COMMERZBANK

CI/CD White Paper

Beyond Banking: How Continuous Integration & Continuous Deployment (CI/CD) Accelerates Customer Value Creation in the Financial Services Industry

The bank at your side

BEI Business Engineering Institute St. Gallen

This White Paper was developed in a collaborative research project between Commerzbank and the Business Engineering Institute St. Gallen.

Impressum

Publisher:
Commerzbank AG
60261 Frankfurt am Main, Germany
+49 69 136-20
info@commerzbank.com

Group Services
Technology Foundations
Cluster CI/CD CHAMP

Design and Layout
Steffen Rädle, Lil-Britt McBrien

Image Credits
Logo by courtesy of Business Engineering Institute St. Gallen (page 2),
Unsplash: C. Käppeli (1) , A. Pettitt (21), A. Ismail (27), P. Richier (39);
Shutterstock (30); Pixabay (16, 40)
All other images: Commerzbank

Beyond Banking: How Continuous Integration & Continuous Deployment (CI/CD) Accelerates Customer Value Creation in the Financial Services Industry

Insights on Commerzbank's Holistic CI/CD Approach

Dear Reader,

The market has changed. Digitization is in demand everywhere today. While in the past it was mainly about hardware in the financial industry, today it is primarily about software. And the way how to deliver software defines the success of every player in the market since today a bank is a software-driven company. Because at Commerzbank, we accept that our DNA has changed. We are now a technology-driven innovation bank.
The concept of Continuous Integration and Continuous Deployment (CI/CD) has gained momentum and changed the way software is developed within the financial services industry. The driving force behind this momentum is acceleration which is key to deliver high-quality and customer-centric software products. At Commerzbank, we have started our CI/CD journey around 3 years ago alongside with a dramatic organizational change to agility. We are looking forward to sharing with you our experience and best practices. Together with various experts from different companies and industries we took a look into the future of CI/CD.
I hope that you will enjoy reading this white paper and gain interesting insights into the concept of CI/CD and how Commerzbank applies this concept to develop superior products for its customers.

Best regards,

Florian Meiser (Commerzbank AG)

Content

Executive Summary

In this white paper, we aim to describe how the software development process can be accelerated by pinpointing the CI/CD concept: its cornerstones, how it affects customer value creation in general and in financial services in particular, how Commerzbank is leveraging CI/CD to drive innovation and shorten release cycles, and which developments we can expect in the upcoming five years in agile software development.
We concluded that CI/CD is a set of best practices and tools which enable organizations to accelerate customer value creation, time-to-market, and the delivery of software quality. We would like to share with you the five key take-aways we identified during our CI/CD journey and the research process of this white paper:

01

CI/CD accelerates customer value creation in the financial services industry

The objective of CI/CD is to be able to daily release high-quality software that creates value for the customer. At Commerzbank, we believe this especially for organizations in the financial services industry which are still undergoing a radical shift driven by increasing competition, high regulatory requirements, structural low interest rates, and fast changing customer needs, CI/CD is key to being closer to the customer. CI/CD enables greater automation in the whole software development and delivery process. This allows us to release daily, new digital banking solutions to our customers and in turn results in a higher customer value.

02

The implementation of a CI/CD pipeline needs to follow a set of technical principles

The CI/CD pipeline is the backbone of every CI/CD approach. It reflects the automation engine alongside the software development lifecycle and is governed by the technical principles of full automation through highly integrated tools connected via APIs. What`s more, a standardized and scalable pipeline, high reliability and availability, and a technology agnostic pipeline are incorporated. Although an effective CI/CD pipeline is built upon these principles, there is no one-size-fits-all approach to implementing a CI/CD pipeline. Since its setup is determined by company-specific internal processes, resources, and infrastructure. The approach of continually gathering feedback during the implementation phase and adjusting the process accordingly is best-practice we discovered. Furthermore, organizations should accentuate those technical elements in their CI/CD pipelines that fit their organizational conditions best, in accordance with regulations to successfully deliver value to their customer.

03

CI/CD is not just about tools and processes, it is also about teams that have end-to-end responsibility

A successful CI/CD approach significantly depends on giving teams end-to-end-responsibility of the software development lifecycle so that they can create software features without significant dependencies on other teams to accelerate speed, quality, and customer value creation [1]. This requires organizations to enable their teams technologically and culturally with the right tools and methods. Our experience also shows that the combination of a self-service platform and availability of an ongoing technical and methodological support for teams, if necessary, lays the foundation for teams to take over end-to-end responsibility within the whole CI/CD software development lifecycle. Although, organizations in the financial services industry need to comply with specific regulations concerning the separation of development and operations entities. Establishing end-to-end responsibility of teams further accelerates the CI/CD software development process.

