

# Advantage of a Capability Model

08/25/2023 3:35 pm CDT

Path to Agility embraces the concept that teams and organizations need to develop capabilities and not focus on achieving a level of maturity. We think this was best captured in **the book [Accelerate](#) by Nicole Forsgren, Jez Humble and Gene Kim.**

## Excerpt

Focus on capabilities, not maturity

Technology leaders need to deliver software quickly and reliably to win in the market. For many companies, this requires significant changes to the way we deliver software. The key to successful change is measuring and understanding the right things with a focus on capabilities -- not on maturity.

While maturity models are very popular in the industry, we cannot stress enough that maturity models are not the appropriate tool to use or mindset to have. Instead, shifting to a capabilities model of measurement is essential for organizations wanting to accelerate software delivery. This is due to four factors.

**First**, maturity models focus on helping an organization "arrive" at a mature state and then declare themselves done with their journey, whereas technology transformations should follow a continuous improvement paradigm. Alternatively, capability models focus on helping an organization continually improve and progress, realizing that the technological and business landscape is everchanging. The most innovative companies and highest-performing organizations are always striving to be better and never consider themselves "mature" or "done" with their improvement or transformation journey -- and we see this in our research .

Shifting to a capabilities model of measurement is essential for organizations wanting to accelerate software delivery.

**Second**, maturity models are quite often a "lock-step" or linear formula, prescribing a similar set of technologies, tooling, or capabilities for every set of teams and organizations to progress through. Maturity models assume that "Level 1" and "Level 2" look the same across all teams and organizations, but those of us who work in technology know this is not the case. In contrast, capability models are multidimensional and dynamic, allowing different parts of the organization to take a customized approach to improvement, and focus on capabilities that will give them the most benefit based on their current context and their short- and long-term goals. Teams have their own context, their own systems, their own goals, and their own constraints, and what we should focus on next to accelerate our transformation depends on those things.

**Third**, capability models focus on key outcomes and how the capabilities, or levers, drive improvement in those outcomes -- that is, they are outcome based. This provides technical leadership with clear direction and strategy on high-level goals (with a focus on capabilities to improve key outcomes). It also enables team leaders and individual contributors to set improvement

goals related to the capabilities their team is focusing on for the current time period. Most maturity models simply measure the technical proficiency or tooling install base in an organization without tying it to outcomes. These end up being vanity metrics: while they can be relatively easy to measure, they don't tell us anything about the impact they have on the business.

**Fourth**, maturity models define a static level of technological, process, and organizational abilities to achieve. They do not take into account the ever-changing nature of the technology and business landscape. Our own research and data have confirmed that the industry is changing: what is good enough and even "highperforming" today is no longer good enough in the next year. In contrast, capability models allow for dynamically changing environments and allow teams and organizations to focus on developing the skills and capabilities needed to remain competitive.

By focusing on a capabilities paradigm, organizations can continuously drive improvement. And by focusing on the right capabilities, organizations can drive improvements in their outcomes, allowing them to develop and deliver software with improved speed and stability. In fact, we see that the highest performers do exactly this, continually reaching for gains year over year and never settling for yesterday's accomplishments.