4

EXPERIMENT

I come across many startups that are struggling to answer the following questions: Which customer opinions should we listen to, if any? How should we prioritize across the many features we could build? Which features are essential to the product's success and which are ancillary? What can be changed safely, and what might anger customers? What might please today's customers at the expense of tomorrow's? What should we work on next?

These are some of the questions teams struggle to answer if they have followed the "let's just ship a product and see what happens" plan. I call this the "just do it" school of entrepreneurship after Nike's famous slogan.[1] Unfortunately, if the plan is to see what happens, a team is guaranteed to succeed—at seeing what happens—but won't necessarily gain validated learning. This is one of the most important lessons of the scientific method: if you cannot fail, you cannot learn.

FROM ALCHEMY TO SCIENCE

The Lean Startup methodology reconceives a startup's efforts as experiments that test its strategy to see which parts are brilliant and which are crazy. A true experiment follows the scientific method. It begins with a clear hypothesis that makes predictions about what is supposed to happen. It then tests those predictions empirically. Just as scientific experimentation is informed by theory, startup experimentation is guided by the startup's vision. The goal of every startup experiment is to discover how to build a sustainable business around that vision.

Think Big, Start Small

Zappos is the world's largest online shoe store, with annual gross sales in excess of $1 billion. It is known as one of the most successful, customer-friendly e-commerce businesses in the world, but it did not start that way.

Founder Nick Swinmurn was frustrated because there was no central online site with a great selection of shoes. He envisioned a new and superior retail experience. Swinmurn could have waited a long time, insisting on testing his complete vision complete with warehouses, distribution partners, and the promise of significant sales. Many early e-commerce pioneers did just that, including infamous dot-com failures such as Webvan and Pets.com.

Instead, he started by running an experiment. His hypothesis was that customers were ready and willing to buy shoes online. To test it, he began by asking local shoe stores if he could take pictures of their inventory. In exchange for permission to take the pictures, he would post the pictures online and come back to buy the shoes at full price if a customer bought them online.

Zappos began with a tiny, simple product. It was designed to answer one question above all: is there already sufficient demand for a superior online shopping experience for shoes? However, a well-designed startup experiment like the one Zappos began with does more than test a single aspect of a business plan. In the course of testing this first assumption, many other assumptions were tested as well. To sell the shoes, Zappos had to interact with customers: taking payment, handling returns, and dealing with customer support. This is decidedly different from market research. If Zappos had relied on existing market research or conducted a survey, it could have asked what customers thought they wanted. By building a product instead, albeit a simple one, the company learned much more:

1. It had more accurate data about customer demand because it was observing real customer behavior, not asking hypothetical questions.
2. It put itself in a position to interact with real customers and learn about their needs. For example, the business plan might call for discounted pricing, but how are customer perceptions of the product affected by the discounting strategy?
3. It allowed itself to be surprised when customers behaved in unexpected ways, revealing information Zappos might not have known to ask about. For example, what if customers returned the shoes?

Zappos' initial experiment provided a clear, quantifiable outcome: either a sufficient number of customers would buy the shoes or they would not. It also put the company in a position to observe, interact with, and learn from real customers and partners. This qualitative learning is a necessary companion to quantitative testing. Although the early efforts were decidedly small-scale, that did not prevent the huge Zappos vision from being realized. In fact, in 2009 Zappos was acquired by the e-commerce giant Amazon.com for a reported $1.2 billion.[2]

For Long-Term Change, Experiment Immediately

Caroline Barlerin is a director in the global social innovation division at Hewlett-Packard (HP), a multinational company with more than three hundred thousand employees and more than $100 billion in annual sales. Caroline, who leads global community involvement, is a social entrepreneur working to get more of HP's employees to take advantage of the company's policy on volunteering.

Corporate guidelines encourage every employee to spend up to four hours a month of company time volunteering in his or her community; that volunteer work could take the form of any philanthropic effort: painting fences, building houses, or even using pro bono or work-based skills outside the company. Encouraging the latter type of volunteering was Caroline's priority. Because of its talent and values, HP's combined workforce has the potential to have a monumental positive impact. A designer could help a nonprofit with a new website design. A team of engineers could wire a school for Internet access.

Caroline's project is just beginning, and most employees do not know that this volunteering policy exists, and only a tiny fraction take advantage of it. Most of the volunteering has been of the low-impact variety, involving manual labor, even when the volunteers were highly trained experts. Barlerin's vision is to take the hundreds of thousands of employees in the company and transform them into a force for social good.

This is the kind of corporate initiative undertaken every day at companies around the world. It doesn't look like a startup by the conventional definition or what we see in the movies. On the surface it seems to be suited to traditional management and planning. However, I hope the discussion in Chapter 2 has prompted you to be a little suspicious. Here's how we might analyze this project using the Lean Startup framework.

Caroline's project faces extreme uncertainty: there had never been a volunteer campaign of this magnitude at HP before. How confident should she be that she knows the real reasons people aren't volunteering? Most important, how much does she really know about how to change the behavior of hundreds of thousand people in more than 170 countries? Barlerin's goal is to inspire her colleagues to make the world a better place. Looked at that way, her plan seems full of untested assumptions—and a lot of vision.

In accordance with traditional management practices, Barlerin is spending time planning, getting buy-in from various departments and other managers, and preparing a road map of initiatives for the first eighteen months of her project. She also has a strong accountability framework with metrics for the impact her project should have on the company over the next four years. Like many entrepreneurs, she has a business plan that lays out her intentions nicely. Yet despite all that work, she is—so far—creating one-off wins and no closer to knowing if her vision will be able to scale.

One assumption, for example, might be that the company's long-standing values included a commitment to improving the community but that recent economic trouble had resulted in an increased companywide strategic focus on short-term profitability. Perhaps longtime employees would feel a desire to reaffirm their values of giving back to the community by volunteering. A second assumption could be that they would find it more satisfying and therefore more sustainable to use their actual workplace skills in a volunteer capacity, which would have a greater impact on behalf of the organizations to which they donated their time. Also lurking within Caroline's plans are many practical assumptions about employees' willingness to take the time to volunteer, their level of commitment and desire, and the way to best reach them with her message.

The Lean Startup model offers a way to test these hypotheses rigorously, immediately, and thoroughly. Strategic planning takes months to complete; these experiments could begin immediately. By starting small, Caroline could prevent a tremendous amount of waste down the road without compromising her overall vision. Here's what it might look like if Caroline were to treat her project as an experiment.

Break It Down

The first step would be to break down the grand vision into its component parts. The two most important assumptions entrepreneurs make are what I call the value hypothesis and the growth hypothesis.

The *value hypothesis* tests whether a product or service really delivers value to customers once they are using it. What's a good indicator that employees find donating their time valuable? We could

survey them to get their opinion, but that would not be very accurate because most people have a hard time assessing their feelings objectively.

Experiments provide a more accurate gauge. What could we see in real time that would serve as a proxy for the value participants were gaining from volunteering? We could find opportunities for a small number of employees to volunteer and then look at the retention rate of those employees. How many of them sign up to volunteer again? When an employee voluntarily invests their time and attention in this program, that is a strong indicator that they find it valuable.

For the *growth hypothesis*, which tests how new customers will discover a product or service, we can do a similar analysis. Once the program is up and running, how will it spread among the employees, from initial early adopters to mass adoption throughout the company? A likely way this program could expand is through viral growth. If that is true, the most important thing to measure is behavior: would the early participants actively spread the word to other employees?

In this case, a simple experiment would involve taking a very small number—a dozen, perhaps—of existing long-term employees and providing an exceptional volunteer opportunity for them. Because Caroline's hypothesis was that employees would be motivated by their desire to live up to HP's historical commitment to community service, the experiment would target employees who felt the greatest sense of disconnect between their daily routine and the company's expressed values. The point is not to find the average customer but to find *early adopters:* the customers who feel the need for the product most acutely. Those customers tend to be more forgiving of mistakes and are especially eager to give feedback.

Next, using a technique I call the *concierge minimum viable product* (described in detail in Chapter 6), Caroline could make sure the first few participants had an experience that was as good as she could make it, completely aligned with her vision. Unlike in a focus group, her goal would be to measure what the customers actually did. For example, how many of the first volunteers actually complete their volunteer assignments? How many volunteer a second time? How many are willing to recruit a colleague to participate in a subsequent volunteer activity?

Additional experiments can expand on this early feedback and learning. For example, if the growth model requires that a certain

percentage of participants share their experiences with colleagues and encourage their participation, the degree to which that takes place can be tested even with a very small sample of people. If ten people complete the first experiment, how many do we expect to volunteer again? If they are asked to recruit a colleague, how many do we expect will do so? Remember that these are supposed to be the kinds of early adopters with the most to gain from the program.

Put another way, what if all ten early adopters decline to volunteer again? That would be a highly significant—and very negative—result. If the numbers from such early experiments don't look promising, there is clearly a problem with the strategy. That doesn't mean it's time to give up; on the contrary, it means it's time to get some immediate qualitative feedback about how to improve the program. Here's where this kind of experimentation has an advantage over traditional market research. We don't have to commission a survey or find new people to interview. We already have a cohort of people to talk to as well as knowledge about their actual behavior: the participants in the initial experiment.