04

An empowering CI/CD approach leveraging the strength of cross-functional teams is key

Building on the foundation of agile software development, cross-functional teams comprising of development and operational functions, alongside business and security experts ensure the targeting of key business outcomes and compliance with security standards throughout the CI/CD lifecycle. We are convinced that "BizDevSecOps" [2] teams will be the new form of collaboration and co-innovation and a decisive factor for organizations to deliver software more quickly, safe and responsive to customer demand. To implement this new team constellation, organizations need to empower their teams with the appropriate training and coaching solutions to bridge the several functions and enable them to carry on the different tasks throughout the CI/CD lifecycle. We think that the alignment of software development and business objectives in accordance with security standards and built upon shared business metrics significantly contributes that the whole team. This enables working towards the shared goal of accelerating the software development process.

05

Everything as Code (EaC) as the new norm, meaning automation via software will be everywhere

We expect that the EaC development will accelerate in the upcoming five years leading to the practice that everything such as operations, infrastructure and configuration management adheres to the same software development practices [3]. The EaC principle plays a central role for other developments such as enhanced cross-functional collaboration or a shift left approach of security, compliance, and documentation. Additionally, EaC provides the technical conditions for establishing a common CI/CD platform that serves as single point of truth for data, documentation, and tracking points that can be established to measure the speed or waste in certain phases of the software development lifecycle. Cross-functional teams using this CI/CD platform in their daily business will use a common language of code that enables collaboration and co-innovation among the different functions. We defined EaC as one important component of our future CI/CD approach and will follow a zero-documentation approach which ensures an automated and compliant documentation processes along the CI/CD lifecycle.

---

[1] For more information, please see our learnings from CI/CD Use Cases on p. 18.
[2] For a detailed explanation of BizDevSecOps, please see p. 24.
[3] For a detailed explanation of software development practices, please see p. 17.

Introduction

As organizations continue their journey of accelerating their agile software development process, Continuous Integration and Continuous Development (CI/CD) has become crucial to deliver customer value faster [1]. With the need to deliver quickly high-quality and customer-centric software, organizations focus on CI/CD to develop and release software in short iteration cycles supported by cross-functional collaboration, automation, and tools. Today, CI/CD forms the operational backbone in a modern DevOps environment while combining automation and cross-functional collaboration from customer needs and business requirements to software development, testing, and operations to improve the overall software development process.

CI/CD as a set of practices and tools enables a faster time to market through rapid, continuous integration of code changes, automated tests and continuous deployments of every code change to production after successfully passing all quality gates [2]. The concept rests on the interrelated cornerstones of the right tools, streamlined and automated processes, and skilled people that if synchronized drive collaboration, innovation, and automation within the organizations which ultimately improve customer experience. **"CI/CD offers organizations immense value: faster release of products and services, stronger customer engagement, quicker feedback on new features, higher-quality code, and ultimately higher customer value,"** says Jörg Oliveri del Castillo-Schulz (Commerzbank AG).

Specifically, the financial services industry might benefit from these advantages since the industry faces significant challenges due to the macroeconomic situation, fierce competition, and high customer expectations [3]. However, the way in which CI/CD is applied in the financial service industry is significantly impacted by national and international regulatory requirements, that need to be considered when designing and introducing CI/CD in agile software development process. Additionally, in the time of the digital transformation of the whole financial services industry, a financial institution is a software-driven organization that needs to use state-of-the-art technologies to provide customers with an exceptional user experience.

At Commerzbank, we have started our work to build a CI/CD Pipeline in 2018 to technically support our vision of being the bank at the side of our customers using an integrated approach. **"With the commitment to provide superior customer experience by offering digital, individualized and highly valuable products and services, at Commerzbank we have the mission to deliver end-to-end solutions through the CI/CD "Commerzbank Hyper Acceleration Master Pipeline" (CHAMP [4]) for all other clusters [5] within Commerzbank,"** says Florian Meiser (Commerzbank AG). Our objective is to support Commerzbank`s digital transformation and the delivery of innovative solutions for customers by providing a fully automated self-service pipeline that comprises all software development steps from the initial idea to the final product while assuring compliance with regulations and enforcing the spread of modern software development techniques.

