

# Challenging Goliath

David needs an engineering stage to stand up to the tech giants.



May 2022  
By Sebastian Heumann  
and Arne Weiner

As a part of Boston Consulting Group (BCG), BCG Platinion provides consulting & engineering services in human centric design, IT architecture, development, and implementation of advanced technology solutions that fuel critical transformation and creation of new-generation business models. Today, our presence spans the globe with offices in Europe, North and South America, South Africa, and Asia Pacific.

# Challenging Goliath

David needs an engineering stage to stand up to the tech giants.



Countless studies, articles, and blog posts about the impact of digital transformation have already been published. Almost every company on the planet is affected by it: Tech giants like Amazon, Google, or Microsoft as well as digital unicorns with unlimited financial power are entering established industries.

To survive in the market, traditional companies have to radically adapt their business model. And so many have jumped head first into digital transformation. Software engineers and business experts work side by side; they are organized into squads, tribes, guilds, and chapters; they use Scrum, Kanban, or similar tools. In short, the company, its organization, and especially IT are agile today.

## But are they successful in doing so?

In many companies, the developed software has grown into an unmanageable patchwork of technologies, the results of the development organization are based on estimated gut feelings, and driven by the constant talent shortage, the tech discussions are often led by rookies. These employees are certainly highly competent, creative, and efficient in many cases - but they are also inexperienced when it comes to the IT architecture for large-scale business and industry specifics.

## Traditional non-tech companies are usually product-oriented organizations.

They manufacture cars, trade in consumer goods, or offer financial services. Software development is seen as a cost factor for the respective product. In the digital market, the opposite is true. The software is the core product, which is then combined with different hardware to create a new better product like Apple's iPhone or Tesla's EVs.

## Agility alone is not enough to face the tech goliaths on equal footing.

Rather, traditional companies need to build world-class software engineering capabilities that can plan, build, and operate great software products in a competitive, dynamic environment.

As in traditional engineering, the software engineering team also needs a foundation to build outstanding products. This foundation is built on three pillars:

1. **Engineering capability model**
2. **Engineering best practices**
3. **Engineering metrics**

# 1. Engineering capability model

*“The best developers develop the best software – digitalization requires building and retaining world class engineering competence“*

A software engineering organization needs skills but also specific experience. This is quite comparable to a traditional engineering organization. The automotive industry ticks differently than the construction industry or the chemical industry. Engineers from car manufacturers cannot easily switch to polymer production because they have different specific experience. A good team needs both, experienced professionals who know the industry and enthusiastic rookies.

Even in a team with the brightest and most talented minds, each engineer needs to gain experience from “developing a story” to “owning a module” and “leading the system development”. Ideally, this experience is based on multiple technologies and has been proven in different systems of varying complexity. The combination of skills and experience should be reflected throughout the organization, for example in team and leadership roles, personal growth path with feedback and training, and hiring criteria for new employees.



# 2. Engineering best practices

*“Do not reinvent the wheel—Leverage existing experience to build new applications or services based on seasoned templates”*

Compared to traditional engineering disciplines, software development has not yet reached the same professional level of industrialization. For example, although the construction of a bridge is very complex, it still follows many defined industry standards, such as for the construction processes, materials, or structural analysis. However, there is no industry standard for setting up a Kubernetes cluster. Therefore, the engineering organization needs a set of best

practices to serve as a foundation for further developments. These best practices should support the organization’s product goals. For example, they may specify how to write and structure source code, how to define APIs, or how to set up error logging. These practices should not just exist as theoretical guidelines, but rather be implemented in physical code templates that can be immediately used by all teams.



## 3. Engineering metrics

*“Things only get better when they are properly measured – Define specific engineering metrics and measure performance of Dev teams“*

According to a widespread IT legend, agility and performance measurement are mutually exclusive. The argument of many agile teams: The nature of the agile setup simply makes it impossible to make predictions. Consequently, it makes no sense to measure results. Is this true? No, this line of reasoning makes no sense. Successful engineering organizations have metrics to measure their

performance. In engineering - both traditional and software development - numbers are a key element in improving results. Metrics such as throughput (how many planned items were delivered?), planning reliability (how accurate was the plan?), or development rate (where did the team spend its time?) are good indicators of team performance and output.

## What do we learn from this?

Agility is a helpful and important tool for driving digitalization forward. However, it is not a panacea for successful digital transformation. Companies that rely on it alone will not stand a chance against tech goliaths. If they want to operate at the same level, companies must gain excellence in software development and establish a solid foundation with a unique software engineering practice. This is not particularly complicated, but it should be put in the hands

of software engineers who have the highest level of technical experience and the necessary entrepreneurial spirit. After building a competent team, they can develop the capability model definition, best practices, and metrics. Then they can follow an iterative approach: Start small and simple, engage the organization, and successfully scale with individual opportunities to learn, teach, and shape the future engineering team.

# About the Authors

**Sebastian Heumann**

*Associate Director*

Munich

[Heumann.Sebastian@bcgplatinion.com](mailto:Heumann.Sebastian@bcgplatinion.com)

**Arne Weiner**

*Director Agile*

Cologne

[Weiner.Arne@bcgplatinion.com](mailto:Weiner.Arne@bcgplatinion.com)