This entire experiment could be conducted in a matter of weeks, less than one-tenth the time of the traditional strategic planning process. Also, it can happen in parallel with strategic planning while the plan is still being formulated. Even when experiments produce a negative result, those failures prove instructive and can influence the strategy. For example, what if no volunteers can be found who are experiencing the conflict of values within the organization that was such an important assumption in the business plan? If so, congratulations: it's time to pivot (a concept that is explored in more detail in Chapter 8).[3]

AN EXPERIMENT IS A PRODUCT

In the Lean Startup model, an experiment is more than just a theoretical inquiry; it is also a first product. If this or any other experiment is successful, it allows the manager to get started with his or her campaign: enlisting early adopters, adding employees to each further experiment or iteration, and eventually starting to build a product. By the time that product is ready to be distributed widely, it will already have established customers. It will have solved real problems and offer detailed specifications for what needs to be built. Unlike a traditional strategic planning or market research process, this specification will be rooted in feedback on what is working today rather than in anticipation of what might work tomorrow.

To see this in action, consider an example from Kodak. Kodak's history is bound up with cameras and film, but today it also operates a substantial online business called Kodak Gallery. Mark Cook is Kodak Gallery's vice president of products, and he is working to change Kodak Gallery's culture of development to embrace experimentation.

Mark explained, "Traditionally, the product manager says, 'I just want this.' In response, the engineer says, 'I'm going to build it.' Instead, I try to push my team to first answer four questions:

1. Do consumers recognize that they have the problem you are trying to solve?
2. If there was a solution, would they buy it?
3. Would they buy it from us?
4. Can we build a solution for that problem?"

The common tendency of product development is to skip straight to the fourth question and build a solution before confirming that customers have the problem. For example, Kodak Gallery offered wedding cards with gilded text and graphics on its site. Those designs were popular with customers who were getting married, and so the team redesigned the cards to be used at other special occasions, such as for holidays. The market research and design process indicated that

customers would like the new cards, and that finding justified the significant effort that went into creating them.

Days before the launch, the team realized the cards were too difficult to understand from their depiction on the website; people couldn't see how beautiful they were. They were also hard to produce. Cook realized that they had done the work backward. He explained, "Until we could figure out how to sell and make the product, it wasn't worth spending any engineering time on."

Learning from that experience, Cook took a different approach when he led his team through the development of a new set of features for a product that makes it easier to share photos taken at an event. They believed that an online "event album" would provide a way for people who attended a wedding, a conference, or another gathering to share photos with other attendees. Unlike other online photo sharing services, Kodak Gallery's event album would have strong privacy controls, assuring that the photos would be shared only with people who attended the same event.

In a break with the past, Cook led the group through a process of identifying risks and assumptions before building anything and then testing those assumptions experimentally.

There were two main hypotheses underlying the proposed event album:

1. The team assumed that customers would want to create the albums in the first place.
2. It assumed that event participants would upload photos to event albums created by friends or colleagues.

The Kodak Gallery team built a simple prototype of the event album. It lacked many features—so many, in fact, that the team was reluctant to show it to customers. However, even at that early stage, allowing customers to use the prototype helped the team refute their hypotheses. First, creating an album was not as easy as the team had predicted; *none* of the early customers were able to create one. Further, customers complained that the early product version lacked essential features.

Those negative results demoralized the team. The usability problems frustrated them, as did customer complains about missing features, many of which matched the original road map. Cook explained that

even though the product was missing features, the project was not a failure. The initial product—flaws and all—confirmed that users did have the desire to create event albums, which was extremely valuable information. Where customers complained about missing features, this suggested that the team was on the right track. The team now had early evidence that those features were in fact important. What about features that were on the road map but that customers didn't complain about? Maybe those features weren't as important as they initially seemed.

Through a beta launch the team continued to learn and iterate. While the early users were enthusiastic and the numbers were promising, the team made a major discovery. Through the use of online surveying tool KISSinsights, the team learned that many customers wanted to be able to arrange the order of pictures before they would invite others to contribute. Knowing they weren't ready to launch, Cook held off his division's general manager by explaining how iterating and experimenting before beginning the marketing campaign would yield far better results. In a world where marketing launch dates were often set months in advance, waiting until the team had really solved the problem was a break from the past.

This process represented a dramatic change for Kodak Gallery; employees were used to being measured on their progress at completing tasks. As Cook says, "Success is not delivering a feature; success is learning how to solve the customer's problem."[4]

THE VILLAGE LAUNDRY SERVICE

In India, due to the cost of a washing machine, less than seven percent of the population have one in their homes. Most people either hand wash their clothing at home or pay a Dhobi to do it for them. Dhobis take the clothes to the nearest river, wash them in the river water, bang them against rocks to get them clean, and hang them to dry, which takes two to seven days. The result? Clothes are returned in about ten days and are probably not that clean.

Akshay Mehra had been working at Procter & Gamble Singapore for eight years when he sensed an opportunity. As the brand manager of the Tide and Pantene brands for India and ASEAN countries, he thought he could make laundry services available to people who previously could not afford them. Returning to India, Akshay joined the Village Laundry Services (VLS), created by Innosight Ventures. VLS began a series of experiments to test its business assumptions.

For their first experiment, VLS mounted a consumer-grade laundry machine on the back of a pickup truck parked on a street corner in Bangalore. The experiment cost less than $8,000 and had the simple goal of proving that people would hand over their laundry and pay to have it cleaned. The entrepreneurs did not clean the laundry on the truck, which was more for marketing and show, but took it off-site to be cleaned and brought it back to their customers by the end of the day.

The VLS team continued the experiment for a week, parking the truck on different street corners, digging deeper to discover all they could about their potential customers. They wanted to know how they could encourage people to come to the truck. Did cleaning speed matter? Was cleanliness a concern? What were people asking for when they left their laundry with them? They discovered that customers were happy to give them their laundry to clean. However, those customers were suspicious of the washing machine mounted on the back of the truck, concerned that VLS would take their laundry and run. To address that concern, VLS created a slightly more substantial mobile cart that looked more like a kiosk.

VLS also experimented with parking the carts in front of a local minimarket chain. Further iterations helped VLS figure out which services people were most interested in and what price they were willing to pay. They discovered that customers often wanted their clothes ironed and were willing to pay double the price to get their laundry back in four hours rather than twenty-four hours.

As a result of those early experiments, VLS created an end product that was a three-foot by four-foot mobile kiosk that included an energy-efficient, consumer-grade washing machine, a dryer, and an extra-long extension cord. The kiosk used Western detergents and was supplied daily with fresh clean water delivered by VLS.

Since then, the Village Laundry Service has grown substantially, with fourteen locations operational in Bangalore, Mysore, and Mumbai. As CEO Akshay Mehra shared with me, “We have serviced 116,000 kgs. in 2010 (vs. 30,600 kg. in 2009). And almost 60 percent of the business is coming from repeat customers. We have serviced more than 10,000 customers in the past year alone across all the outlets.”[5]

A LEAN STARTUP IN GOVERNMENT?

On July 21, 2010, President Obama signed the Dodd–Frank Wall Street Reform and Consumer Protection Act into law. One of its landmark provisions created a new federal agency, the Consumer Federal Protection Bureau (CFPB). This agency is tasked with protecting American citizens from predatory lending by financial services companies such as credit card companies, student lenders, and payday loan offices. The plan calls for it to accomplish this by setting up a call center where trained case workers will field calls directly from the public.

Left to its own devices, a new government agency would probably hire a large staff with a large budget to develop a plan that is expensive and time-consuming. However, the CFPB is considering doing things differently. Despite its $500 million budget and high-profile origins, the CPFB is really a startup.

President Obama tasked his chief technology officer, Aneesh Chopra, with collecting ideas for how to set up the new startup agency, and that is how I came to be involved. On one of Chopra's visits to Silicon Valley, he invited a number of entrepreneurs to make suggestions for ways to cultivate a startup mentality in the new agency. In particular, his focus was on leveraging technology and innovation to make the agency more efficient, cost-effective, and thorough.

My suggestion was drawn straight from the principles of this chapter: treat the CFPB as an experiment, identify the elements of the plan that are assumptions rather than facts, and figure out ways to test them. Using these insights, we could build a minimum viable product and have the agency up and running—on a micro scale—long before the official plan was set in motion.

The number one assumption underlying the current plan is that once Americans know they can call the CFPB for help with financial fraud and abuse, there will be a significant volume of citizens who do that. This sounds reasonable, as it is based on market research about the amount of fraud that affects Americans each year. However, despite all that research, it is still an assumption. If the actual call volume differs

markedly from that in the plan, it will require significant revision. What if Americans who are subjected to financial abuse don't view themselves as victims and therefore don't seek help? What if they have very different notions of what problems are important? What if they call the agency seeking help for problems that are outside its purview?

Once the agency is up and running with a $500 million budget and a correspondingly large staff, altering the plan will be expensive and time-consuming, but why wait to get feedback? To start experimenting immediately, the agency could start with the creation of a simple hotline number, using one of the new breed of low-cost and fast setup platforms such as Twilio. With a few hours' work, they could add simple voice prompts, offering callers a menu of financial problems to choose from. In the first version, the prompts could be drawn straight from the existing research. Instead of a caseworker on the line, each prompt could offer the caller useful information about how to solve her or his problem.

Instead of marketing this hotline to the whole country, the agency could run the experiment in a much more limited way: start with a small geographic area, perhaps as small as a few city blocks, and instead of paying for expensive television or radio advertising to let people know about the service, use highly targeted advertising. Flyers on billboards, newspaper advertisements to those blocks, or specially targeted online ads would be a good start. Since the target area is so small, they could afford to pay a premium to create a high level of awareness in the target zone. The total cost would remain quite small.

As a comprehensive solution to the problem of financial abuse, this minimum viable product is not very good compared with what a $500 million agency could accomplish. But it is also not very expensive. This product could be built in a matter of days or weeks, and the whole experiment probably would cost only a few thousand dollars.

