Table of Contents

Introduction .................................................................................................................................................. 3
The New Magic Triangle ............................................................................................................................. 3
Engineering Practices in the Survey ............................................................................................................ 4
Towards a Software Company ...................................................................................................................... 5
Insight: Innovation at Volkswagen Group .................................................................................................. 9
“New Normal” Means that New Is Normal ................................................................................................. 10
Acknowledgment ...................................................................................................................................... 11
References .................................................................................................................................................. 11
Introduction
The often-used “new normal” alliteration means that new is normal. This holds specifically as practically all companies evolve towards software companies. Read in this whitepaper conclusions from an industry survey on trends that will impact us in 2022 and beyond. Practical guidance is provided from our consulting projects and includes software productivity, AI, green software, cybersecurity, and agile practices. A case study from Volkswagen shows the transformation to a software company.

At year-end 2021 and in a continuous flow of waves of Covid-19, Vector Consulting with IEEE Software contacted industry experts on their challenges, lessons learned and outlook. We have been asking almost 3000 persons across different industries worldwide. With a response rate of 5.5% covering different industry domains, the survey well represents different B2B business models and diverse regions in the world.

From our parallel interviews with technology leaders, we outline concrete recommendations for winning in this new normal. We summarize survey results and best practices from our recent projects during the lockdown and thereafter. Based on survey findings we derive which survival strategies are available, both hard and soft topics, to succeed in the new normal.

The New Magic Triangle
For decades we have learned a magic triangle has three poles, namely quality, cost and time. With the new normal we see a shift towards a new magic triangle. Participants in the 2022 Vector trends survey observe a new magic triangle with three major challenges.

Innovation. Companies worldwide see innovation as the major challenge both short-term and for the future. The path to innovative solutions is marked by the four ACES that is Autonomy, Convergence, Ecology, and Services. Addressing these basic needs will create new business models to thrive upon [1,2,3]. There is no simple cookbook recipe for innovation. History is littered with the skeletons of companies that slept over innovations and rather maintained short-term cash cows until they starved to death. Kodak invented the digital photography but bet on their classic cash cow of physical imaging, until they were gone. Nokia was the mobile phone market leader and lost within few years due to not improving user experience. Netscape had an extremely strong market position with internet browsers and over 90% market share yet lost in the so-called “browser-war” due to not controlling their architecture and technical debt. Blackberry experimented with novel designs for smartphones yet stayed with keys. The transformation towards a software company is illustrated in our case study from Cariad.

Competences. Software competences are scarce, both in terms of quantity and quality. The global fight for competence will further increase with the push for software innovations [2,3]. The same software engineers who manage the convergence of product IT and enterprise IT are demanded across all industry domains, from IT to IoT, from app development to big systems, from automotive to robotics. During the pandemic years trainings have been reduced in the assumptions, that engineers will somehow learn from online videos. In many client projects we have faced a decline in software engineering skills over the past decade. Today developers know lots about coding and libraries, but not much about architecture, configuration and test methods, systems engineering, and software process [4].

Cost-complexity. Software innovation is severely impacted by the growing cost-complexity trap. Products have grown in terms of embedded and connected software, a variety of features and high demands on quality. What is delivered must be mature. Yet, most software projects are hampered by lack of architecture and top-down systems engineering, piecemeal design, technical debt, lack of robustness, and insufficient testability. Brook’s law still applies after fifty years. Young engineers working from home are frequently thrown to challenging projects in the expectation that they will somehow deliver. However, quality cannot be compensated by quantity as Fred Brooks observed. In consequence, demands on product liability are increasing, and so is legal action due to insufficient quality.