With this white paper, at Commerzbank we aimed at creating a common understanding of the CI/CD concept, especially about the way how it affects customer value creation. Moreover, we want to give you a detailed insight into our holistic CI/CD approach and would like to share with you our experience on our CI/CD journey so far – things we have learned and our best practices that might hopefully support you in your upcoming agile software development endeavors. Additionally, we discussed with 14 subject matter experts from different industries in which direction agile software development is expected to heading to in the next five years. We will provide you with the result in the form of five central hypotheses about the future of agile software development until 2026.

Today, CI/CD has become an essential technology enabler for creating customer value, and we expect that the way in which software is developed within companies will become an even more decisive success factor. We hope that this white paper will provide insightful information on our Commerzbank CI/CD CHAMP approach, contribute to the discussion of the future of agile software development, and trigger new questions that will drive all of us towards new horizons of agile software development.

[4] For a detailed explanation, please see "How Commerzbank is living CI/CD" on p. 13.
[5] For a detailed explanation refer to "Glossary", page 36.

Continuous Integration & Continuous Deployment – What is it About?

As software continues to eat the world, organizations are required to accelerate their software development process. In this context, CI/CD has become a key technological enabler for the fast delivery of high-quality software features that satisfy customer expectations. But what is CI/CD and how does it benefit organizations? And which aspects are underlying? The following chapter aims at demystifying the concept of CI/CD by providing the theoretical foundations of this concept including its cornerstones and possible business impact. Additionally, we explain why CI/CD is one crucial technology foundation for Commerzbank's agile transformation and how we leverage this set of technical practices to maximize value for our customers.

CI/CD Explained

CI/CD has its roots in agile software development [1], which aims at a software development process being lean, collaborative, goal-oriented, and responsive to change and customer feedback [4]. In essence, the objective of agile software development is to produce high-quality software that best satisfies customer needs [4]by cultivating and promoting organizational capabilities such as sensing, learning, adaptability, resilience, speed, innovation, collaboration, and efficiency [5]. One approach to operationalize these capabilities is following the paradigm of DevOps, which aims at bridging the gap between the functions of development and operations and aligning their activities across the organization [2][6]. On the one hand, DevOps combines functional skills from business, software development, testing, and operations in one integrated team improving cross-functional collaboration and communication [2]. On the other hand, it empowers teams with end-to-end responsibility which increases the ability to faster deliver high quality software.

This stands in contrast to the formerly applied software development processes, where functional and informational silos or step-by-step sequential processes lead to long development times: "in enterprise software 1-2 years, in new microprocessors 2-4 years, in automobiles 3-5 years, and in aircraft a decade" [7]. Before DevOps, software was often developed isolated as a single piece and, after finishing the complete development, tested as a whole [8]. Now, crucial activities, such as testing, are parallelized with the development phase [9] and software is developed in iterative cycles, building on intermediary results and feedback loops [10]. Hence, possible pitfalls can be detected faster, and solutions can be implemented more quickly. Another important strength is the increasingly integrated technology stack comprising development, testing, and integration tools [1], which will be connected throughout all essential software development steps [2].

The concept of CI/CD forms the operational backbone of DevOps practices and enables the automation and acceleration of the whole software development process, from planning software over testing it to finally deploying software to production [11]. As seen in figure 1 CI/CD is composed of two interrelated value streams – a continuous integration and continuous deployment cornerstone covering the respective activities of the software development process.

In accordance with the overarching idea of agile software development, CI facilitates merging of software changes from multiple contributors to a shared source code repository frequently, often many times a day [12]. After each merge, the chunk of code is automatically built and tested to ensure the code is still functional after each code change, to avoid potential integration issues and to guarantee that it is safe for other developers to write code on top of the new changes [13]. Companies that start with introducing CI usually face two challenges comprising changing the habits of developers to do smaller frequent code commits and ensuring the needed automation to support the software development flow and reduce the time to integrate and validate code changes. The validation itself comprises not only the automated unit tests but also the automated static code analysis and security scans to early detect code vulnerabilities. The practice of shifting this validation while performing unit or integration tests and code quality checks early in the software development process is referred to as "shift left" and can result in significant cost reduction as well as in increased software quality since potential pitfalls in the software code are identified early during the development process [12].