What we would learn from this experiment would be invaluable. On the basis of the selections of those first callers, the agency could immediately start to get a sense of what kinds of problems Americans believe they have, not just what they "should" have. The agency could begin to test marketing messages: What motivates people to call? It could start to extrapolate real-world trends: What percentage of people in the target area actually call? The extrapolation would not be perfect, but it would establish a baseline behavior that would be far more accurate than market research.

Most important, this product would serve as a seed that could germinate into a much more elaborate service. With this beginning, the agency could engage in a continuous process of improvement, slowly but surely adding more and better solutions. Eventually, it would staff the hotline with caseworkers, perhaps at first addressing only one category of problems, to give the caseworkers the best chance of success. By the time the official plan was ready for implementation, this early service could serve as a real-world template.

The CFPB is just getting started, but already they are showing signs of following an experimental approach. For example, instead of doing a geographically limited rollout, they are segmenting their first products by use case. They have established a preliminary order of financial products to provide consumer services for, with credit cards coming first. As their first experiment unfolds, they will have the opportunity to closely monitor all of the other complaints and consumer feedback they receive. This data will influence the depth, breadth, and sequence of future offerings.

As David Forrest, the CFPB's chief technology officer, told me, "Our goal is to give American citizens an easy way to tell us about the problems they see out there in the consumer financial marketplace. We have an opportunity to closely monitor what the public is telling us and react to new information. Markets change all the time and our job is to change with them."[6]

The entrepreneurs and managers profiled in this book are smart, capable, and extremely results-oriented. In many cases, they are in the midst of building an organization in a way consistent with the best practices of current management thinking. They face the same challenges in both the public and private sectors, regardless of industry. As we've seen, even the seasoned managers and executives at the world's best-run companies struggle to consistently develop and launch innovative new products.

Their challenge is to overcome the prevailing management thinking that puts its faith in well-researched plans. Remember, planning is a tool that only works in the presence of a long and stable operating history. And yet, do any of us feel that the world around us is getting more and more stable every day? Changing such a mind-set is hard

but critical to startup success. My hope is that this book will help managers and entrepreneurs make this change.

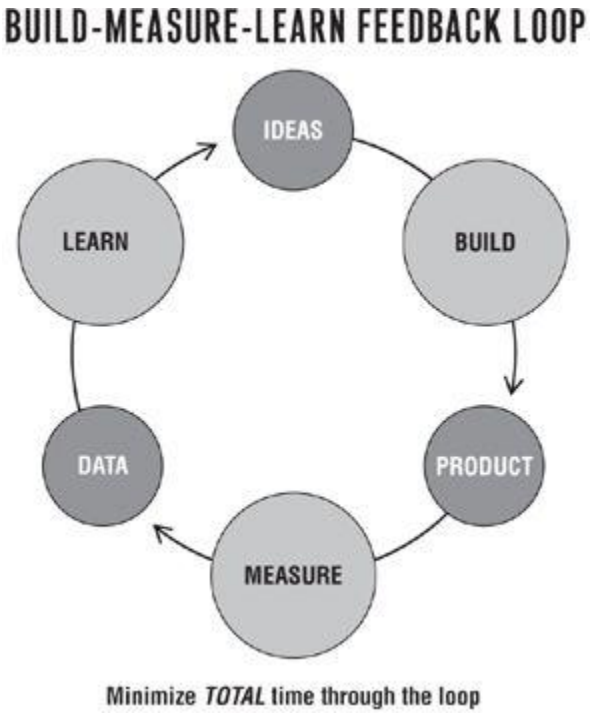
Part Two

STEER

How Vision Leads to Steering

At its heart, a startup is a catalyst that transforms ideas into products. As customers interact with those products, they generate feedback and data. The feedback is both qualitative (such as what they like and don't like) and quantitative (such as how many people use it and find it valuable). As we saw in Part One, the products a startup builds are really experiments; the learning about how to build a sustainable business is the outcome of those experiments. For startups, that information is much more important than dollars, awards, or mentions in the press, because it can influence and reshape the next set of ideas.

We can visualize this three-step process with this simple diagram:



This Build-Measure-Learn feedback loop is at the core of the Lean Startup model. In Part Two, we will examine it in great detail.

Many people have professional training that emphasizes one element of this feedback loop. For engineers, it's learning to build things as efficiently as possible. Some managers are experts at strategizing and learning at the whiteboard. Plenty of entrepreneurs focus their energies on the individual nouns: having the best product idea or the best-designed initial product or obsessing over data and metrics. The truth is that none of these activities by itself is of paramount importance. Instead, we need to focus our energies on minimizing the *total* time through this feedback loop. This is the essence of steering a startup and is the subject of Part Two. We will walk through a complete turn of the Build-Measure-Learn feedback loop, discussing each of the components in detail.

The purpose of Part One was to explore the importance of learning as the measure of progress for a startup. As I hope is evident by now, by focusing our energies on validated learning, we can avoid much of the waste that plagues startups today. As in lean manufacturing, learning where and when to invest energy results in saving time and money.

To apply the scientific method to a startup, we need to identify which hypotheses to test. I call the riskiest elements of a startup's plan, the parts on which everything depends, *leap-of-faith* assumptions. The two most important assumptions are the value hypothesis and the growth hypothesis. These give rise to tuning variables that control a startup's engine of growth. Each iteration of a startup is an attempt to rev this engine to see if it will turn. Once it is running, the process repeats, shifting into higher and higher gears.

Once clear on these leap-of-faith assumptions, the first step is to enter the Build phase as quickly as possible with a minimum viable product (MVP). The MVP is that version of the product that enables a full turn of the Build-Measure-Learn loop with a minimum amount of effort and the least amount of development time. The minimum viable product lacks many features that may prove essential later on. However, in some ways, creating a MVP requires extra work: we must be able to measure its impact. For example, it is inadequate to build a prototype that is evaluated solely for internal quality by engineers and designers. We also need to get it in front of potential customers to

gauge their reactions. We may even need to try selling them the prototype, as we'll soon see.

When we enter the Measure phase, the biggest challenge will be determining whether the product development efforts are leading to real progress. Remember, if we're building something that nobody wants, it doesn't much matter if we're doing it on time and on budget. The method I recommend is called *innovation accounting*, a quantitative approach that allows us to see whether our engine-tuning efforts are bearing fruit. It also allows us to create *learning milestones*, which are an alternative to traditional business and product milestones. Learning milestones are useful for entrepreneurs as a way of assessing their progress accurately and objectively; they are also invaluable to managers and investors who must hold entrepreneurs accountable. However, not all metrics are created equal, and in Chapter 7 I'll clarify the danger of *vanity metrics* in contrast to the nuts-and-bolts usefulness of *actionable metrics*, which help to analyze customer behavior in ways that support innovation accounting.

Finally, and most important, there's the *pivot*. Upon completing the Build-Measure-Learn loop, we confront the most difficult question any entrepreneur faces: whether to pivot the original strategy or persevere. If we've discovered that one of our hypotheses is false, it is time to make a major change to a new strategic hypothesis.

The Lean Startup method builds capital-efficient companies because it allows startups to recognize that it's time to pivot sooner, creating less waste of time and money. Although we write the feedback loop as Build-Measure-Learn because the activities happen in that order, our planning really works in the reverse order: we figure out what we need to learn, use innovation accounting to figure out what we need to measure to know if we are gaining validated learning, and then figure out what product we need to build to run that experiment and get that measurement. All of the techniques in Part Two are designed to minimize the total time through the Build-Measure-Learn feedback loop.

5

LEAP

In 2004, three college sophomores arrived in Silicon Valley with their fledgling college social network. It was live on a handful of college campuses. It was not the market-leading social network or even the first college social network; other companies had launched sooner and with more features. With 150,000 registered users, it made very little revenue, yet that summer they raised their first $500,000 in venture capital. Less than a year later, they raised an additional $12.7 million.

Of course, by now you've guessed that these three college sophomores were Mark Zuckerberg, Dustin Moskovitz, and Chris Hughes of Facebook. Their story is now world famous. Many things about it are remarkable, but I'd like to focus on only one: how Facebook was able to raise so much money when its actual usage was so small.[1]

By all accounts, what impressed investors the most were two facts about Facebook's early growth. The first fact was the raw amount of time Facebook's active users spent on the site. More than half of the users came back to the site every single day.[2] This is an example of how a company can validate its value hypothesis—that customers find the product valuable. The second impressive thing about Facebook's early traction was the rate at which it had taken over its first few college campuses. The rate of growth was staggering: Facebook launched on February 4, 2004, and by the end of that month almost three-quarters of Harvard's undergraduates were using it, without a dollar of marketing

7

MEASURE

At the beginning, a startup is little more than a model on a piece of paper. The financials in the business plan include projections of how many customers the company expects to attract, how much it will spend, and how much revenue and profit that will lead to. It's an ideal that's usually far from where the startup is in its early days.

A startup's job is to (1) rigorously measure where it is right now, confronting the hard truths that assessment reveals, and then (2) devise experiments to learn how to move the real numbers closer to the ideal reflected in the business plan.

Most products—even the ones that fail—do not have zero traction. Most products have some customers, some growth, and some positive results. One of the most dangerous outcomes for a startup is to bumble along in the land of the living dead. Employees and entrepreneurs tend to be optimistic by nature. We want to keep believing in our ideas even when the writing is on the wall. This is why the myth of perseverance is so dangerous. We all know stories of epic entrepreneurs who managed to pull out a victory when things seemed incredibly bleak. Unfortunately, we don't hear stories about the countless nameless others who persevered too long, leading their companies to failure.

