Todd Zenger, MBC Online, Testing Your Theory of Value

This video segment assumes that you have already built a theory of value, and we want to talk in this video segment about how you use that theory then to test its validity. Test whether this is a useful theory, whether this theory is indeed going to lead you to build an effective business model that will introduce new value into the world. Part of the exercise here is really about building confidence, reducing uncertainty, such that you not only convince yourself that this is a good theory of value but you'll be able to, with both the logic of that theory and the persuasion of evidence that you collect from trying to test that theory, that you're also able to convince investors and other stakeholders that may be essential to building out your business model that they'll be willing to jump on board and provide the resources that you need. We're going to continue to extend and build on this value lab tool that we've introduced in past lectures and really shift to the right-hand side of this figure and talk about how we use it to run experiments and shop for solutions and explore different types of solutions to critical subproblems that your theory has identified. What your theory of value provides is, of course, a vehicle for composing testable hypotheses. This is what I think should be true. In addition to that, it provides a vehicle for engaging in relatively costless thought experiments. What if we engaged in the following kinds of things? How will this impact my ability to change the world to introduce new value? It also provides a vehicle for specifying experiments that you can run and data that you can collect, and most importantly, an ability to interpret that data that you collect. This data tells me that that assumption is correct or that I can solve this particular subproblem that I have identified as needful in introducing this value. In that sense, it's a vehicle to structure the search for solutions to subproblems and again, a logical architecture for then interpreting the results of search efforts and your experimental efforts, and your data collection efforts. Let me talk for a minute about what that means in practical reality. To do that, go back to a theory or really apply this concept to a particular corporation infer what their theory might have been and talk about how experimentation can help provide validation that this is a good theory, one that we think will result in a business model that can be effective. If you think about Uber, the firm that we're all familiar with, the triggering problem that led to this business was this problem of how do you find fast, reliable taxi service, especially at times when taxis are difficult to secure. The story is that they were waiting for a taxi on a rainy day and couldn't get a single vehicle to come and pick them up. This was the framing problem and their belief, this contrarian belief that they had was, there is a vast reservoir of personal vehicles and the drivers attached to those vehicles that would be willing to be called into service to provide taxi service for me and my acute need to satisfy this demand. However, to bring that belief to fruition, there is a core problem. That is how do you compose a system that efficiently can match these untapped drivers and their vehicles to customers and of course, in the process, make both sides feel confident about being in this same vehicle and comfortable and safe. Then there are bunch of subproblems that then link to that core problem. If we're going to solve that core problem of an efficient matching system and a mechanism that enables people to feel confident and comfortable in that setting in a vehicle together, you need to be able to solve a problem of how do you facilitate exchange of funds in an efficient way. You have to deal with the problem of arrivals uncertainty, and how do you manage the peak load problem? How do you call more drivers into service when they're more needed? How do you create that reliability in this system? That is a problem in the pre-existing taxi business. With this problem framed in the architecture of this theory, which is if we can solve these subproblems, then we solve our core problem, which will allow us to bring this belief into reality. Now they're queued up as to what they need to engage in, in the way of experiments and search out what kinds of technology can we discover and find that might help us support this efficient payment

management. How do we make people feel, and how do we create a system that gives people confidence in driving with strangers? They've laid out the architecture that will then lead to a set of experiments, and probably in this case, more importantly, solution search to the sub-problems. Can we find technology that can help us solve these subproblems? Let me give you one other example, perhaps even more familiar early on when personal computers were introduced, both Apple and IBM and then a bunch of followers, personal computers were clunky and unreliable and really difficult to use unless you were a geek or you had tech support constantly providing you with the support that you needed to keep your system up and running. That meant that for the most part, these were devices that would be sold in a business setting where there would be the tech support that was needed to help you install new software and peripherals and navigate this clunky machine. Of course, Steve Jobs's belief was that if he could make these personal computers easy to use and reliable that the masses would flock to these, then they would become a consumer product, also sell them at premium prices. The core problem was then of course, how do we make personal computers that are easy to use and reliable and the architecture of this theory was, we need to make in particular this operating system less clumsy and make