Figure 1 shows this new magic triangle of our 2022 industry survey. The horizontal axis provides short-term challenges, while the vertical axis shows more mid-term obstacles. Each reply allowed up to five challenges in both dimensions, adding to the sum of 100%. It is a magic triangle because each of these factors impacts the others. Innovation is hampered by lack of engineers and unmastered product complexity. Competent software engineers prefer working in start-up situations with much less product and organization legacy and related complexities. Complexity though understood as fueling a downward spiral is not cut, because engineers and sales put too much emphasis on base factors deemed as not negotiable, even if we see with companies such as Tesla and Apple that standard platforms with limited defined variation is the holy grail of business success.
Mitigating the risks of this magic triangle is challenging. The initial reflex of cost first is a recipe for failure. For instance, in 2020 we observed with many clients that trainings of engineers had been cancelled to save on cost. In turn they had been demotivated and lacked the necessary energy and know-how to master the crisis. A vicious circle had started. Today we face its outcome with insufficient competences on board, engineers heavily changing jobs, and thus a lack of innovation. In parallel technical debt of products was increasing with the effect of even higher effort today on legacy than what would be reasonable. Cost reduction programs negatively impact innovation and quality. The worst with such programs is that they are self-fueling. Lower quality and less innovative products inevitably increase red ocean, which further fuels the cost pressure. It is a downward moving cycle that many companies currently feel. In fact, we have seen several once strong players drowning in cost savings programs starting in 2020, and now being unable to recuperate along the necessary software innovations.

Complexity management with agile methods and systems engineering is one relevant answer to cope with this magic triangle [4,5]. It reduces complexity and thus allows to focus limited competences on real innovation. It connects agile methods such as “value focus” and “waste reduction” with a consistent system understanding by means of modeling and traceability to avoid those scarce resources are wasted in firefighting. Let us look towards recommendations for these challenges which we have identified in our projects and in numerous interviews with industry experts over the past months.

**Engineering Practices in the Survey**

In looking to answers to address the magic triangle of the new normal we looked to the forces at its poles. One question was about what the major differences of the “new normal” are compared to the pre-pandemic way of working in the “old normal” before 2020. The focus of the question was about what is perceived as a stable change which will be preserved in our new working modes. Fig. 2 provides the results. Each respondent could name a maximum of three impacts which explains why we see a sum of over 100%. Not surprisingly online work and training rank first. Most of our clients plan now that engineers will spend some two days a week in office, with the remaining time working from home office. One world-market leader as a tier-1 already went as far as having changed contracts that engineers would only occupy office for 10% of their time. This allows dramatically reducing office capacity and thus cost, while adjusting to offices which will remain anyway empty. It will also change office allocation from fixed rooms to mobile office with the advantage that those who are in office will also better interact due to proximity.

More relevant that innovation and digital transformation are also ranking high, combined with need for more efficiency. Success with a software-driven company needs digital transformation across workflows, organization, and value streams [1,4,5,6].
Within the survey we also looked to the major success factors of the new normal. Fig. 3 shows the major themes. Not surprisingly those are primarily about soft skills and work culture. Technologies and tools have been around for a while yet processes and practices need to still be adapted to new work styles. Commitment and task management rank high as engineers have learned that not everybody working from home and without direct supervision will deliver as planned [2]. Some of our clients observed that half of their engineering work force degraded in terms of commitment and quality.

With the pandemic cocooning companies face a commitment divide of their engineers, as we had been often told in the past year. One part of the team is highly committed and is working in task forces and on weekends. Others are neither visible nor delivering. The situation cannot be repaired by technology alone but needs leadership which takes benefit of the described tools to motivate and follow up. Good use of collaboration tools will help integrating the entire teams to tasks – with visible participation and progress [6].

Towards a Software Company

In talking with many companies and industry leaders through the past year, we have distilled five key success factors for software-driven innovations, namely:

> Create new needs
> Build on an existing platform
To understand these five innovation guidelines, let us look to a simple, yet powerful, software-driven innovation. Some 50 years ago Ray Tomlinson developed the first electronic mail. Working on ARPANET (Advanced Research Projects Agency Network), a network connecting computers and predecessor of today’s Internet, he discovered that users communicated by messaging to other users on the same computer. He thought beyond this simplistic pattern and anticipated sending asynchronous messages to any computer — worldwide. Tomlinson used the “@” to indicate a destination in the format <username>@<name of computer>, which is essentially how email has been addressed ever since. When inventing the e-mail fifty years ago, in innovating collaboration software, he followed the above five success factors. He stimulated new needs to allow people asynchronous communication. He used an existing platform rather than waiting for something fancy and perfect. He worked in a small team and incrementally improved the service. He proved value by bringing the results to his community, where he for instance coached a woman who used e-mail to create one of the first newsletters — and eventually became his wife.

