

Management

The Principles of Quantum Team Management

This article is by [James Everingham](#), Head of Engineering at [Instagram](#).

I didn't intend to go into computer science, and I definitely didn't expect to become a manager. Instead, I started out thinking I'd be a laboratory scientist. In college, this led me to astronomy and physics, which prompted me to start writing software. And while these leaps felt intuitive, the subsequent shift into leading teams did not. In fact, I disliked my early engineering management job at **Oracle** so much that I took a less senior role at **Netscape**, just to start coding again.

But life is unpredictable. I discovered challenges at Netscape that were so important and interesting that I couldn't just stand by — I had to take the lead (which meant managing a team) and figure out how to do it in a way that made sense to me. **That's when it struck me that instead of approaching management like being a therapist (only with more process and politics to deal with), I could think of it from a problem-solving perspective.**

I started to design a management system the way I would design a machine or software system, with **few dependencies, single owners, minimal decision points**. Using this model, we immediately saw a jump in productivity, output, and happiness. Our ultimate email and news product, code-named [Grendel](#), [was the only piece of Netscape's massive Java rewrite that survived](#), and it remained a part of Mozilla's software for a decade.

Setting up your team the way you would set up a machine can give you a ton of leverage — as long as you realize how complicated and unpredictable the people in

that machine can be. This is where quantum mechanics (and my term ‘**quantum leadership**’) comes into play. As a discipline, it makes the unpredictable understandable. Similarly, by applying these quantum principles to management, you can find solutions to your team’s seemingly unsolvable problems.

Managing for Unimaginable Outcomes

There’s a fundamental principle of quantum physics called ‘superposition’ that asserts: if the state of an object is unknown and unchecked, the object exists in all possible states simultaneously. The Nobel prize-winning physicist [Erwin Schrödinger](#) proposed a theoretical experiment in which a cat, a vial of hydrocyanic acid, and a small amount of a radioactive substance are placed together inside of a box. If a single atom of the substance decays during the test period, it causes the hammer to break the vial and kill the cat. As long as the box remains unopened, however, the experimenter won’t know whether or not the sequence has been triggered. Thus, according to the principle of superposition, **the cat exists as both dead and alive simultaneously** until a measurement is taken — that is, until someone looks in the box.

Now, as someone who found it difficult to flip the cognitive switch from science to management, I understand that trying to map the implications of Schrödinger’s theoretical experiment to the everyday challenges of running an engineering team might feel slippery at best, but bear with me.

If observation does in fact affect outcome, and engineers and their completed projects can exhibit the property of superposition, then the dependent variable in my amateur managerial experiment is the state of success or failure of those projects, and the independent variable is the presence or absence of my (or the manager's) observation.

Every team’s project is a cat, and every manager has to constantly decide whether to look in the box at the risk of

killing it.

The observer effect is real in the workplace, and you can affect the outcome of any project as a manager simply by inserting yourself. Often, a manager will take their team into a room and say, “Here’s what we need to do,” or “Here’s what I’ve been thinking,” or “Here’s one way we can think about this...” as they start sketching on a whiteboard. They’re trying to add value. We always want to add value. But if you’re in any position of authority and you do this, you’ve just limited the number of outcomes and your path to success pretty dramatically.

Instead, if you simply outline the problem and what success looks like — let’s say it’s increasing revenue by 100% — all paths to success are still possible, including those you haven’t thought of yourself. It’s very likely that someone on your team will think of a better solution, but as soon as you say what you think, everyone gets a whole lot less creative.

I used to make this mistake a lot when I was a junior manager. I would give my team ideas to get them started, and as soon as I thought they were headed toward failure or a dead end, I’d stop them and say something to turn them around. It seemed like it was in everyone’s best interests to avoid the wrong solution, but a mentor of mine told me that my team would never get better if I didn’t let them learn from failure.

When I finally loosened my grip and let things go, I realized that my ideas were actually only right half of the time. The other half, my team’s ideas were far better than mine. I’d been an idiot for 10 years of my career, I realized.

Being a good manager is not about avoiding failure — it’s about enabling as many different paths forward as possible for as long as possible.

Getting Started

Okay, so how do you actually do this? You should ask questions (Socrates style) designed to influence your people the least and keep options open. For example:

What are some other ways we might be able to increase revenue?

Where are you running into roadblocks or friction?

Is there another route to where we want to go?

Let's say your goal is to increase revenue, and you're running into a roadblock where you can't get above a 2% clickthrough rate on an ad, for instance. You might ask: How much do we make per click? Why? The answer is probably, "That's just how it is." So then you could ask: What would it take for someone to pay more? Maybe the answer is more expensive inventory. Okay, then how do we make that happen?