CD picks up where CI ends comprising the software development process steps of deploying and releasing to production. Within the CD phase the software artefact is automatically launched and distributed to end-users after it has successfully passed automated tests to validate if changes to a codebase are correct and stable for immediate automated deployment to the production environment [14]. In living customer centricity, it is important that new products or features are deployed smoothly and without delay. CD complements the process stage of CI leading to a faster and smoother release of new software and makes it possible to continuously integrate customer feedback early in the product lifecycle.

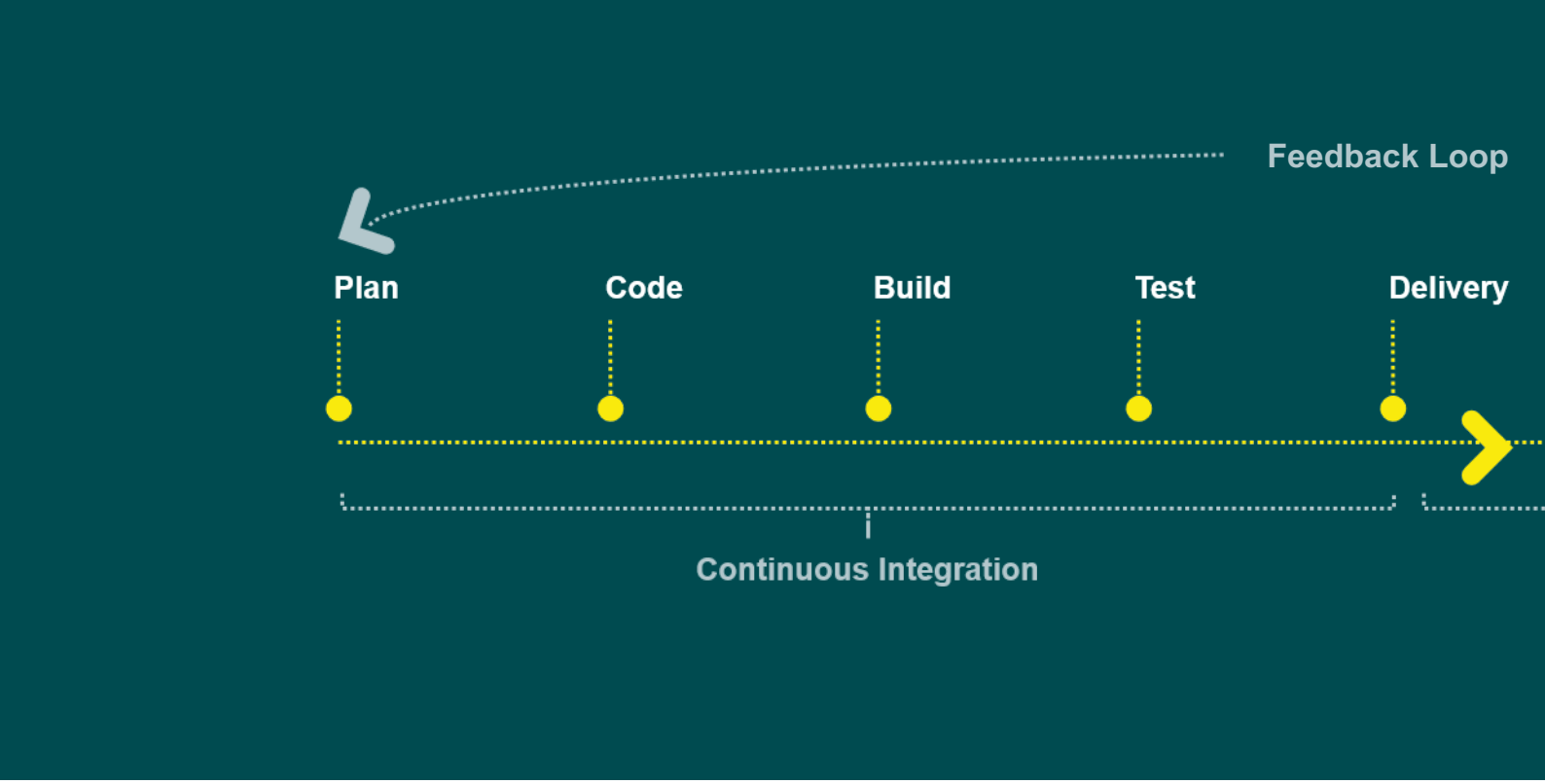


Figure 1: The CI/CD Lifecycle and its Process Steps

According to the CI/CD lifecycle view, all steps of the software development process are continuously integrated and automated to ensure a seamless and holistic software development process. The automation of the CI and CD process steps is typically referred to as "CI/CD pipelines", an analogy of traditional factory product automation pipelines. CI/CD pipelines are often complemented by an integrated customer feedback loop which allows organizations to measure customer reactions, collect metrics and actionable insights from software in production. This feedback loop closes the CI/CD cycle and ensures quality of the software product or feature beyond deployment and a smooth customer experience.