WHY SOMETHING AS SEEMINGLY DULL AS ACCOUNTING WILL CHANGE YOUR LIFE

People are accustomed to thinking of accounting as dry and boring, a necessary evil used primarily to prepare financial reports and survive audits, but that is because accounting is something that has become taken for granted. Historically, under the leadership of people such as Alfred Sloan at General Motors, accounting became an essential part of the method of exerting centralized control over far-flung divisions. Accounting allowed GM to set clear milestones for each of its divisions and then hold each manager accountable for his or her division's success in reaching those goals. All modern corporations use some variation of that approach. Accounting is the key to their success.

Unfortunately, standard accounting is not helpful in evaluating entrepreneurs. Startups are too unpredictable for forecasts and milestones to be accurate.

I recently met with a phenomenal startup team. They are well financed, have significant customer traction, and are growing rapidly. Their product is a leader in an emerging category of enterprise software that uses consumer marketing techniques to sell into large companies. For example, they rely on employee-to-employee viral adoption rather than a traditional sales process, which might target the chief information officer or the head of information technology (IT). As a result, they have the opportunity to use cutting-edge experimental techniques as they constantly revise their product. During the meeting, I asked the team a simple question that I make a habit of asking startups whenever we meet: are you making your product better? They always say yes. Then I ask: how do you know? I invariably get this answer: well, we are in engineering and we made a number of changes last month, and our customers seem to like them, and our overall numbers are higher this month. We must be on the right track.

This is the kind of storytelling that takes place at most startup board meetings. Most milestones are built the same way: hit a certain product milestone, maybe talk to a few customers, and see if the numbers go

up. Unfortunately, this is not a good indicator of whether a startup is making progress. How do we know that the changes we've made are related to the results we're seeing? More important, how do we know that we are drawing the right lessons from those changes?

To answer these kinds of questions, startups have a strong need for a new kind of accounting geared specifically to disruptive innovation. That's what innovation accounting is.

An Accountability Framework That Works Across Industries

Innovation accounting enables startups to prove objectively that they are learning how to grow a sustainable business. Innovation accounting begins by turning the leap-of-faith assumptions discussed in Chapter 5 into a quantitative financial model. Every business plan has some kind of model associated with it, even if it's written on the back of a napkin. That model provides assumptions about what the business will look like at a successful point in the future.

For example, the business plan for an established manufacturing company would show it growing in proportion to its sales volume. As the profits from the sales of goods are reinvested in marketing and promotions, the company gains new customers. The rate of growth depends primarily on three things: the profitability of each customer, the cost of acquiring new customers, and the repeat purchase rate of existing customers. The higher these values are, the faster the company will grow and the more profitable it will be. These are the drivers of the company's growth model.

By contrast, a marketplace company that matches buyers and sellers such as eBay will have a different growth model. Its success depends primarily on the network effects that make it the premier destination for both buyers and sellers to transact business. Sellers want the marketplace with the highest number of potential customers. Buyers want the marketplace with the most competition among sellers, which leads to the greatest availability of products and the lowest prices. (In economics, this sometimes is called supply-side increasing returns and demand-side increasing returns.) For this kind of startup, the important

thing to measure is that the network effects are working, as evidenced by the high retention rate of new buyers and sellers. If people stick with the product with very little attrition, the marketplace will grow no matter how the company acquires new customers. The growth curve will look like a compounding interest table, with the rate of growth depending on the “interest rate” of new customers coming to the product.

Though these two businesses have very different drivers of growth, we can still use a common framework to hold their leaders accountable. This framework supports accountability even when the model changes.

HOW INNOVATION ACCOUNTING WORKS—THREE LEARNING MILESTONES

Innovation accounting works in three steps: first, use a minimum viable product to establish real data on where the company is right now. Without a clear-eyed picture of your current status—no matter how far from the goal you may be—you cannot begin to track your progress.

Second, startups must attempt to tune the engine from the baseline toward the ideal. This may take many attempts. After the startup has made all the micro changes and product optimizations it can to move its baseline toward the ideal, the company reaches a decision point. That is the third step: pivot or persevere.

If the company is making good progress toward the ideal, that means it's learning appropriately and using that learning effectively, in which case it makes sense to continue. If not, the management team eventually must conclude that its current product strategy is flawed and needs a serious change. When a company pivots, it starts the process all over again, reestablishing a new baseline and then tuning the engine from there. The sign of a successful pivot is that these engine-tuning activities are more productive after the pivot than before.

Establish the Baseline

For example, a startup might create a complete prototype of its product and offer to sell it to real customers through its main marketing channel. This single MVP would test most of the startup's assumptions and establish baseline metrics for each assumption simultaneously. Alternatively, a startup might prefer to build separate MVPs that are aimed at getting feedback on one assumption at a time. Before building the prototype, the company might perform a smoke test with its marketing materials. This is an old direct marketing technique in which customers are given the opportunity to preorder a product that has not

yet been built. A smoke test measures only one thing: whether customers are interested in trying a product. By itself, this is insufficient to validate an entire growth model. Nonetheless, it can be very useful to get feedback on this assumption before committing more money and other resources to the product.

These MVPs provide the first example of a *learning milestone*. An MVP allows a startup to fill in real baseline data in its growth model—conversion rates, sign-up and trial rates, customer lifetime value, and so on—and this is valuable as the foundation for learning about customers and their reactions to a product even if that foundation begins with extremely bad news.

When one is choosing among the many assumptions in a business plan, it makes sense to test the riskiest assumptions first. If you can't find a way to mitigate these risks toward the ideal that is required for a sustainable business, there is no point in testing the others. For example, a media business that is selling advertising has two basic assumptions that take the form of questions: Can it capture the attention of a defined customer segment on an ongoing basis? and can it sell that attention to advertisers? In a business in which the advertising rates for a particular customer segment are well known, the far riskier assumption is the ability to capture attention. Therefore, the first experiments should involve content production rather than advertising sales. Perhaps the company will produce a pilot episode or issue to see how customers engage.

Tuning the Engine

Once the baseline has been established, the startup can work toward the second learning milestone: tuning the engine. Every product development, marketing, or other initiative that a startup undertakes should be targeted at improving one of the drivers of its growth model. For example, a company might spend time improving the design of its product to make it easier for new customers to use. This presupposes that the *activation rate* of new customers is a driver of growth and that its baseline is lower than the company would like. To demonstrate validated learning, the design changes must improve the activation rate of new customers. If they do not, the new design should be judged a

failure. This is an important rule: a good design is one that changes customer behavior for the better.

Compare two startups. The first company sets out with a clear baseline metric, a hypothesis about what will improve that metric, and a set of experiments designed to test that hypothesis. The second team sits around debating what would improve the product, implements several of those changes at once, and celebrates if there is any positive increase in any of the numbers. Which startup is more likely to be doing effective work and achieving lasting results?

Pivot or Persevere

Over time, a team that is learning its way toward a sustainable business will see the numbers in its model rise from the horrible baseline established by the MVP and converge to something like the ideal one established in the business plan. A startup that fails to do so will see that ideal recede ever farther into the distance. When this is done right, even the most powerful reality distortion field won't be able to cover up this simple fact: if we're not moving the drivers of our business model, we're not making progress. That becomes a sure sign that it's time to pivot.

INNOVATION ACCOUNTING AT IMVU

Here's what innovation accounting looked like for us in the early days of IMVU. Our minimum viable product had many defects and, when we first released it, extremely low sales. We naturally assumed that the lack of sales was related to the low quality of the product, so week after week we worked on improving the quality of the product, trusting that our efforts were worthwhile. At the end of each month, we would have a board meeting at which we would present the results. The night before the board meeting, we'd run our standard analytics, measuring conversion rates, customer counts, and revenue to show what a good job we had done. For several meetings in a row, this caused a last-minute panic because the quality improvements were not yielding any change in customer behavior. This led to some frustrating board meetings at which we could show great product "progress" but not much in the way of business results. After a while, rather than leave it to the last minute, we began to track our metrics more frequently, tightening the feedback loop with product development. This was even more depressing. Week in, week out, our product changes were having no effect.