Let us now dive into technology evolution. Fig. 4 shows major evolution paths structured along business, technologies and working modes. A major theme is the move from software to services. This is often abbreviated with XaaS, that is everything as a service. Software-driven services allow flexibility. Business-oriented services generate value to the user [1,2,3]. They are highly money-generating, if done well. Unlike in the past, we now have the technology. It follows the Kano model at its best because a good service on top of even a mediocre product can be a real excitement. Give a 24/7 online support and you earn a big “wow” if you deliver.

Digital innovation happens today in practically all industry domains [1,2,3]. It is just not sufficiently shared to learn from. The medical industry for instance, since long is dealing with very similar challenges as we have today in IoT and automotive. Medical implants connect with cloud services, medical image analysis is a powerhouse of AI, digital twins have been used long before they became fashionable in automotive and industry automation, and regulations and homologation are certainly more demanding than in automotive and aerospace. Digitalization enables new treatments and certainly more efficiency in healthcare workflows. More data is generated, hence more need for AI to analyze. The future of healthcare brings fundamental changes. Today worldwide medical data and thus knowledge doubles every 73 days according to Siemens Healthcareers. This creates a relentless pressure to innovate.

Ecologic behavior is the need across the world to mitigate impacts of climate change. Software and IT play a pivotal role towards ecologic behaviors for many reasons [6]. Being aware that IT systems alone already consume 10% of global electricity, the leading software practitioners must embark on green IT and green coding. Green IT and green coding are describing a paradigm switch how software engineers, developers, testers, and IT administrators can make their solutions and services more energy efficient. As a designer ensure that new software versions do not require more computing power, hard-disk space, or bandwidth than before. This is easily feasible by optimizing algorithms. A simple checkpoint is that previous core functionality remains available on older hardware. As a developer for embedded software or IoT components, i.e., massively deployed software components residing close to energy hungry-hardware, there are several best practices for green software, e.g., avoid always-on, choose efficient algorithms, use pulse- rather than continuous connectivity (which also better protects your systems for cybersecurity), reduce unnecessary precision and storage demands. As a UX designer, make sure that users can configure software according to their individual needs, such as night mode, movie blocking, reduced resolution movies etc. UX should clearly show the power usage of the respective software. Create the awareness by showing with red/amber/green how much energy is currently consumed.

With the digital transformation, liability and governance must receive more attention than before [1,3]. Connectivity makes software products and companies a primary attack target for hackers. Risk-oriented cybersecurity is the need of the hour. Ensure strict guidelines on how software, both design and data are handled. Security is a lifecycle activity with cross-cutting requirements considered at all stages. We recommend continuous code quality management. Static code analysis throws a huge bunch of errors, which by engineers is perceived as slowdown. So, the most relevant and the most critical risks must be selected in each sprint. A combination of both agile and DevOps facilitates automatic verification.

Balancing quality and cost directly impact architecture and design decisions. Many companies focus too much on product development and not enough on systems engineering, architecture, and cross-cutting features such as security and performance. Unsystematic testing
is a major problem in most companies as it not only wastes effort due to too many overlaps, but also bears the risk of blind spots due to lack of decent coverage measurements. Introduce KPIs for testing. Reduce test cases towards those that matter. Automate testing towards the 100%. Testers overlay multiple approaches without overarching test strategy. For instance, we face heavy focus on integration but no criteria for test case selection and coverage. Unit test is growing but static analysis is hardly applied, while being much more sustainable. Agile quality methods such as Test-Driven Development (TDD) and Test-Driven Requirements Engineering (TDRE) help to focus test on where it is necessary.