"Overall, the concept of CI/CD is an important enabler for organizations to accelerate the software development process leading to a potential reduction of the time-to-market of new products and features to customer," says Sabine Vigelius (Commerzbank AG). In this context, the benefits of CI/CD are manifold. First, CI/CD leads to a shorter time-to-market since development teams can release software builds faster due to incremental software development [1]. Second, CI/CD ensures an early detection and fixing of software vulnerabilities and defects due to the shift left approach which ultimately results in higher code quality [15]. Third, the automation inherent in the CI/CD lifecycle ensures that crucial process steps of the software development process are less prone to human error and easier to manage by the team [16]. Fourth, CI/CD reduces the cost of communication as it supports the DevOps practice of aligning cross-functional teams. Fifth, CI/CD allows for integrating customer feedback early in the software product lifecycle enabled by short feedback-loops [1].

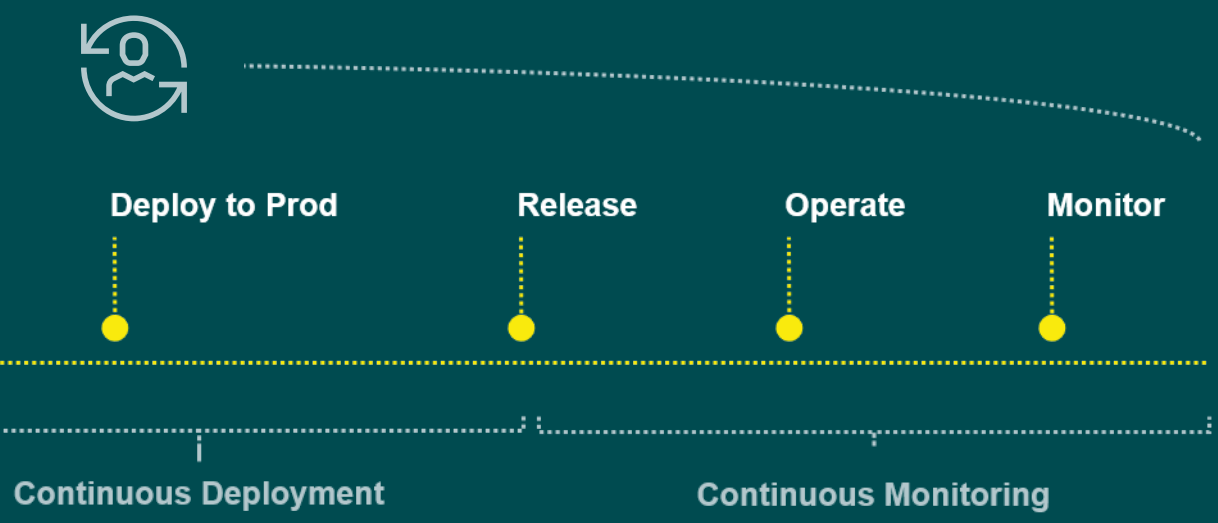
The Cornerstones of CI/CD

The concept of CI/CD rests on the cornerstones of the right tools, streamlined and automated processes, and skilled people, as illustrated in figure 2, to succeed by continually providing value to the customer. It is imperative to take a holistic view when implementing CI/CD since it is the multi-dimensional aspect of all three cornerstones that drive collaboration, innovation, and automation. In the following we want to further describe the characteristics of each cornerstone and how they interact with each other.

