and the volume of AI-generated code compiled (versus Human coded). Measuring our progress will provide significant insight into our path forward. These metrics will be provided to you individually on a weekly basis so you have your own gauge. We use the same numbers collectively to measure progress and usage, ensuring transparency and leaving no room for misinterpretation.

Reading of the week: V. forwarded this excellent article to me, and I highly recommend it. In summary, the AI transformation vectors are:

- Agentic AI is an **organizational change** (which the author calls "institutional change"), not just coding or automating a few tasks. It impacts every aspect of the company, from engineering to sales, product management, and client support.
- Agentic AI is **not about cost reduction**; it is about revenue — specifically, increasing your revenue by being rapidly responsive to clients and market trends.

The famous "moat" is no longer your accumulated knowledge or software barriers; it is your velocity in adopting new trends and your time to market. It comes down to culture and mindset.

Sincerely,
Ali

The only constant is Change Internal

A

Ali Pichvai April 14th, 2026

For the meeting, I still have questions. We need to look at the organizational changes with Agentic. Scrum, Sprints, team sizes, creating a centralized unit for continuous tools R&D and how to measure our actual quality of code.

E

Engineering Lead April 14th, 2026

We agree, we are going to move to shorter sprints to have higher frequency (like my PC going from 100mhz to 1ghz, Pentium 1 to i5!).

Scrum will adjust, as we will make smaller teams, no more coding language division of tasks (but rather back-end versus UI/Client side), so for the Scrum and team size, the two are tied.

I think that there is going to be a rotation between roles too.

I would say that tools, there is overlap between some of the initiatives, but we all agree that we will have super-agents, and orchestrating as maintained by one single team.

For the rules/skills, we are centralising, but we need to see how to push it more.

For measuring the quality of the code, some of it can be delegated to an AI model, but we need to investigate further.

A

Ali Pichvai April 14th, 2026

OK. What about the metrics?

E

Engineering Lead April 14th, 2026

For token usage, we have it already, and it is actually easy. It is now in the telemetry / grafana dashboard.

Then comes the AI compilation versus human, this is on-going.

A

Ali Pichvai April 14th, 2026

I saw in the C. chat and I understand that not all developers have done what was asked of them to do. He said he is going to identify the AI vs Human code directly via CI/CD.

E

Engineering Lead April 14th, 2026

Maybe a last push and then we try the CI/CD route.

A

Ali Pichvai April 14th, 2026

And we can talk about Mythos (like the Greek Myths).

Myths and mythos

Internal Inbox

A

Ali Pichvai <a...@quodfinancial.com>
to @engineering ▾

April 16th, 2026

Dear all,

> *"There are decades where nothing happens; and there are weeks when decades happen*[2]*."*

Reading and reflecting on the Mythos announcement and the US government's immediate reaction, I have come to the firm conclusion that we have suddenly crossed

a frontier. The 'low-to-no' regulation phase of AI is coming to an end. If a model can create such existential threats to the Internet and the overall digital architecture of our economy, there must be a government-led response to address them.

In practice, this means the following for us:

- Much of the software (and hardware) we use will soon be changed or rapidly patched.
- Our finance clients will sooner or later, and I imagine sooner rather than later, demand that we use powerful models to find and fix our vulnerabilities, including those in our embedded software and hardware.
- The amount of work will be massive, as it involves not just a single threat, but often rewriting code to resolve the deep vulnerabilities we have accumulated over the years.

In this context, being early in our AI Transformation will not only provide a first-mover advantage, but will also define us as a truly secure software provider. Those who fail to adapt will be out of business.

In broader terms, and with the caveat that Mythos may be more marketing than an actual leap, security is going to be a major issue with AI. What was thought to be safe might not be. An MCP could act as a backdoor or an unseen SQL injection breach. For this reason, we are establishing a dedicated AI security unit within our broader security team.

Finally, we are making progress as teams begin to adopt Agentic workflows. A simple piece of advice to all those who are not yet comfortable: "Practice makes perfect."

Sincerely,
Ali

The belly of an architect

Internal Inbox

A **Ali Pichvai** <a...@quodfinancial.com>
to @engineering ▾

April 24th, 2026

Dear all,

As we progress toward our first major milestone on June 15th, a few critical ideas are becoming the foundation of our new organization. I want to share these main principles with you to spark your own research and reflection.

1. **Spec-Driven Development:** Currently, the main job of developers and testers is often to translate a well-defined task or subtask into code, a test case, or a script. AI will now become the implementer, but it lacks the implicit knowledge of our business and software. The only way to avoid bad outcomes and unfamiliar code is to write, or more precisely, curate, highly complete and detailed specifications, reaching a level of detail that makes the AI deterministic. This ensures precise guardrails are enforced at the specification level.