it much more intuitive. We need to make integrating peripherals and new software onto this machine much more streamlined and seamless. It's got to work the first time. There was also a problem at the time, which was that there was a complete disconnect between what you would see on your computer if you were in a word processing program, for instance, and what you would actually see on the paper when it was printed, which seems like a little bit of a trivial problem today, but at the time it was an extraordinarily important problem that was associated with these devices. Then they were just unreliable and unstable, constantly shutting down as you switch to a new software or install some new device. How do you make these systems much more reliable? These were then the subproblems that were the architecture of his theory. This then set a motion and informed a bunch of experiments search for solutions that could be run in order to bring this belief forward and make it a reality. You have a theory of value that you're trying to test, you're obviously not trying to test Apple's or Uber's theory of value. How do you go about selecting a set of experiments? How do you go about finding a path to explore solutions to these sub-problems that you have identified, which ones do you take up and explore first. How do you go about efficiently testing a theory? You have composed your own theory and your interest is not in testing Uber's or Apple's but your own theory, so what's the approach you should adopt to test that theory? The general principle I want to push is that you should experiment first and search first where you can learn the most with the lowest cost investment. It's maximize learning, minimize cost. What does that mean? It means that you would want to experiment first to test what might be the weakest assumption that you have in your theory, the one that you're not quite sure is accurate, the ones that you have the greatest doubt about, those that if false would negate the entire theory. You might also want to search first for solutions to sub-problems that are most critical. Those sub-problems which if we can't figure out how to solve this one it's really going to negate the entire theory. Again, all of this is while conserving on resources, so whether you experiment first with the cheapest experiments or you experiment first with those experiments that are going to deliver the most reduction in uncertainty. You have to balance that cost versus what's going to be most useful and informative to you. But it ought to be that balance that guides this selection of experiments that you pursue. I want to just talk for a minute about how this relates to a concept you've probably heard a lot about and we referenced in prior lectures this notion of minimum viable product. One of the big pushes in the entrepreneurship space and incubators is what you ought to do is get really quickly to a minimum viable product, put that out to customers so that they can react to that and that's well and good as long

as the type of theory you have actually enables you to compose a minimum viable product relatively cheaply. If you were to tell Apple to put out a minimum viable product that would be the Macintosh that really wasn't a possibility. The amount of investment required to pull all that together into a minimum viable product would have been exorbitant. Similarly, that would have been true with Uber as well. To put together that entire composition of systems that would enable drivers to be called into service and customers to feel confident stepping into those vehicles and to have an efficient approach to payment and have ratings and all that they ultimately compose in their business model to put that together in an MVP would have been extraordinarily costly. What do you do instead? Well, your other option is to engage in what we'll cause a theory-based learning and search and experimentation. As we have, composer theory was sub-problems and assumptions and then test individual assumptions and say, hey, you know what? That assumption is wrong, that sub-problem, we need to find a different way to solve it because this approach isn't working. You're able to get feedback and learning about particular sub-problems and assumptions and use that theory to guide your path of experimentation. As opposed to saying, we're going to put together the whole business model, throw it out, and see how the customer reacts. That'll work if you've got a relatively simple product and you can easily throw out a minimum viable product. The problem of giving you all advice of put out the minimum viable product. For some of you it will work because doing that is relatively cheap and efficient. For others of you, it's a much more complex theory that you have and what you need to think about is providing maybe minimum viable solutions to particular sub-problems and getting feedback and information in response to those sub-problems. Let's go back to our Uber example. What would this approach to experimentation lead Uber to do in order to test its theory at the outset? You certainly want to know if customers will ride with strangers in private cars and you would ask, so what experiment could we run that would give us that feedback? Are there other situations that are analogous that would provide us some information about the feasibility of that? That's going to be the most difficult I think experiment or was I'm sure the most difficult experiment for them to run and the most difficult feedback for them to get because you almost have to have everything in place to get that. Can we compose an efficient matching app? Well, there were lots of analogous apps, dating apps, and marketplace applications that had already been developed that I assume provided them with confidence. We could modify this, change this, and develop a solution to that particular