Software-driven innovations to our observation in various client projects are often hampered by insufficient internal alignment of technologies, competences, management, processes. Companies often struggle with heterogeneous and legacy tool-chains and processes. Bigger organizations suffer from “over-the-fence” mentality where artifacts are thrown to another location, which will eventually start a ping-pong of who must solve which problem. Overly complex frameworks such as agile SAFe also contribute, which creates even more burden and complexity than historic frameworks such as CMMI. Complexity is increasing with multiple development partners and the need to coordinate them. To mitigate such risks, Vector has been contributing to the recent guideline on agile collaboration. The objective is to facilitate new forms of development, partnerships between organization and several levels of collaboration such as linked, aligned, and combined.

Innovation with autonomous systems gained speed recently as some legal obstacles have been removed. Vector is not only active with its products such as data loggers, but also in standardization such as safety of the intended functionality (SOTIF). A major theme in Vector consulting is novel testing methods. Vector as the testing company has strong credibility in testing of autonomous systems and is asked from companies worldwide on how to innovate testing methods, away from brute force and questionable coverage schemes. Bodenhausen explains, Data matters in AI and machine learning, as well as testing. The saying is there is no data like more data. But this means brute force with high effort and still big risks of overlooking a critical corner case. Synthetic generation of data will facilitate efficient data generation while capturing the corner cases. In this topic, Vector is highly engaged with various research projects in this field, such as University of Stuttgart with its Robo-Test incubator.

Cost pressure and lack of competences can only be addressed with steady productivity improvement. Automated workflows will reduce the interface ping-pong and organizational frictions. Working in the new normal needs a yardstick to measure productivity in remote situations. Agile with its focus on getting things done, facilitates such tracking mechanisms. A work product that is ready and can be installed is seen as showing progress. Teams which look to work progress based on value-oriented deliverables always see what went right and what went wrong. A clear Definition of Done (DoD) is key and can be tracked by status indicators in each work products. Keep it simple. It is not about many statuses, but rather binary: Done and open. One of the most eminent agile failure points of many teams and projects is that there is no committed DoD with measurably exit criteria. For instance, Test-Driven Requirements Engineering (TDRE) allows already during elicitation that software requirements can serve as a test case, and ensure they are traceable to the market requirements.

<table>
<thead>
<tr>
<th>Business</th>
<th>Tangible products</th>
<th>XaaS, Services, fluid eco-systems</th>
</tr>
</thead>
<tbody>
<tr>
<td>IT</td>
<td>IT versus embedded</td>
<td>Convergence of business IT and product IT</td>
</tr>
<tr>
<td>Risks</td>
<td>Project cost, delays</td>
<td>Product liability, complexity, security</td>
</tr>
<tr>
<td>Technology</td>
<td>Components, libraries</td>
<td>Services, AI/ML, edge computing, Green IT</td>
</tr>
<tr>
<td>Architecture</td>
<td>Monolithic, HW-driven</td>
<td>Flexible, cloud+edge, SOA, micro-services</td>
</tr>
<tr>
<td>Methodology</td>
<td>Static, function-driven</td>
<td>Systems engineering, modeling, low-code</td>
</tr>
<tr>
<td>Testing</td>
<td>Ad-hoc, coverage</td>
<td>Efficiency, TDD, TDRE, RPA, cognitive testing</td>
</tr>
<tr>
<td>Lifecycle</td>
<td>Versions, variants</td>
<td>Evolutionary, ContinuousX, DevOps</td>
</tr>
<tr>
<td>Teams</td>
<td>Function oriented</td>
<td>Distributed, ecosystems</td>
</tr>
<tr>
<td>Tools/Process</td>
<td>Legacy, clumsy</td>
<td>Federated ALM/PLM, agile, flexible, sharing</td>
</tr>
</tbody>
</table>

Fig. 4: Technology Trends – The Future Has Already Arrived, However Not Yet for Everybody
Agile allows to combine innovation and efficiency. Agile development methods such as Scrum with Kanban are useful and ready-to-use, yet often appear as a slogan than efficient work organization. A major lever is complexity control. The past years have accumulated lots of unnecessary features, gold-plating and variation in products which must be trimmed down. Same with processes. Heavy governance schemes add towards complexity. We recommend allocating requirements and features to life-cycle cost and optimize platform-based reuse.