Improving a Product on Five Dollars a Day

We tracked the "funnel metrics" behaviors that were critical to our engine of growth: customer registration, the download of our application, trial, repeat usage, and purchase. To have enough data to learn, we needed just enough customers using our product to get real numbers for each behavior. We allocated a budget of five dollars per day: enough to buy clicks on the then-new Google AdWords system. In those days, the minimum you could bid for a click was 5 cents, but there was no overall minimum to your spending. Thus, we could afford

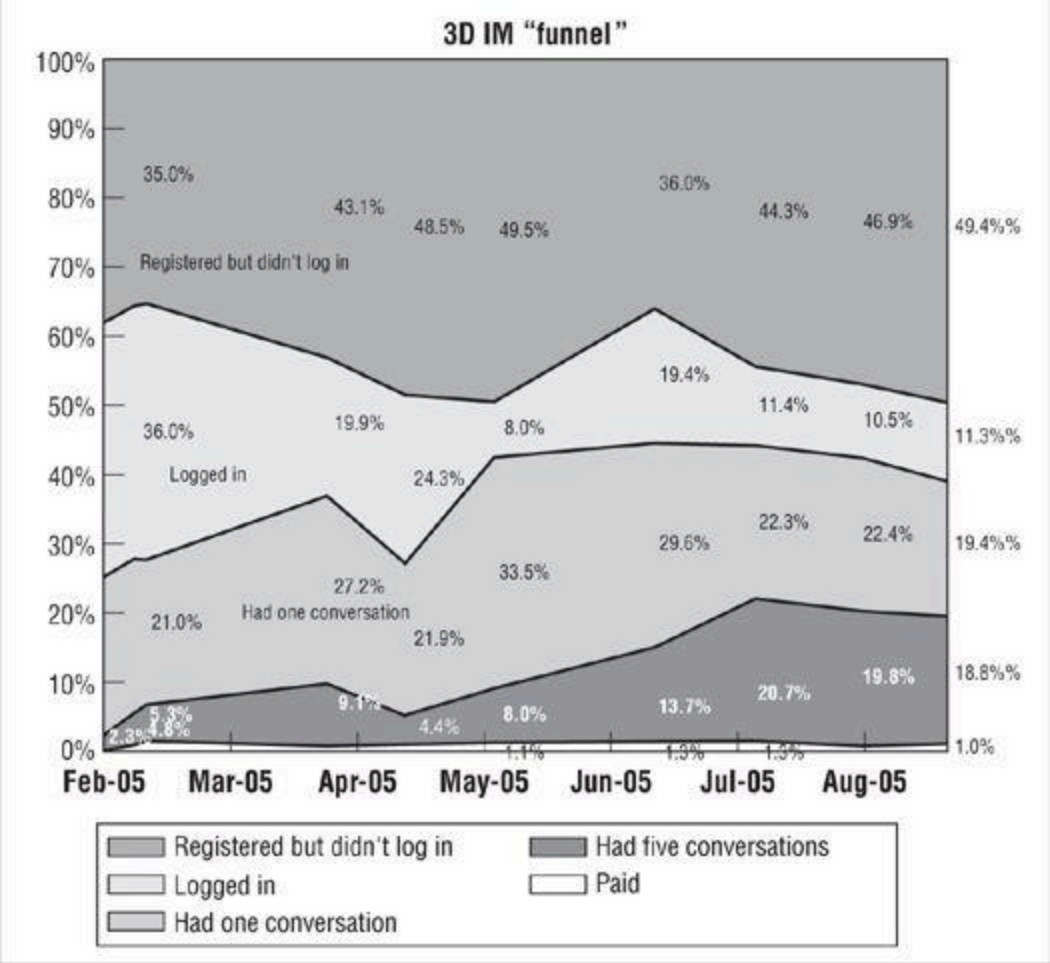
to open an account and get started even though we had very little money.[1]

Five dollars bought us a hundred clicks—every day. From a marketing point of view this was not very significant, but for learning it was priceless. Every single day we were able to measure our product's performance with a brand new set of customers. Also, each time we revised the product, we got a brand new report card on how we were doing the very next day.

For example, one day we would debut a new marketing message aimed at first-time customers. The next day we might change the way new customers were initiated into the product. Other days, we would add new features, fix bugs, roll out a new visual design, or try a new layout for our website. Every time, we told ourselves we were making the product better, but that subjective confidence was put to the acid test of real numbers.

Day in and day out we were performing random trials. Each day was a new experiment. Each day's customers were independent of those of the day before. Most important, even though our gross numbers were growing, it became clear that our funnel metrics were not changing.

Here is a graph from one of IMVU's early board meetings:



This graph represents approximately seven months of work. Over that period, we were making constant improvements to the IMVU product, releasing new features on a daily basis. We were conducting a lot of in-person customer interviews, and our product development team was working extremely hard.

Cohort Analysis

To read the graph, you need to understand something called *cohort analysis*. This is one of the most important tools of startup analytics. Although it sounds complex, it is based on a simple premise. Instead of looking at cumulative totals or gross numbers such as total revenue and total number of customers, one looks at the performance of each group of customers that comes into contact with the product independently.

Each group is called a cohort. The graph shows the conversion rates to IMVU of new customers who joined in each indicated month. Each conversion rate shows the percentage of customer who registered in that month who subsequently went on to take the indicated action. Thus, among all the customers who joined IMVU in February 2005, about 60 percent of them logged in to our product at least one time.

Managers with an enterprise sales background will recognize this funnel analysis as the traditional sales funnel that is used to manage prospects on their way to becoming customers. Lean Startups use it in product development, too. This technique is useful in many types of business, because every company depends for its survival on sequences of customer behavior called flows. Customer flows govern the interaction of customers with a company's products. They allow us to understand a business quantitatively and have much more predictive power than do traditional gross metrics.

If you look closely, you'll see that the graph shows some clear trends. Some product improvements are helping—a little. The percentage of new customers who go on to use the product at least five times has grown from less than 5 percent to almost 20 percent. Yet despite this fourfold increase, the percentage of new customers who pay money for IMVU is stuck at around 1 percent. Think about that for a moment. After months and months of work, thousands of individual improvements, focus groups, design sessions, and usability tests, the percentage of new customers who subsequently pay money is exactly the same as it was at the onset even though many more customers are getting a chance to try the product.

Thanks to the power of cohort analysis, we could not blame this failure on the legacy of previous customers who were resistant to change, external market conditions, or any other excuse. Each cohort represented an independent report card, and try as we might, we were getting straight C's. This helped us realize we had a problem.

I was in charge of the product development team, small though it was in those days, and shared with my cofounders the sense that the problem had to be with my team's efforts. I worked harder, tried to focus on higher- and higher-quality features, and lost a lot of sleep. Our frustration grew. When I could think of nothing else to do, I was finally ready to turn to the last resort: talking to customers. Armed with our failure to make progress tuning our engine of growth, I was ready to ask the right questions.

Before this failure, in the company's earliest days, it was easy to talk to potential customers and come away convinced we were on the right track. In fact, when we would invite customers into the office for in-person interviews and usability tests, it was easy to dismiss negative feedback. If they didn't want to use the product, I assumed they were not in our target market. "Fire that customer," I'd say to the person responsible for recruiting for our tests. "Find me someone in our target demographic." If the next customer was more positive, I would take it as confirmation that I was right in my targeting. If not, I'd fire another customer and try again.

By contrast, once I had data in hand, my interactions with customers changed. Suddenly I had urgent questions that needed answering: Why aren't customers responding to our product "improvements"? Why isn't our hard work paying off? For example, we kept making it easier and easier for customers to use IMVU with their existing friends. Unfortunately, customers didn't want to engage in that behavior. Making it easier to use was totally beside the point. Once we knew what to look for, genuine understanding came much faster. As was described in Chapter 3, this eventually led to a critically important pivot: away from an IM add-on used with existing friends and toward a stand-alone network one can use to make new friends. Suddenly, our worries about productivity vanished. Once our efforts were aligned with what customers really wanted, our experiments were much more likely to change their behavior for the better.

This pattern would repeat time and again, from the days when we were making less than a thousand dollars in revenue per month all the way up to the time we were making millions. In fact, this is the sign of a successful pivot: the new experiments you run are overall more productive than the experiments you were running before.

This is the pattern: poor quantitative results force us to declare failure and create the motivation, context, and space for more qualitative research. These investigations produce new ideas—new hypotheses—to be tested, leading to a possible pivot. Each pivot unlocks new opportunities for further experimentation, and the cycle repeats. Each time we repeat this simple rhythm: establish the baseline, tune the engine, and make a decision to pivot or persevere.

OPTIMIZATION VERSUS LEARNING

Engineers, designers, and marketers are all skilled at optimization. For example, direct marketers are experienced at split testing value propositions by sending a different offer to two similar groups of customers so that they can measure differences in the response rates of the two groups. Engineers, of course, are skilled at improving a product's performance, just as designers are talented at making products easier to use. All these activities in a well-run traditional organization offer incremental benefit for incremental effort. As long as we are executing the plan well, hard work yields results.

However, these tools for product improvement do not work the same way for startups. If you are building the wrong thing, optimizing the product or its marketing will not yield significant results. A startup has to measure progress against a high bar: evidence that a sustainable business can be built around its products or services. That's a standard that can be assessed only if a startup has made clear, tangible predictions ahead of time.

In the absence of those predictions, product and strategy decisions are far more difficult and time-consuming. I often see this in my consulting practice. I've been called in many times to help a startup that feels that its engineering team "isn't working hard enough." When I meet with those teams, there are always improvements to be made and I recommend them, but invariably the real problem is not a lack of development talent, energy, or effort. Cycle after cycle, the team is working hard, but the business is not seeing results. Managers trained in a traditional model draw the logical conclusion: our team is not working hard, not working effectively, or not working efficiently.

Thus the downward cycle begins: the product development team valiantly tries to build a product according to the specifications it is receiving from the creative or business leadership. When good results are not forthcoming, business leaders assume that any discrepancy between what was planned and what was built is the cause and try to specify the next iteration in greater detail. As the specifications get more detailed, the planning process slows down, batch size increases,

and feedback is delayed. If a board of directors or CFO is involved as a stakeholder, it doesn't take long for personnel changes to follow.

A few years ago, a team that sells products to large media companies invited me to help them as a consultant because they were concerned that their engineers were not working hard enough. However, the fault was not in the engineers; it was in the process the whole company was using to make decisions. They had customers but did not know them very well. They were deluged with feature requests from customers, the internal sales team, and the business leadership. Every new insight became an emergency that had to be addressed immediately. As a result, long-term projects were hampered by constant interruptions. Even worse, the team had no clear sense of whether any of the changes they were making mattered to customers. Despite the constant tuning and tweaking, the business results were consistently mediocre.

Learning milestones prevent this negative spiral by emphasizing a more likely possibility: the company is executing—with discipline!—a plan that does not make sense. The innovation accounting framework makes it clear when the company is stuck and needs to change direction.