sub-problem. Can we efficiently confidently arrange for safe payment among strangers? Well, that had also been worked out across different Internet platforms and probably gave them confidence that there was a plugin solution to that sub-problem that they could solve. Can we ensure and motivate quality, give people confidence that stepping in a vehicle with this particular driver was a safe thing and would be a pleasant experience while these rating systems would provide that confidence? Similarly, drivers is this a customer that we should allow into our vehicle. Again, this two-sided rating system. There were other businesses that had evolved that had these already in place and provided Uber presumably with confidence that they could do this. Knowing in particular these last three things they could probably solve gave them some confidence that maybe we can get people to step into the cars with strangers because we can solve these other sub-problems. As you think about your own theory of value, the sub-problems that you have to solve, think about experiments that you can run, places that you can look to find solutions to particular sub-problems that you highlighted. Let me just say a little bit about why this is particularly a useful approach. A scientific study was done actually by some colleagues and co-authors of mine, at least one of them was co-author. This was a study that was done actually in Milan. It has been repeated in a couple of other locations across the globe. The nature of the experiment was to divide a bunch of entrepreneurs into two

separate groups, a control group and a treated group and the treated group would essentially get what amounted to the value lab intervention, a real push to develop a theory, a structured set of beliefs, testable hypotheses from that theory. Whereas the other group was given standard startup training tools and methods. Both of them actually got the standard startup tools and methods but only one of them got this structured theory composition approach. Then they were set off and said go out and test your business model and see what information you glean and develop. Those that were trained to start with a theory were relative to that control group much more likely to develop clear theories and hypotheses not surprisingly but more importantly, to develop targeted experiments that would test those theories. They were as a consequence of this more likely to pivot. That is, to change their theory, to change their proposed business model. Why? Because the information that they gleaned from the searching they did and the experiments that they ran provided much more useful information that they could interpret, and say, Oh my gosh, this theory is not going to work because we can't solve that particular sub-problem or this hypothesis, here is incorrect, we need to modify it. They were therefore much more quick to pivot because the information they received as feedback was much more interpretable. They were also similar in an analogous way, much more likely to exit, just shut this thing down. This is not going to work as opposed to spending lots more time and resources pursuing a business idea where the evidence coming back filtered through this theory was telling them it wasn't going to work. They were much more likely to exit and save on resources. Then ultimately the thing that you probably care about most because you're all convinced your theories are correct and you very well should, they were also more likely to generate using the measures they had available at the time more successful businesses, more successful business models. Using this theory-based learning, running experiments filtered through theory led to quicker pivots, faster exits, and more successful businesses overall. What you're trying to do with this theory-based learning which I'm pushing you to do is to really in a detailed way compose that theory. You've done that in prior lectures and exercises, then run these experiments, and run the search efforts to solve your sub-problems, generate feedback, and use that feedback to update your theory. It could be to abandon your theory or it could be simply to give you confidence in your theory moving forward. In this sense we've gone through this last phase of the value lab, we're using that theory to help us think about what experiments to run, where to shop for investments, where can we find access to the resources that we need to build out this theory? Where do we need to search for solutions to sub-problems? I encourage you as an exercise to really think about what are the hypotheses that we need to test? What are the assumptions we need to test? What are those sub-problems? What are the critical resources that we need to secure in order to bring about this theory, to bring about this belief? Just a preview of where we're going from here as you develop this theory and you're starting to run experiments and explore critical sub-problems. Ultimately where we want to get to is building out a business model that's consistent with that theory updated by the experiments that you are running. This is where we're going to go next is to think about how do we build a business model based on the theory that you have composed. As we close out this lecture let me just push you to think about a couple of critical questions. Again, you've composed your theory at this point and now it's time to test it and to think about how to test it. What is the critical information that I would like to possess that would help me feel more confident about this theory? What are the critical assumptions that I'm making? What do I wish I knew? What information might cause me to pivot to a different business model or to a different theory? What are the critical hypotheses or assumptions that I really think I need to step out and test before I move any further with this particular theory?