If no one has any immediate ideas and all you're hearing is crickets, you have the option to open the box very slowly and carefully. You can drop a breadcrumb to lead the team to a next conclusion they can use as a jumping off point — a hint that doesn't give away what you think they should do. But the more breadcrumbs you drop, the narrower their thinking will become, so you have to be careful and thoughtful about what you reveal.

Having too many ideas in a room is an entirely different — and much better — problem. Still, it can stymie some teams. When this happens, ask questions about probability and make a matrix (one of the few times it's acceptable for you to whiteboard as a manager). What is the likelihood of success for each of the ideas proposed? Let your team discuss and stack rank proposals based on their probability of working. Groupthink will work to your advantage. Aggregate solutions are usually pretty good in these cases.

No matter what, whether there are too many ideas or too few, never supply your own opinion, judgment or ideas prematurely as a manager.



The 5 Principles of Quantum Team Management

To be successful quantum managers, we need to have vigilant awareness about our motivations for altering outcomes. **Common motivations include:** thinking your idea is the best, not trusting your people, or thinking you're not doing your job unless you're weighing in all the time. You actually want to follow a different set of instincts.

Allow yourself to be motivated by: observable metrics going one direction or the other, your team being visibly frustrated with the rate of progress, or a deserved lack of trust.

If we sense a project going sideways, it might be worth looking in the box and consciously changing the outcome. On the other hand, if the state of a project is entirely unknown or we sense that it's going well, it's better to allow for an unpredictable outcome. We do this by following these five principles:

1. Manage to multiple “states” as opposed to singular outcomes.

The wisdom of traditional mechanical management teaches us to guide our team toward a single outcome, but such a tactic assumes that managers understand projects better than the team members working on them. An effective quantum manager will do everything in their power to organize teams strategically and then step back. By avoiding prescribed paths, agreeing to outcomes, and providing each team with equal parts space and guidance, quantum managers set up their team for infinite success.

Focus on increasing the number of “states of success” if you can. Your cat doesn't have to be either dead or alive. Your cat could be happy, not doing that well, or doing pretty poorly but still alive. These are all other outcomes. The more states of success you can define, the more paths forward your team will see. Don't base success on one variable like revenue. Maybe one state is becoming profitable. Another might be getting a million people to use your product. It could be getting 100,000 people to use your product for one hour a day. Point out as many metrics of success as you can to give your team more milestones and momentum.

2. Be hyper-aware of the observer effect.

As we've already determined, observing a team will inherently change its state — a reality that can work for or against us. The art of quantum management lies in discovering methods for gathering information about the unobservable and preparing for all forms of success and failure while keeping the box closed. An effective quantum manager will resist super-“imposing” and instead ensure superposition.

By asking probing questions that challenge and change the way engineers think about a problem, we offer struggling teams a way to solve a seemingly unsolvable problem without potentially destroying the existing work that was already succeeding.

3. Know when to open the box.

Eventually all quantum managers will find themselves in a situation that leaves them no other choice but to open the box. Unfortunately, any predictions we make about the state of our cat will be wrong — unless we predict that it exists in a combination of any and every state simultaneously. In the case of a milestone plan, a team might be working so hard that they feel both exhilarated and exhausted at the same time. Hopefully, most well-oiled machines will achieve some degree of success, even in the worst case scenario. If not, and the cat dies, any previous need to keep the box closed disappears, and managers must immediately understand as much as they can about what happened.

So, what's the tactic? Use the time you might have spent generating your own ideas for how to solve problems — or generally stressing your team out with micromanagement — to devise non-judgmental, thought-provoking questions. It takes time to construct these. A lot of managers freestyle it and jump right in but end up saying too much or falling back on their own opinions. Asking questions designed to empower and not instruct requires a lot of forethought. **Consider putting 10 minutes on your calendar before any meeting to think through which questions will be helpful and won't interfere with your team's ability to win.**

4. Understand and create strategic entanglements.

The theory of **quantum entanglement** — what Einstein referred to as “spooky action at a distance” — finds that basic particles can be linked together in such a way that when something happens to one, it also happens to the other. Since most of us don't have a particle collider, we can instead use quantum management to entangle and motivate positive traits on our teams, a strategy that accelerates progress, increases quality, and

boosts morale. In short, if you put something positive in motion, generally more positive things will happen. Here are a few types of Quantum Management entanglements:

Accountability. Encourage an accountability entanglement by holding yourself to the same standards to which you hold your team; they'll mirror your behavior. It's human nature to mimic. People look for inspirational leaders. If you're rigorously accountable, your team becomes more accountable. Whatever you say you're going to do, just do it. You want to set up systems where you're creating behaviors tied to something in a spooky way. You see things organically unfold because you've changed your behavior. Also, you have to set up your machine so that someone CAN be accountable. I can't make someone guarantee delivery on something if it's dependent on someone else. Encourage people to be very clear about what they have to do and what they're going to do by being that way yourself first.

