aws

Humility

Fueling innovation through servant
leadership in an age of uncertainty

Mark Schwartz, Enterprise Strategist
at Amazon Web Services

You might say that humility is the essence of digital transformation.

In the digital world, we are willing to be surprised and to learn. In the old days, we relied on a plan—prepared in advance—to guide our activities. The plan was made by someone, or some collection of someones, who knew enough to specify what needed to be done over a longish period.

But in a world dominated by uncertainty, in a world in which none of us knows all there is to know about our customers, about the users of our IT systems, or about what will change in our competitive or regulatory environment, how can someone presume to put together a plan that will surely deliver business value? To do so seems more and more an act of hubris. Have we never been surprised by the market reaction to our products or by the way users used our software?

Important lessons from startups

In *The Lean Startup*, Eric Ries describes product development as a learning journey where the company tries out a business idea with a minimal viable product (MVP), gathers feedback from the market, changes its design based on that feedback, and repeats this process over and over. The goal is to maximize learning early in the process: to get something into the hands of customers and be prepared to be surprised by their reactions. At any given moment, Ries says, the company should have two hypotheses in mind: a *value hypothesis* about what product and product attributes will create value for the customer, and a *growth hypothesis* about how the product will be able to ramp up its sales or usage. Each of these hypotheses is just that—a hypothesis, something that must be confirmed or disproved through experimental evidence.

This is a process founded in humility. The startup—even if it is filled with experts on the target market or the industry—admits that its ideas are only conjectures and that in order to serve a market, it must learn from that market. Periodically, the team decides whether to persevere with its current strategy or to pivot to another direction. The arrogance of old-school product design—the idea that a marketing or product expert knows enough to define a product that can then be "marketed" or pushed out to customers, seems increasingly out of touch with business reality.

> "
>
> Even for IT applications intended for use within the company, an attitude of humility is necessary."

IT needs humility too

Ries's principles do not just apply to startups or to products created for an external market. Even for IT applications intended for use within the company, an attitude of humility is necessary. When a part of "the business" approaches IT with a set of requirements, those requirements are simply a set of IT capabilities *they believe* will accomplish a particular business objective. The business objective is the real… well, objective, while the requirements themselves are someone's view of how to achieve the objective. The writer of the requirements has an unstated, and often unacknowledged, hypothesis.

People who participate regularly in IT initiatives are constantly surprised by what happens when new IT capabilities are actually rolled out. When I was CIO at US Citizenship and Immigration Services (USCIS), we developed a new, streamlined, all-digital process to speed up adjudication of I-539 applications, applications non-immigrant visitors to the US use to change their status. When adjudicators began using the new process, we discovered that it caused processing time to increase from fifteen minutes to over an hour.

Yet that process had been designed by user interface experts working with adjudicators and their managers, and the system had been built precisely according to their requirements—and tested. For the same reasons that a startup needs to validate its designs through minimal viable products and customer feedback, a company creating an internal system to accomplish a business objective must test the assumptions behind its requirements.

Avoiding the risk of arrogance

Each time someone translates a business objective into a requirement, they have added risk—namely, the risk that a particular requirement will not actually accomplish the objective. The only way to manage that risk is to treat the requirement as a hypothesis, test it, and then modify it as necessary. It is arrogance that leads us to label the hypothesis a "requirement," and it requires humility to accept that your requirement might need to be modified, even if you are paid to be an expert.

After seeing this sort of thing happen again and again, we introduced an IT framework at USCIS where software development was driven directly by the business objective itself. Rather than being presented with a set of requirements to implement, the delivery team—including both technical and business folks—was given a business objective to accomplish. They formulated their own hypotheses about what to deliver to meet that objective, tested their hypotheses, and pivoted based on what they learned.

Be a champion, not impediment, to innovation

Humility doesn't only apply to requirements and product design; it also applies to how managers interact with their employees. With the development of agile IT techniques came an interest in servant leadership, the idea that managers exist to support their employees and remove impediments for them. Since employees are delivering the value-adding product, managers need to provide the conditions under which the employees can do so efficiently and at a high level of quality. Employees frequently run up against obstacles when they are trying to deliver value—bureaucratic rules, subject-matter experts they need to talk to who are unavailable, or even the inability to book rooms for meetings or get access to other needed resources. The job of servant leaders is to quickly make impediments disappear. It takes humility to acknowledge that your employees are the ones creating value, and that your role is to make them successful.

Humility is also important for managers trying to stimulate innovation in their organizations. What happens when an employee approaches his or her manager with a new idea? Does the manager try to poke holes in it—"Did you think of this? Did you think of that? How about this? Did you make sure it's okay with Igor? Did you project revenues? How do you know your projections are right?"

We have been taught that managers are supposed to know more than their employees and that they should avoid risk by weeding out bad ideas. But this is no way to stimulate innovation.

"The job of servant leaders is to quickly make impediments disappear."

About the Author

Mark Schwartz is an Enterprise Strategist at Amazon Web Services and the author of *The Art of Business Value and A Seat at the Table: IT Leadership in the Age of Agility.*

For a humble manager, there is a better way to reduce risk. The manager can assume that he or she doesn't know whether the idea will turn out to be a good one, and instead work with the employee to find a way to quickly and inexpensively test the idea. The manager can coach the employee on finding a good, minimal test, and what success would look like. Then, once the test is conducted, the manager and employee can agree on whether the idea is promising or not.

Adopt practices that encourage humility

DevOps was created to support this humble approach to delivery. A DevOps team tries to reduce lead times for getting capabilities to production so that they can get fast feedback from users. Feedback, by the way, is not necessarily verbal feedback from asking customers what they like. More commonly, it is quantitative measures relating to goals the company has established.

The cloud also helps enable this way of working. With the cloud, it is easy for the enterprise to create a quick MVP and then iteratively refine it. The cloud mitigates risk because any infrastructure or services that have been provisioned can quickly be de-provisioned if an idea doesn't work. The cloud makes the DevOps processes of continuous integration and continuous delivery possible and lets the company put in place security, compliance, and reliability guardrails, which then let that company experiment at high velocity.

Humility enables better companies

Is it surprising that Lean and Agile IT require humility? Or that today's uncertain and rapidly changing business environment demands it?

We have always thought that managers and experts should know what their markets want. But in a business environment of uncertainty, complexity, and rapid change, where customers, regulators, and even nature are full of surprises, it should come as no surprise that humility can improve a company's competitive position, facilitate innovation, and help build relationships with customers.