   In a way, this reverses the Agile methodology's emphasis on frequent verbal communication; we are now relying on well-written specifications and testing protocols (acceptance criteria), somewhat echoing the Waterfall era. Consequently, you must transition from task-driven coding to acting as software architects. You will control the quality and precision of the specs and the software planning, including the AI-driven breakdown of components, tasks, and subtasks. This is a new role for many, a responsibility previously delegated to a few software architects or senior staff with deep knowledge of our business logic. This presents a significant challenge that we must recognize and organize around.

2. **Centralized vs. Individualized Functions:** To maintain a coherent software architecture, we will centralize a set of core functions, such as Skills and Rules. I have noticed an eagerness among engineers to create their own individual agents, Skills, and Rules. However, this approach will result in a multitude of conflicting styles and incomplete rules rather than unified, logical software. Centralization is the only way to maintain a coherent, high-performance system. As argued above, your roles are evolving into business-driven software architects.

3. **Reviews:** We see two risks here. First, as we substantially increase coding productivity, we might occupy all our senior staff with reviewing poor or average AI-generated code, creating a new bottleneck. Second, there is a risk of losing sight of what constitutes "good" software (in terms of design and logic, rather than just syntax).

Our strategy, which will inevitably evolve alongside AI capabilities, is built on two core principles:

- **Impact-based reviewing:** We will determine during the specification stage whether a new development is highly impactful. All impactful features will require careful human review. For less impactful features, we will rely on sample-based checking.
- **Model separation:** We will never use the same "brain" (LLM) for both coding and reviewing. Since LLMs often attempt to please the user, using different models will lower the probability of "hallucinations" or false confirmations.

4. **Technology Dependency and Cost:** Initially, managing technology dependency and costs was pushed to a later phase of our AI Transformation plan, as our primary objective was establishing agentic engineering. However, AI is changing rapidly, and we are learning by doing. Token costs are quickly becoming an issue, escalating as major AI providers seek to recoup their massive capital and operational expenditures. We also recognize the risk of 'vendor lock-in' from relying too heavily on a single provider, which compromises our independence and makes switching costs prohibitive.

   To address this, we are rapidly shifting toward open-source agents and super-agents, to create a multi-model capability where we can easily switch providers on or off. Additionally, we will deploy different models for different purposes. This mix will include open-source models for variety (e.g., for reviews) and lower-cost models for simpler tasks. Furthermore, commercial providers typically push their latest, most expensive models, but we can achieve our goals with older models in all but the rarest cases. We will default to more cost-effective models to avoid exhausting our token quotas.

This email serves as more of a discussion paper, but as we approach the point of going fully agentic, we need to tackle fundamental and highly practical issues alike.

Sincerely,
Ali

Agentic is Logic. Software is Action. Internal

Inbox

A **Ali Pichvai** <a...@quodfinancial.com>
to @engineering ▾

May 1st, 2026

Dear all,

I would like to discuss a few key topics:

- **New Product and Lessons Learned:** This week, the dedicated V. team presented the first demo of the [New Product] with full server-side integration. Development was fully agentic. Building this first functional version took six weeks, and V. has analysed the initial lessons learned from the first half of the project. I invite you to read his presentation, which provides rich detail on our current insights (lessons learned) on a full agentic development.
- **Bottlenecks and Process Changes:** We have invested heavily in Scrum over the past six years. Thinking outside the box requires us to accept that this approach is now obsolete. Scrum was designed to manage human communication when building complex software. The underlying challenge, however, has always been how to manage resources, which have traditionally been our limiting factor. In a highly productive, agentic AI world, human resources are not the main bottleneck. At the same time, the agentic approach creates new bottlenecks. Code review can be one, QA is another, and Release Management (DevOps) is yet another, largely because managing this increased productivity requires significantly more compute power. We will address these shifts with new processes that not only account for these anticipated bottlenecks but also fundamentally change the way we produce our software.
- **Diversity, Token Economy, and Software as Part of Agentic Architecture:** As some of you may have noticed, [AI Provider] increased its prices this week. This was disguised as a reduction in allocated tokens, even for very simple prompts. This trend represents the near future of the industry, as every commercial provider is currently absorbing gargantuan compute costs per prompt and client. As discussed last week, this creates an imperative to adopt a multi-model approach, ensuring we use the most expensive models only for specific, complex tasks and delegate the rest to lower-cost or open-source alternatives.