Yet agile is far from trivial. Fig. 5 with results from our survey in 2021 show that a vast majority of respondents face new risk and problems triggered by agile. With clients we often see organizations in the trap of scaling down with what they consider agile behaviors, but rather reducing quality and heaping up technical debts. Agile transformations primarily fail due to insufficient change management [4,5]. We strongly recommend using professional change management. Tailor agile principles around your specific needs and company culture. No two agile transformations are the same.

Fig. 5: Agile Working Brings New Risks and Challenges

Flexibility is key – in the new normal but especially for software engineers. We need to be prepared to work across time zones and at the same time be sufficiently mobile to work in different sites, from home office to offices at the premises of our company and our customers, up to being flexible with work assignments. Gone are the days of a written job profile. Fig. 6 shows the various perceived patterns of flexibility across respondents of the survey.

Create a culture and mindset of fearlessness – to embrace new ideas, novel methods, and risk-taking mentality. Why is it so important to be fearless? Because it drives innovation. It redefines standards of the future. It allows your talents to learn, grow and flourish. It attracts the right skills and talents for this transformation from manufacture-based innovation to services innovation.
Fig. 6: Flexibility is Key for Engineers

**Insight: Innovation at Volkswagen Group**

Our ambition at Vector Consulting is to share experiences from global leaders and thus facilitate learning and translating to your own environments. This case study of Volkswagen Group provides insights how to innovate with software. During the pandemic lock-down during 2020/21, Volkswagen has created its new global software division, called Cariad. While others have been complaining about the constraints with the lock-down, Volkswagen group has taken it as trigger for software-driven innovation. The name is program, as Cariad stands for “Car. I Am Digital.”

The convergence of business IT and product IT is visible in vehicles as well as practically all other industries. Traditional vehicle engineering is evolving to systems and software engineering of mobility services. Same in medical and finance where a variety of bots increasingly take over diagnostic and advising functions with higher precision and quality than human could do. This needs dedicated architectures to handle the rising complexity. In implementing the new architectures, we currently see an evolution away from traditional functional design and the current domain architectures towards future zone architectures which are cloud-oriented and can be easily adjusted with new content, both user experience, functionality, services, and software. Obviously not all functions will move to cloud-based services. Comfort based functions may move into domain or zone-based, while safety critical and real-time will remain as individual smart actuators within the product IT.

Users, specifically the young adults benchmark enterprise and product IT with smartphone-like UX expectations. Features such as automated driving functions, OLED displays, personalized infotainment, augmented reality, all serve one purpose, namely, to provide an advanced user experience (UX) and thus satisfy clients beyond their basic expectations. Without a compelling and exciting user experience, it will be difficult to retain customers and gain new customers. This evolution leads to the software-defined vehicle with much more computing power than what we are used to. Smart actuators will merge various functions and demand fast and reliable bus systems for efficient communication. With their specific demands on functional safety and real-time performance, they operate independent of the cloud, yet consider inputs from the cloud. Imagine a braking system in a vehicle being aware that there is a slippery or icy road ahead. It can adjust to these changing road conditions and thus improve safety of vehicle and driver.

Dr. Maximilian Montenbruck, department head for vehicle motion at Porsche and now also working in Cariad summarizes some of the key learning along the way towards a software company. He sees three major methodical challenges in developing software with high functional integration across industries:

- **Architecture**
- **Toolchain**, 
These challenges interact with each other and thus cannot be solved in isolation.

Traditionally, there was an organization which established a tool chain and then looked to software architecture. This is changing today, by giving priority to architecture. Across industries, the three-tier layered architecture with distributed responsibilities is established. Fig. 6 shows this evolution which can be mapped to the abstraction layers of the ISO/OSI stack. Porsche’s current solution is the HCP platform project with high-performance computers in the middle tier serving intelligent actuators and sensors and connecting to the cloud for flexible updates.