1. DevOps Teams

Excelling in CI/CD requires the right people with the appropriate skills and mindset as it applies to the successful implementation of almost every agile software development practice [16] accumulated in one DevOps team. A recent study of the DevOps Institute [17] elaborates that people skills such as collaborating, problem solving, knowledge transfer and adaptability are key for success in living DevOps and CI/CD. Additionally, individuals with an open and innovative mindset are needed to further develop existing products and services to deliver customer value. According to the State of DevOps Report 2021 [18], organizations that have successfully adopted the concept of CI/CD ensure that their DevOps teams have strong identities, clear responsibilities with a high degree of autonomy in their own function as well as distinct interaction ceremonies and communication channels with other teams.

A corporate culture that promotes a workforce living CI/CD principles and successfully applying the technical practices



of the CI/CD concept is based on a collaborative environment [19]. Another important aspect is to impose end-to-end responsibility among DevOps teams that work with CI/CD practices and tools ensuring that all team members together are fully accountable for the software artefact across the whole software development cycle. This supports a collaborative team spirit while providing the possibility for each team member to contribute his or her ideas and strengths during the whole software development life cycle [17]. Last but not least, encouraging continuous learning. This helps us support the development of DevOps culture. This can be achieved by promoting employees to embrace failure and learn from it by creating a psychological safe environment through meaningful and constructive feedback [17].

Developing an appropriate organizational culture that promotes the required CI/CD skills and mindset requires cultural change. In this context, the ability of employees to embrace change is one important, but challenging factor. A study of Atlassian and CITE Research [20] states that almost 35% of the interviewed professionals consider the corporate culture change as well as the lack of required skills as the most pronounced barriers to realize the full potential of CI/CD. Thus, the DevOps teams cornerstone is a decisive factor for a sustainable CI/CD transformation since it significantly determines the application of CI/CD principles and technical practices.

2. Software Development Processes

Besides the DevOps teams cornerstone, there is the need to establish the right software development processes promoting agile software development techniques to realize full potential of an organization's CI/CD pipeline. Since agile software development is built on fast feedback loops and constant collaboration between different stakeholders, the processes within the CI/CD pipeline need to foster this environment [21]. The process along the CI/CD pipeline typically comprises the steps of plan (setting up the software project with its requirements), code (the stage where developers implement the IT product requirements), build (the stage where the IT artefact is created), test (the stage where code is tested), deploy (the stage where code is deployed to production), and release (the final delivery of the software product to the customer) [2]. A powerful CI/CD pipeline that increases speed and quality of the whole software development process is based on an optimal orchestration and seamless connection of the beforementioned process stages.

A relevant aspect in this context is the shift left approach[6] that leads to a parallelization of process steps within the software development workflow since instead of sending multiple changes to a separate test team, DevOps teams working with CI/CD practices and tools perform continuous automated tests throughout the whole development process [11]. This allows to detect and fix defects early in the process resulting in an improvement of code quality and higher velocity in software delivery [22].

The integration of CI/CD principles related to the cornerstone Software Development Processes leads to an iterative process workflow that offers possibility for continuous testing and feedback and thus to the consequent implementation of the paradigm of agile software development. The streamlined process along the CI/CD pipeline lays

[6] See sub-chapter "CI/CD Explained" for a detailed explanation.

the foundation for teams to collaborate and communicate throughout the CI/CD process workflow to maintain alignment, velocity, and quality [17]. Moreover, teams using a CI/CD pipeline benefit from more freedom in the way they conduct business and collaborate since the process alongside the CI/CD pipeline reduces processual limitations [18].

Although, adapting processes with sight to the CI/CD pipeline offer valuable benefits, this also might come with some costs since changing processes from existing well-established processes to CI/CD-ready processes requires an extensive change effort within the whole organization [18]. In a running organization it would be almost impossible to stop the existing process approach and start completely new from scratch. In addition, it is a costly and time-consuming process to integrate an organization to a CI/CD pipeline since the long-grown technology stack is characterized by a high degree of complexity [18].

3. CI/CD Tools

The process on the CI/CD pipeline is supported by various tools that ensure each process step is seamlessly integrated within the software lifecycle and automated to reach the maximum acceleration potential of the CI/CD pipeline. These tools connect different systems and ensure the reflection of any change in the source code across different environments [21]. Additionally, the cloud native tools of the CI/CD pipeline technically support the agile software development process by accelerating the building of new applications and integration of user feedback and by contributing to dismantling borders between development and operation teams [23]. Each tool used in the CI/CD pipeline might add its special features to the specific process stage. It is important to create interfaces between separate tools to enhance the seamless and automated interaction between them. Moreover, each tool comes with its very own limitations that it brings to the whole process which need to be considered to avoid potential negative effects on the functionality of the CI/CD pipeline [22]. Additionally, too many tools used in the CI/CD pipeline increase complexity which could negatively impact their usability as well as accuracy. These factors might negatively affect the smooth process workflow of the CI/CD pipeline and could therefore decrease quality delivered to the customers. Hence, it is important to find the right balance between the number of tools with their special features for each CI/CD process stage and ensuring manageability of the whole tool chain.