Empathy. Direct teams toward projects that will improve their day-to-day operations. The more your team interacts with their product, the more invested they'll be in the outcome. This tends to be easier if you're building consumer software, like at Instagram. I use Instagram, as does everyone on the team, so we feel the same pain as our users. But what if you're building some sort of enterprise system for long-distance truckers and you don't even drive? Look for even minor or tangential ways to empathize with your users. As an example, when I worked at **LiveOps**, we were building call center infrastructure. We looked to our own customer service department to get their feedback and pipe it back into our product development loop. Entangle yourself with your customers to understand what they're experiencing on a deeper level, why it needs to be better, and how you can improve it.

Good software is an act of empathy.

Incentive. Drive better results by linking your team's rewards to the success of the project, not just how they individually perform. Remember that rewards don't have to be monetary — global impact, visibility within the organization, career growth, or increased passion for the project can all incentivize engineers to perform better. A lot of managers forget how important their approval is. **The bulk of your team is there and working as hard as they can to please you.** That's just the way it is, so you have to let people know when you're pleased, when you're excited, when you're impressed. If you don't, and if your team doesn't get this type of reinforcement, you're literally limiting their potential.

Togetherness. Reduce stress, boost creativity and increase productivity by giving engineers the time and space to build camaraderie. Happier, compatible teams create long-lasting positive outcomes. I take my teams out to dinner and encourage them to go out together without me. **I seat my teams together because proximity builds relationships.** The more you know someone, the more you trust them. The more you trust the people you work with, the better the product you're going to build. Quantum managers encourage people to get to know each other on a personal level. They start meetings with people sharing about their lives, not just their work. They seed conversations about topics that are much more expansive than the task at hand so people truly get to know each other.

5. Embrace the challenge of self-observation.

As managers, it's difficult to recognize if we're using our quantum leadership techniques effectively because — like Schrödinger's cat — we exist in a state of success and failure simultaneously. Seeking constant feedback from those outside our quantum management box — like from our peers managing other segments of the business — allows us to stretch and grow without limiting our own outcomes.

Quantum Management Applied

In 2016, companies competing in any form of innovation can no longer pay employees for mindless, repetitive work. Employees have to create, to challenge, to think nimbly, to respond quickly, and to keep up in a rapidly changing environment. Yet, despite this transformation in the workforce, few organizations have invested in the development of corresponding management approaches, or have considered how traditional corporate infrastructure can negatively influence team dynamics.

Conventional wisdom tells us that it's the responsibility of bosses and managers to keep their team on course. But visible hierarchy tends to limit creativity. If no one knows each other's titles or levels of seniority, ideas speak for themselves.

As soon as a chain of authority is introduced, ideas no longer stand on their own merit.

I've witnessed dozens of extremely intelligent leaders make the same mistake because some part of them believed the myth that managers know best. This mistake — and the meeting in which it's made — looks nearly identical across organizations, regardless of the problem being solved or the people trying to solve it. Almost always, this mistake is made during a crisis when the manager meets with his team to address it. Perhaps revenue has dropped significantly in a short period of time. The manager reiterates the dire need to drive it back up, then waits for his engineering team to respond. There is a moment of silence. The team is thinking.

Now, I can't say with certainty why what happens next happens next. Maybe the manager is absolutely terrified that no one has a viable solution, or maybe she genuinely wants feedback on her idea, but either way, the mistake inevitably follows. The manager steps up to the whiteboard and sketches out a solution to their engagement problem, and, just by virtue of her seniority and the power behind her

suggestion, she effectively closes off the possibility of her team producing a superior solution. In essence, she (or he, of course) unwittingly kills the cat.

As a manager, I know this mistake intimately. I, too, have killed the cat many times. As a quantum manager, however, I also know the incredible rewards we reap when we resist the urge to prescribe a path, when we refuse to look in the box — even when we fear it's on fire and think we're the only one holding an extinguisher. I've found myself at this crossroads many times, most notably at my last startup, **Luminate**. Engagement had suddenly fallen off a cliff. At the time, I had plenty of reasons to think the box was on fire and enough experience working on engineering teams to believe that I could teach my team how to build an extinguisher.

As the founder of the company, I didn't want us to fail. As a human being, I wanted affirmation that my idea was a good one. **As a quantum manager, I bit my tongue.** Ultimately, I didn't engage in the brainstorming process because I knew my team would latch onto anything I offered. Instead, I just told them what winning would look like: a 300% increase in user engagement seemed like a lofty goal, but I pitched it anyways. A few days later, my team returned with a solution that I could never have imagined, and the results were stunning.

Though their strategy was remarkable from an engineer's perspective, what I learned from the exchange as a manager turned out to be even more valuable. By not offering my own idea, I enabled the creation of a better one. By not suggesting a destination, we all ended up somewhere extraordinary — a place I didn't even know we were going.

No cats were harmed in the making of this article.