In the example above, early in the company's life, the product development team was incredibly productive because the company's founders had identified a large unmet need in the target market. The initial product, while flawed, was popular with early adopters. Adding the major features that customers asked for seemed to work wonders, as the early adopters spread the word about the innovation far and wide. But unasked and unanswered were other lurking questions: Did the company have a working engine of growth? Was this early success related to the daily work of the product development team? In most cases, the answer was no; success was driven by decisions the team had made in the past. None of its current initiatives were having any impact. But this was obscured because the company's gross metrics were all "up and to the right."

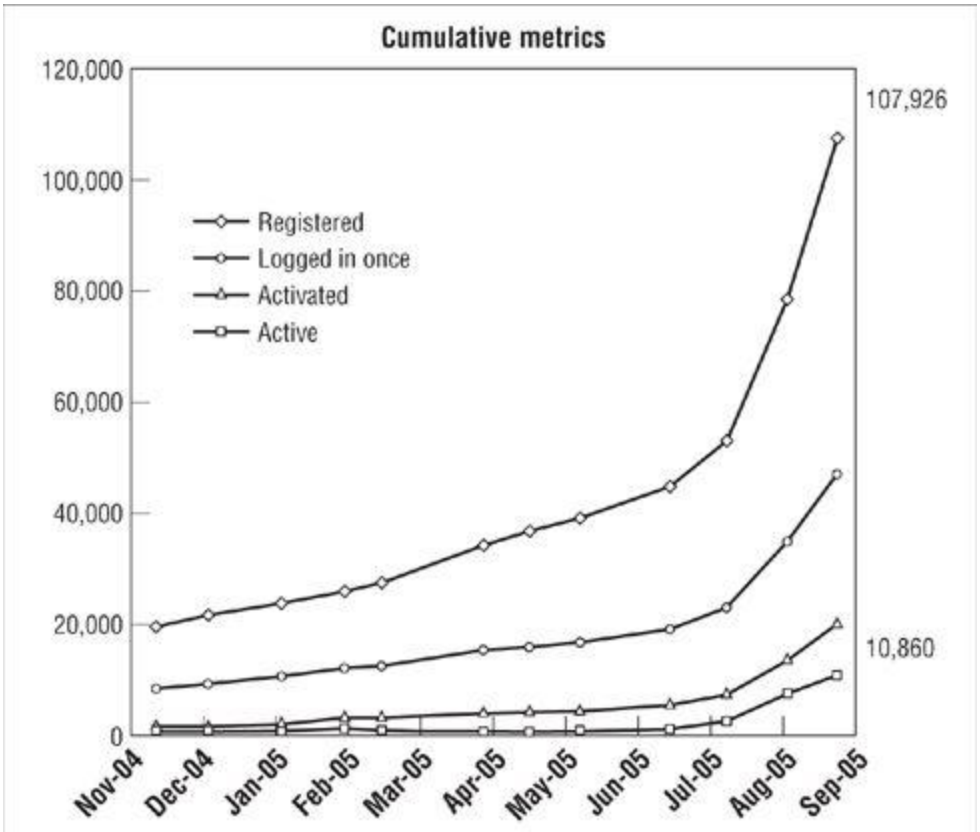
As we'll see in a moment, this is a common danger. Companies of any size that have a working engine of growth can come to rely on the wrong kind of metrics to guide their actions. This is what tempts managers to resort to the usual bag of success theater tricks: last-minute ad buys, channel stuffing, and whiz-bang demos, in a desperate attempt to make the gross numbers look better. Energy invested in

success theater is energy that could have been used to help build a sustainable business. I call the traditional numbers used to judge startups “vanity metrics,” and innovation accounting requires us to avoid the temptation to use them.

VANITY METRICS: A WORD OF CAUTION

To see the danger of vanity metrics clearly, let's return once more to the early days of IMVU. Take a look at the following graph, which is from the same era in IMVU's history as that shown earlier in this chapter. It covers the same time period as the cohort-style graph on this page; in fact, it is from the same board presentation.

This graph shows the traditional gross metrics for IMVU so far: total registered users and total paying customers (the gross revenue graph looks almost the same). From this viewpoint, things look much more exciting. That's why I call these vanity metrics: they give the rosiest possible picture. You'll see a traditional hockey stick graph (the ideal in a rapid-growth company). As long as you focus on the top-line numbers (signing up more customers, an increase in overall revenue), you'll be forgiven for thinking this product development team is making great progress. The company's growth engine is working. Each month it is able to acquire customers and has a positive return on investment. The excess revenue from those customers is reinvested the next month in acquiring more. That's where the growth is coming from.



But think back to the same data presented in a cohort style. IMVU is adding new customers, but it is not improving the yield on each new group. The engine is turning, but the efforts to tune the engine are not bearing much fruit. From the traditional graph alone, you cannot tell whether IMVU is on pace to build a sustainable business; you certainly can't tell anything about the efficacy of the entrepreneurial team behind it.

Innovation accounting will not work if a startup is being misled by these kinds of vanity metrics: gross number of customers and so on. The alternative is the kind of metrics we use to judge our business and our learning milestones, what I call *actionable metrics*.

ACTIONABLE METRICS VERSUS VANITY METRICS

To get a better sense of the importance of good metrics, let's look at a company called Grockit. Its founder, Farbood Nivi, spent a decade working as a teacher at two large for-profit education companies, Princeton Review and Kaplan, helping students prepare for standardized tests such as the GMAT, LSAT, and SAT. His engaging classroom style won accolades from his students and promotions from his superiors; he was honored with Princeton Review's National Teacher of the Year award. But Farb was frustrated with the traditional teaching methods used by those companies. Teaching six to nine hours per day to thousands of students, he had many opportunities to experiment with new approaches.[2]

Over time, Farb concluded that the traditional lecture model of education, with its one-to-many instructional approach, was inadequate for his students. He set out to develop a superior approach, using a combination of teacher-led lectures, individual homework, and group study. In particular, Farb was fascinated by how effective the student-to-student peer-driven learning method was for his students. When students could help each other, they benefited in two ways. First, they could get customized instruction from a peer who was much less intimidating than a teacher. Second, they could reinforce their learning by teaching it to others. Over time, Farb's classes became increasingly social—and successful.

As this unfolded, Farb felt more and more that his physical presence in the classroom was less important. He made an important connection: "I have this social learning model in my classroom. There's all this social stuff going on on the web." His idea was to bring social peer-to-peer learning to people who could not afford an expensive class from Kaplan or Princeton Review or an even more expensive private tutor. From this insight Grockit was born.

Farb explains, "Whether you're studying for the SAT or you're studying for algebra, you study in one of three ways. You spend some time with experts, you spend some time on your own, and you spend

some time with your peers. Grockit offers these three same formats of studying. What we do is we apply technology and algorithms to optimize those three forms."

Farb is the classic entrepreneurial visionary. He recounts his original insight this way: "Let's forget educational design up until now, let's forget what's possible and just redesign learning with today's students and today's technology in mind. There were plenty of multi-billion-dollar organizations in the education space, and I don't think they were innovating in the way that we needed them to and I didn't think we needed them anymore. To me, it's really all about the students and I didn't feel like the students were being served as well as they could."

Today Grockit offers many different educational products, but in the beginning Farb followed a lean approach. Grockit built a minimum viable product, which was simply Farb teaching test prep via the popular online web conferencing tool WebEx. He built no custom software, no new technology. He simply attempted to bring his new teaching approach to students via the Internet. News about a new kind of private tutoring spread quickly, and within a few months Farb was making a decent living teaching online, with monthly revenues of $10,000 to $15,000. But like many entrepreneurs with ambition, Farb didn't build his MVP just to make a living. He had a vision of a more collaborative, more effective kind of teaching for students everywhere. With his initial traction, he was able to raise money from some of the most prestigious investors in Silicon Valley.

When I first met Farb, his company was already on the fast track to success. They had raised venture capital from well-regarded investors, had built an awesome team, and were fresh off an impressive debut at one of Silicon Valley's famous startup competitions.

They were extremely process-oriented and disciplined. Their product development followed a rigorous version of the agile development methodology known as Extreme Programming (described below), thanks to their partnership with a San Francisco–based company called Pivotal Labs. Their early product was hailed by the press as a breakthrough.

There was only one problem: they were not seeing sufficient growth in the use of the product by customers. Grockit is an excellent case study because its problems were not a matter of failure of execution or discipline.

Following standard agile practice, Grockit's work proceeded in a series of *sprints*, or one-month iteration cycles. For each sprint, Farb would prioritize the work to be done that month by writing a series of *user stories*, a technique taken from agile development. Instead of writing a specification for a new feature that described it in technical terms, Farb would write a story that described the feature from the point of view of the customer. That story helped keep the engineers focused on the customer's perspective throughout the development process.

Each feature was expressed in plain language in terms everyone could understand whether they had a technical background or not. Again following standard agile practice, Farb was free to reprioritize these stories at any time. As he learned more about what customers wanted, he could move things around in the *product backlog*, the queue of stories yet to be built. The only limit on this ability to change directions was that he could not interrupt any task that was in progress. Fortunately, the stories were written in such a way that the batch size of work (which I'll discuss in more detail in Chapter 9) was only a day or two.

This system is called agile development for a good reason: teams that employ it are able to change direction quickly, stay light on their feet, and be highly responsive to changes in the business requirements of the product owner (the manager of the process—in this case Farb—who is responsible for prioritizing the stories).