Another important aspect in the context of CI/CD cornerstone CI/CD tools is the question of build or buy. Although, the open-source community is gaining more and more traction nowadays regarding powerful DevOps Tools and the development in cloud-computing enables a shift of the process towards a more powerful environment [21], at Commerzbank, we believe that an organization should assess its strength and weaknesses before choosing a CI/CD toolchain. An off-the-shelf tool or even a pre-made pipeline including all necessary tools might be simpler than building a tool itself, but it might sacrifice flexibility. In our experience, organizations can combine the best of both approaches individually deciding about which tools should be collected from a vendor or if a tool built by the organization itself will more fit the individual organizational requirements. We believe this is more pronounced in the Financial Services Industry since financial institutions have

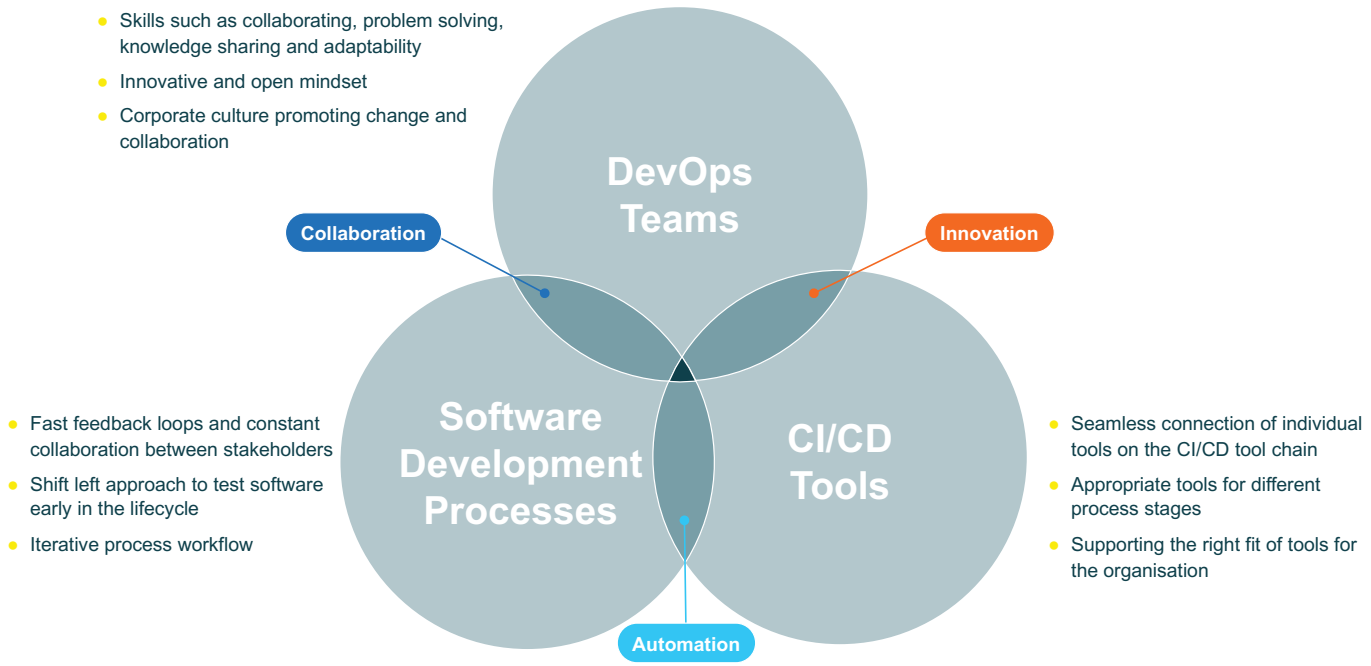


Figure 2: The Three Cornerstones of CI/CD

oftentimes specific requirements due to their long-grown software stack and regulatory parameters that need to be considered. Thus, the possibility that financial institutions can simply leverage the advantages of a pre-made CI/CD pipeline including a ready-to-go tool chain might be illusionary and therefore they usually focus on building their CI/CD pipelines integrating relevant tools on their own.