An even more important consideration is that the overuse of AI is an architectural mistake and a security risk. AI is a probabilistic (i.e., non-deterministic) computing approach that is not highly performance-tuned. The correct method is to use agents as

orchestrators, leveraging their reasoning and human-like logical abilities to utilize highly efficient software for specific, high-performance actions. M. wrote an excellent paper this week containing a simple and powerful statement: **AI is Logic. Software is Action.** This provides clarity for our industry on why our software is a vital part of the new enterprise architecture, rather than a dead branch. It also holds very true for our work in agentic engineering.

In this regard, we already have many initiatives either in place or in the testing phase. For instance, Z. has developed a super-agent, which contains an optimized vector database including every Quod artifact (code, specs, support tickets, etc.) and an orchestrator that routes to different models (brains) for different steps of the agentic process. This will be tested very soon and looks extremely powerful; the combination of indexed knowledge and the use of different models provides in-depth knowledge of our code and reduces random code generation (hallucinations).

Sincerely,
Ali

Sakura (matsuri)? Internal Inbox

A **Ali Pichvai** <a...@quodfinancial.com>
to @engineering ▾

May 8th, 2026

Dear all,

We made excellent progress this week, demonstrating a few concrete examples of how AI will positively impact us:

- On the [Client] front, we are managing an exceptionally tight project: delivering a replacement for a 25-plus-year-old system in just four months. We have successfully mobilized an entire team and are seeing an approximately 50% increase in productivity. Given the late discovery and the sheer volume of development required, this productivity boost is undoubtedly one of the reasons we will succeed (though the primary driver remains the deep personal commitment of numerous team members).

- C., our new Product Manager (who joined about a month ago), presented his sleek [New Product 2] demo to us in London this week. He built it using a purely agentic approach, despite having no background in coding. His demo will be ported to [New Product 1], which will allow us to start pitching to prospective clients. I doubt we could have achieved such a rapid and impressive result without agentic tools. This perfectly illustrates our strategy of leveraging these capabilities to open up new market segments.

We are now nearly three weeks away from our milestone of full agentic engineering. We are observing excellent adoption. While we still see a small number of late adopters, the vast majority of you have embraced this new set of engineering methods and tools.

Much like a gardener, we have let this new plant grow unconstrained, but now it is time to shape the tree. We are going to centralize a core set of functions and tools to prevent a multitude of contradictory rules, skills, and tools from creating a chaotic engineering environment. This centralization will cover:

- Guardrails, Rules, and Skills Management
- Applications/Super-Agents (for indexing, coding, reviewing, etc.)
- Quality Assurance (led by C. and P.)

Finally, in two weeks, we will launch an audit led by B. (a friend, and more importantly, a brilliant mentor to a lot of young people and a thought leader). It aims to have an accurate state of where we are before the big leap.

Sincerely,
Ali

The decisions we have made were based on our size, the complexity and performance demands of our software suite, and the markets in which we operate. Furthermore, we developed our entire business on a single platform and have never acquired, nor been acquired by another software company. This unique position allows our dedicated R&D team to work exclusively on one unified platform. Another fact was our constant pursuit of productivity gains, which to be frank was short of impressive. It used to be single digits improvement, despite Scrum, automation, and trying to address all technical and organizational bottlenecks. AI was an answer to years of productivity improvement and experimentation.

We decided early on the following:

- **AI Transformation:** The choice was between an experimentation-based approach, slowly integrating AI into our company as mature technologies emerged, or a radical shift toward an agentic model. We chose the radical approach, confident that our teams would rapidly adapt. But we also think that being first would also provide competitive advantage.
- **A 12-Month Phased Approach:** We manage highly complex, high-performance software for existing clients, and we will continue onboarding new clients throughout this transformation. Because the risks of destabilizing the software or overwhelming the organization were real, we implemented a 12-month phased plan. We also wanted to allow everyone to build their own path toward AI engineering rather than turning it into a forced march. The most difficult part is the initial stage, which we are currently completing, involving the creation of an agentic engineering organization. This foundational phase is always the largest and hardest because it is essentially a new beginning: the social transformation. Subsequent phases will be more incremental, involving adjustments to our organizational structure and technology mix. Adoption follows a hockey-stick curve: a significant initial effort yields relatively small productivity gains at first, but this is followed by an acceleration phase where our teams will truly reap the rewards of AI.

Soros's book shows that he is not an Übermensch, but rather someone who excels at discovery. He uses a trial-and-error methodology to let ideas find their own pathways to results. This is just another (social) evolutionary system.

Here we tried to honestly share the beginning of our journey, without pretending that the process has been straightforward or free from resistance and mistakes on our part.

Our journey just started!

---

[1] AI Cambrian explosion in software — quodfinancial.com/ai-cambrian-explosion-software/

[2] Quote attributed to Vladimir Lenin.