How did the team feel at the end of each sprint? They consistently delivered new product features. They would collect feedback from customers in the form of anecdotes and interviews that indicated that at least some customers liked the new features. There was always a certain amount of data that showed improvement: perhaps the total number of customers was increasing, the total number of questions answered by students was going up, or the number of returning customers was increasing.

However, I sensed that Farb and his team were left with lingering doubts about the company's overall progress. Was the increase in their numbers actually caused by their development efforts? Or could it be due to other factors, such as mentions of Grockit in the press? When I met the team, I asked them this simple question: How do you know that the prioritization decisions that Farb is making actually make sense?

Their answer: “That’s not our department. Farb makes the decisions; we execute them.”

At that time Grockit was focused on just one customer segment: prospective business school students who were studying for the GMAT. The product allowed students to engage in online study sessions with fellow students who were studying for the same exam. The product was working: the students who completed their studying via Grockit achieved significantly higher scores than they had before. But the Grockit team was struggling with the age-old startup problems: How do we know which features to prioritize? How can we get more customers to sign up and pay? How can we get out the word about our product?

I put this question to Farb: “How confident are you that you are making the right decisions in terms of establishing priorities?” Like most startup founders, he was looking at the available data and making the best educated guesses he could. But this left a lot of room for ambiguity and doubt.

Farb believed in his vision thoroughly and completely, yet he was starting to question whether his company was on pace to realize that vision. The product improved every day, but Farb wanted to make sure those improvements mattered to customers. I believe he deserves a lot of credit for realizing this. Unlike many visionaries, who cling to their original vision no matter what, Farb was willing to put his vision to the test.

Farb worked hard to sustain his team’s belief that Grockit was destined for success. He was worried that morale would suffer if anyone thought that the person steering the ship was uncertain about which direction to go. Farb himself wasn’t sure if his team would embrace a true learning culture. After all, this was part of the grand bargain of agile development: engineers agree to adapt the product to the business’s constantly changing requirements but are not responsible for the quality of those business decisions.

Agile is an efficient system of development from the point of view of the developers. It allows them to stay focused on creating features and technical designs. An attempt to introduce the need to learn into that process could undermine productivity.

(Lean manufacturing faced similar problems when it was introduced in factories. Managers were used to focusing on the utilization rate of each machine. Factories were designed to keep machines running at

full capacity as much of the time as possible. Viewed from the perspective of the machine, that is efficient, but from the point of view of the productivity of the entire factory, it is wildly inefficient at times. As they say in systems theory, that which optimizes one part of the system necessarily undermines the system as a whole.)

What Farb and his team didn't realize was that Grockit's progress was being measured by vanity metrics: the total number of customers and the total number of questions answered. That was what was causing his team to spin its wheels; those metrics gave the team the sensation of forward motion even though the company was making little progress. What's interesting is how closely Farb's method followed superficial aspects of the Lean Startup learning milestones: they shipped an early product and established some baseline metrics. They had relatively short iterations, each of which was judged by its ability to improve customer metrics.

However, because Grockit was using the wrong kinds of metrics, the startup was not genuinely improving. Farb was frustrated in his efforts to learn from customer feedback. In every cycle, the type of metrics his team was focused on would change: one month they would look at gross usage numbers, another month registration numbers, and so on. Those metrics would go up and down seemingly on their own. He couldn't draw clear cause-and-effect inferences. Prioritizing work correctly in such an environment is extremely challenging.

Farb could have asked his data analyst to investigate a particular question. For example, when we shipped feature X, did it affect customer behavior? But that would have required tremendous time and effort. When, exactly, did feature X ship? Which customers were exposed to it? Was anything else launched around that same time? Were there seasonal factors that might be skewing the data? Finding these answers would have required parsing reams and reams of data. The answer often would come weeks after the question had been asked. In the meantime, the team would have moved on to new priorities and new questions that needed urgent attention.

Compared to a lot of startups, the Grockit team had a huge advantage: they were tremendously disciplined. A disciplined team may apply the wrong methodology but can shift gears quickly once it discovers its error. Most important, a disciplined team can experiment with its own working style and draw meaningful conclusions.

Cohorts and Split-tests

Grockit changed the metrics they used to evaluate success in two ways. Instead of looking at gross metrics, Grockit switched to cohort-based metrics, and instead of looking for cause-and-effect relationships after the fact, Grockit would launch each new feature as a true split-test experiment.

A split-test experiment is one in which different versions of a product are offered to customers at the same time. By observing the changes in behavior between the two groups, one can make inferences about the impact of the different variations. This technique was pioneered by direct mail advertisers. For example, consider a company that sends customers a catalog of products to buy, such as Lands' End or Crate & Barrel. If you wanted to test a catalog design, you could send a new version of it to 50 percent of the customers and send the old standard catalog to the other 50 percent. To assure a scientific result, both catalogs would contain identical products; the only difference would be the changes to the design. To figure out if the new design was effective, all you would have to do was keep track of the sales figures for both groups of customers. (This technique is sometimes called A/B testing after the practice of assigning letter names to each variation.) Although split testing often is thought of as a marketing-specific (or even a direct marketing–specific) practice, Lean Startups incorporate it directly into product development.

These changes led to an immediate change in Farb's understanding of the business. Split testing often uncovers surprising things. For example, many features that make the product better in the eyes of engineers and designers have no impact on customer behavior. This was the case at Grockit, as it has been in every company I have seen adopt this technique. Although working with split tests seems to be more difficult because it requires extra accounting and metrics to keep track of each variation, it almost always saves tremendous amounts of time in the long run by eliminating work that doesn't matter to customers.

Split testing also helps teams refine their understanding of what customers want and don't want. Grockit's team constantly added new ways for their customers to interact with each other in the hope that those social communication tools would increase the product's value.

Inherent in those efforts was the belief that customers desired more communication during their studying. When split testing revealed that the extra features did not change customer behavior, it called that belief into question.

The questioning inspired the team to seek a deeper understanding of what customers really wanted. They brainstormed new ideas for product experiments that might have more impact. In fact, many of these ideas were not new. They had simply been overlooked because the company was focused on building social tools. As a result, Grockit tested an intensive solo-studying mode, complete with quests and gamelike levels, so that students could have the choice of studying by themselves or with others. Just as in Farb's original classroom, this proved extremely effective. Without the discipline of split testing, the company might not have had this realization. In fact, over time, through dozens of tests, it became clear that the key to student engagement was to offer them a combination of social and solo features. Students preferred having a choice of how to study.

Kanban

Following the lean manufacturing principle of *kanban*, or capacity constraint, Grockit changed the product prioritization process. Under the new system, user stories were not considered complete until they led to validated learning. Thus, stories could be cataloged as being in one of four states of development: in the product backlog, actively being built, done (feature complete from a technical point of view), or in the process of being validated. Validated was defined as "knowing whether the story was a good idea to have been done in the first place." This validation usually would come in the form of a split test showing a change in customer behavior but also might include customer interviews or surveys.

The *kanban* rule permitted only so many stories in each of the four states. As stories flow from one state to the other, the buckets fill up. Once a bucket becomes full, it cannot accept more stories. Only when a story has been validated can it be removed from the *kanban* board. If the validation fails and it turns out the story is a bad idea, the relevant feature is removed from the product (see the chart on this page).

KANBAN DIAGRAM OF WORK AS IT PROGRESSES FROM STAGE TO STAGE

(No bucket can contain more than three projects at a time.)

BACKLOG	IN PROGRESS	BUILT	VALIDATED
A B C	D E	F	

Work on A begins. D and E are in development. F awaits validation.

BACKLOG	IN PROGRESS	BUILT	VALIDATED
G H I	 B C	D E A	F

F is validated. D and E await validation. G, H, I are new tasks to be undertaken. B and C are being built. A completes development.

BACKLOG	IN PROGRESS	BUILT	VALIDATED
 H → I →	G B → C →	D E A	F

B and C have been built, but under *kanban*, cannot be moved to the next bucket for validation until A, D, E have been validated. Work cannot begin on H and I until space opens up in the buckets ahead.

I have implemented this system with several teams, and the initial result is always frustrating: each bucket fills up, starting with the "validated" bucket and moving on to the "done" bucket, until it's not possible to start any more work. Teams that are used to measuring their productivity narrowly, by the number of stories they are delivering, feel stuck. The only way to start work on new features is to investigate some of the stories that are done but haven't been validated. That often requires nonengineering efforts: talking to customers, looking at split-test data, and the like.

Pretty soon everyone gets the hang of it. This progress occurs in fits and starts at first. Engineering may finish a big batch of work, followed by extensive testing and validation. As engineers look for ways to increase their productivity, they start to realize that if they include the validation exercise from the beginning, the whole team can be more productive.

For example, why build a new feature that is not part of a split-test experiment? It may save you time in the short run, but it will take more time later to test, during the validation phase. The same logic applies to a story that an engineer doesn't understand. Under the old system, he or she would just build it and find out later what it was for. In the new system, that behavior is clearly counterproductive: without a clear hypothesis, how can a story ever be validated? We saw this behavior at IMVU, too. I once saw a junior engineer face down a senior executive over a relatively minor change. The engineer insisted that the new feature be split-tested, just like any other. His peers backed him up; it was considered absolutely obvious that all features should be routinely tested, no matter who was commissioning them. (Embarrassingly, all too often I was the executive in question.) A solid process lays the foundation for a healthy culture, one where ideas are evaluated by merit and not by job title.

Most important, teams working in this system begin to measure their productivity according to validated learning, not in terms of the production of new features.