The high quality of Porsche cars demands software updates over a long life-cycle, much longer than other cars. This needs thorough software updates management and high emphasis on cybersecurity. Flexible and adaptive software packages being continuously updated create new challenges towards quality and testing. Often the new technologies demand heavy test efforts in brute force mode. With new technologies, machine learning, adaptive AUTOSAR, all deal with real time communication. It is tough to have test depth. New test methods and coverage schemes are demanded.

Looking to sometimes bizarre software and architecture decisions in our industries, Montenbruck warns about Conway’s law: “No matter what you do, the software architecture will resemble the organizational architecture.” Volkswagen group is mitigating this risk with new architecture paradigms. The new software organization is creating modular software with low interaction and coupling together with a strong toolchain to overcome this legacy organization trap.

**Fig. 6: Layered Architecture Facilitates Distributed Responsibilities**

**“New Normal” Means that New Is Normal**

Today the smartphone is a benchmark for any software-driven innovation, from user experience to platforms, and from continuous upgrades to highly integrated software stacks. Indeed, there is a lot we can learn from smartphones. For instance, platform strategies, continuous deployment, flexible services and – most relevant – an extraordinary user experience are much more advanced than what most industries currently deliver. But safety-critical industry products such as automotive, medical and transport are more challenging. They demand functional safety and very high reliability. For instance, when there are adaptive algorithms in vehicles, their behaviors must be specified, approved, and homologized. Underlying AI rules must be transparent, and upon self-learning, it must be ensured that rules that are still valid, would not be overwritten.

Traditional IT systems have centrally governed replicated operations, which for instance facilitate flexible updates and roll-back in case of deficiencies. They also allow a smooth transition towards new hardware and software stacks. Converging systems must be autonomous in their usage and updates of software. While many IT systems, both apps as well as cloud-services, often crash or restart, mission-critical devices at most are allowed to fail operationally, not to endanger humans. While smartphones and consumer goods are thrown
away after few years, embedded hardware platforms are expected to run for many years, sometimes decades as we see in transport. They are updated and maintained long after end of production, while traditional – and not ecologic thinking – IT suppliers stop support a few years after release. In short, while we enjoy a smartphone-like user experience, we don’t want to face its downsides, such as frequent crashes and nondeterministic behaviors.

Software-driven innovation not just happens. It demands hard work. Statesman Winston Churchill, who certainly has seen more trouble than most of us, once stated: “A pessimist sees the difficulty in every opportunity. An optimist sees the opportunity in every difficulty.” Let us advance in the new normal with such optimistic attitude. There is still a lot in front of us to be done.

Acknowledgment

We want to thank all survey participants for supporting this study and thus ensuring validity. Specifically, we appreciate our clients and global consulting team for allowing us together to strive for continuous improvement – and thus boost innovation and competitiveness.

References


Authors:

**Dr. Christof Ebert** is the managing director of Vector Consulting Services. Many Fortune 100 companies have already used his competence to improve their performance and competitiveness. As a business angel and professor at the University of Stuttgart and the Sorbonne University in Paris, he facilitates innovative business. He serves on the editorial boards of IEEE Software and other leading journals. In his leisure time he is active on various keyboards and engaged in social and community work.

Follow Christof on Twitter and LinkedIn: @ChristofEbert

Tel. +49-711-80670-1520

E-Mail: consulting-info@vector.com

[www.vector.com/consulting](http://www.vector.com/consulting)

Twitter: @VectorVCS

---

Revised: 14. February 2022
Press Contact:
Vector Consulting Services, Germany
Stuttgart
Ms. Lena Kast
Phone: +49 711 80670-1520
E-mail: consulting-info@vector.com

About Vector Consulting Services (Revised: September 2021):

Vector Consulting Services is a globally active consulting firm with focus on development and IT, agile transformation and interim management. Renowned companies from automotive, information technology, manufacturing, transport and aerospace rely on the professional solutions and pragmatic implementation. A subsidiary of the Vector Group with over 3300 employees, VCS supports its clients worldwide with sustainable consulting solutions covering the entire life cycle and the related infrastructure. To ensure independent and customer-oriented consulting the firm is managed by partners.

Details and further information: www.vector.com/consulting