Hypothesis Testing at Grockit

When Grockit made this transition, the results were dramatic. In one case, they decided to test one of their major features, called lazy registration, to see if it was worth the heavy investment they were making in ongoing support. They were confident in this feature because lazy registration is considered one of the design best practices for online services. In this system, customers do not have to register for the service up front. Instead, they immediately begin using the service and are asked to register only after they have had a chance to experience the service's benefit.

For a student, lazy registration works like this: when you come to the Grockit website, you're immediately placed in a study session with other students working on the same test. You don't have to give your name, e-mail address, or credit card number. There is nothing to prevent you from jumping in and getting started immediately. For Grockit, this was essential to testing one of its core assumptions: that customers would be willing to adopt this new way of learning only if they could see proof that it was working early on.

As a result of this hypothesis, Grockit's design required that it manage three classes of users: unregistered guests, registered (trial) guests, and customers who had paid for the premium version of the product. This design required significant extra work to build and maintain: the more classes of users there are, the more work is required to keep track of them, and the more marketing effort is required to create the right incentives to entice customers to upgrade to the next class. Grockit had undertaken this extra effort because lazy registration was considered an industry best practice.

I encouraged the team to try a simple split-test. They took one cohort of customers and required that they register immediately, based on nothing more than Grockit's marketing materials. To their surprise, this cohort's behavior was exactly the same as that of the lazy registration group: they had the same rate of registration, activation, and subsequent retention. In other words, the extra effort of lazy registration was a complete waste even though it was considered an industry best practice.

Even more important than reducing waste was the insight that this test suggested: customers were basing their decision about Grockit on

something other than their use of the product.

Think about this. Think about the cohort of customers who were required to register for the product before entering a study session with other students. They had very little information about the product, nothing more than was presented on Grockit's home page and registration page. By contrast, the lazy registration group had a tremendous amount of information about the product because they had used it. Yet despite this information disparity, customer behavior was exactly the same.

This suggested that improving Grockit's positioning and marketing might have a more significant impact on attracting new customers than would adding new features. This was just the first of many important experiments Grockit was able to run. Since those early days, they have expanded their customer base dramatically: they now offer test prep for numerous standardized tests, including the GMAT, SAT, ACT, and GRE, as well as online math and English courses for students in grades 7 through 12.

Grockit continues to evolve its process, seeking continuous improvement at every turn. With more than twenty employees in its San Francisco office, Grockit continues to operate with the same deliberate, disciplined approach that has been their hallmark all along. They have helped close to a million students and are sure to help millions more.

THE VALUE OF THE THREE A'S

These examples from Grockit demonstrate each of the three A's of metrics: actionable, accessible, and auditable.

Actionable

For a report to be considered actionable, it must demonstrate clear cause and effect. Otherwise, it is a vanity metric. The reports that Grockit's team began to use to judge their learning milestones made it extremely clear what actions would be necessary to replicate the results.

By contrast, vanity metrics fail this criterion. Take the number of hits to a company website. Let's say we have 40,000 hits this month—a new record. What do we need to do to get more hits? Well, that depends. Where are the new hits coming from? Is it from 40,000 new customers or from one guy with an extremely active web browser? Are the hits the result of a new marketing campaign or PR push? What is a hit, anyway? Does each page in the browser count as one hit, or do all the embedded images and multimedia content count as well? Those who have sat in a meeting debating the units of measurement in a report will recognize this problem.

Vanity metrics wreak havoc because they prey on a weakness of the human mind. In my experience, when the numbers go up, people think the improvement was caused by their actions, by whatever they were working on at the time. That is why it's so common to have a meeting in which marketing thinks the numbers went up because of a new PR or marketing effort and engineering thinks the better numbers are the result of the new features it added. Finding out what is actually going on is extremely costly, and so most managers simply move on, doing the best they can to form their own judgment on the basis of their experience and the collective intelligence in the room.

Unfortunately, when the numbers go down, it results in a very different reaction: now it's somebody else's fault. Thus, most team members or departments live in a world where their department is constantly making things better, only to have their hard work sabotaged by other departments that just don't get it. Is it any wonder these departments develop their own distinct language, jargon, culture, and defense mechanisms against the bozos working down the hall?

Actionable metrics are the antidote to this problem. When cause and effect is clearly understood, people are better able to learn from their actions. Human beings are innately talented learners when given a clear and objective assessment.

Accessible

All too many reports are not understood by the employees and managers who are supposed to use them to guide their decision making. Unfortunately, most managers do not respond to this complexity by working hand in hand with the data warehousing team to simplify the reports so that they can understand them better. Departments too often spend their energy learning how to use data to get what they want rather than as genuine feedback to guide their future actions.

There is an antidote to this misuse of data. First, make the reports as simple as possible so that everyone understands them. Remember the saying "Metrics are people, too." The easiest way to make reports comprehensible is to use tangible, concrete units. What is a website hit? Nobody is really sure, but everyone knows what a person visiting the website is: one can practically picture those people sitting at their computers.

This is why cohort-based reports are the gold standard of learning metrics: they turn complex actions into people-based reports. Each cohort analysis says: among the people who used our product in this period, here's how many of them exhibited each of the behaviors we care about. In the IMVU example, we saw four behaviors: downloading the product, logging into the product from one's computer, engaging in a chat with other customers, and upgrading to the paid version of the product. In other words, the report deals with people and their actions,

which are far more useful than piles of data points. For example, think about how hard it would have been to tell if IMVU was being successful if we had reported only on the total number of person-to-person conversations. Let's say we have 10,000 conversations in a period. Is that good? Is that one person being very, very social, or is it 10,000 people each trying the product one time and then giving up? There's no way to know without creating a more detailed report.

As the gross numbers get larger, accessibility becomes more and more important. It is hard to visualize what it means if the number of website hits goes down from 250,000 in one month to 200,000 the next month, but most people understand immediately what it means to lose 50,000 customers. That's practically a whole stadium full of people who are abandoning the product.

Accessibility also refers to widespread access to the reports. Grockit did this especially well. Every day their system automatically generated a document containing the latest data for every single one of their split-test experiments and other leap-of-faith metrics. This document was mailed to every employee of the company: they all always had a fresh copy in their e-mail in-boxes. The reports were well laid out and easy to read, with each experiment and its results explained in plain English.

Another way to make reports accessible is to use a technique we developed at IMVU. Instead of housing the analytics or data in a separate system, our reporting data and its infrastructure were considered part of the product itself and were owned by the product development team. The reports were available on our website, accessible to anyone with an employee account.

Each employee could log in to the system at any time, choose from a list of all current and past experiments, and see a simple one-page summary of the results. Over time, those one-page summaries became the de facto standard for settling product arguments throughout the organization. When people needed evidence to support something they had learned, they would bring a printout with them to the relevant meeting, confident that everyone they showed it to would understand its meaning.

Auditable

When informed that their pet project is a failure, most of us are tempted to blame the messenger, the data, the manager, the gods, or anything else we can think of. That's why the third A of good metrics, "auditable," is so essential. We must ensure that the data is credible to employees.

The employees at IMVU would brandish one-page reports to demonstrate what they had learned to settle arguments, but the process often wasn't so smooth. Most of the time, when a manager, developer, or team was confronted with results that would kill a pet project, the loser of the argument would challenge the veracity of the data.

Such challenges are more common than most managers would admit, and unfortunately, most data reporting systems are not designed to answer them successfully. Sometimes this is the result of a well-intentioned but misplaced desire to protect the privacy of customers. More often, the lack of such supporting documentation is simply a matter of neglect. Most data reporting systems are not built by product development teams, whose job is to prioritize and build product features. They are built by business managers and analysts. Managers who must use these systems can only check to see if the reports are mutually consistent. They all too often lack a way to test if the data is consistent with reality.

The solution? First, remember that "Metrics are people, too." We need to be able to test the data by hand, in the messy real world, by talking to customers. This is the only way to be able to check if the reports contain true facts. Managers need the ability to spot check the data with real customers. It also has a second benefit: systems that provide this level of auditability give managers and entrepreneurs the opportunity to gain insights into why customers are behaving the way the data indicate.

Second, those building reports must make sure the mechanisms that generate the reports are not too complex. Whenever possible, reports should be drawn directly from the master data, rather than from an intermediate system, which reduces opportunities for error. I have noticed that every time a team has one of its judgments or assumptions overturned as a result of a technical problem with the data, its confidence, morale, and discipline are undermined.

When we watch entrepreneurs succeed in the mythmaking world of Hollywood, books, and magazines, the story is always structured the same way. First, we see the plucky protagonist having an epiphany, hatching a great new idea. We learn about his or her character and personality, how he or she came to be in the right place at the right time, and how he or she took the dramatic leap to start a business.

Then the photo montage begins. It's usually short, just a few minutes of time-lapse photography or narrative. We see the protagonist building a team, maybe working in a lab, writing on whiteboards, closing sales, pounding on a few keyboards. At the end of the montage, the founders are successful, and the story can move on to more interesting fare: how to split the spoils of their success, who will appear on magazine covers, who sues whom, and implications for the future.

Unfortunately, the real work that determines the success of startups happens during the photo montage. It doesn't make the cut in terms of the big story because it is too boring. Only 5 percent of entrepreneurship is the big idea, the business model, the whiteboard strategizing, and the splitting up of the spoils. The other 95 percent is the gritty work that is measured by innovation accounting: product prioritization decisions, deciding which customers to target or listen to, and having the courage to subject a grand vision to constant testing and feedback.

One decision stands out above all others as the most difficult, the most time-consuming, and the biggest source of waste for most startups. We all must face this fundamental test: deciding when to pivot and when to persevere. To understand what happens during the photo montage, we have to understand how to pivot, and that is the subject of Chapter 8.