To summarize, the realization of the whole potential provided by CI/CD pipelines requires a holistic view of this practice. At Commerzbank, we believe that the reflection of the CI/CD components and their interconnectedness in the respective CI/CD endeavors ensures that CI/CD leads to a more collaborative culture, an innovation-driven mindset, and a higher-level of automation. In the following we will present how we are living CI/CD at Commerzbank and in which way we have integrated the CI/CD cornerstones.

How Commerzbank is Living CI/CD

The way in which the DevOps paradigm and the CI/CD concept are applied in the financial services industry is significantly impacted by national and international regulatory requirements. Regulators such as the European Central Bank oversee not only the business model of financial institutions but also impose regulations on their IT Infrastructure and thus the banking specific software development. These regulatory requirements need to be considered when financial institutions apply CI/CD practices and principles, build their CI/CD pipelines as well as set up relevant methods and processes in their DevOps teams.

At Commerzbank, we started with an integrated approach to build our CI/CD pipeline in 2018 as a technology enabler of our agile transformation journey to implement our contribution of being "the bank at your side [7]" . **"What drives us is our commitment to provide a superior customer experience by offering digital, individualized and highly valuable products and services,"** says Dr. Alena Kretzberg (Commerzbank AG). Today, Commerzbank has established the cluster "Continuous Integration/ Continuous Deployment CHAMP" (CI/CD CHAMP) as a technology foundation for the fast and automated delivery of new functionalities to the organization's customers.

The mission of the cluster CI/CD CHAMP is to deliver end-to-end solutions through the CI/CD "Commerzbank Hyper Acceleration Master Pipeline" (CHAMP) for all other clusters within Commerzbank. **"Our objective is to significantly accelerate the development of new digital and customized solutions for our private and corporate customers,"** says Damian Waszak (Commerzbank AG). CI/CD CHAMP is our approach at Commerzbank to realize CI/CD and DevOps. As with every acronym each word has certain meaning. Let us take a moment to explain it:

- First of all, it is **Commerzbank,** it is our way of implementing CI/CD and DevOps. We are not claiming that this is how everyone should do it. But we think this is the best way how it works for us considering Commerzbank's history and objectives.

- Then there is **Hyper Acceleration:** We want to support DevOps teams to speed up, being able to deliver more and more financial features to Commerzbank's private and corporate customers. As this objective should be ambitious, at Commerzbank, we think hyper acceleration is key. Releases should happen daily, ideally on-demand, whenever a new feature is ready. Only with this approach we can realize our goals - deliver delightful software products to customers and get ahead of competition.

- Next, the **Pipeline:** It is mostly a technical solution that executes CI/CD at Commerzbank and enables teams to deliver to production with high quality. We provide various tools in each stage of the software development process - from integrated development environments (IDE) to deployment tools. Since all tools are seamlessly integrated in one pipeline, synergy effects are achieved. On top of this, at Commerzbank we offer coaching solutions - training teams in efficiently using the tools and methods. This is complemented by a set of internal policies that govern the software development process, implemented as a set of documents and tools that guide users in fulfilling regulatory requirements.

- Lastly, a **Master** (Pipeline), with all the breadth of tools, platforms, target environments, middleware components, programming languages, frameworks and high national and international regulatory requirements coming at Commerzbank, we think that it is crucial that there is one single master pipeline that fulfils all those requirements. This pipeline leads to a high level of confidence since everything runs smoothly if well orchestrated.

Since CI/CD CHAMP is focused on the acceleration of Commerzbank's delivery organization, we prefer to employ the analogy of a racetrack in Formula E. As in Formula E, a successful CI/CD pipeline is about speed, engineering discipline and outstanding teamwork to facilitate the complete automation of the delivery process from the initial requirement to delivering the final product to our customers. As seen in figure 3, CI/CD CHAMP is composed of three interrelated services: (1) **Frameworks** assuring the compliance with regulatory requirements during the whole software development process; (2) **Automated Pipeline** comprising all technical steps and tools from development over testing to production, and (3) **DevOps Coaching** promoting state-of-the-art development methods and practices within the cross-functional agile teams.

It is important to mention that the interplay between the three CI/CD CHAMP services makes our CI/CD approach so unique. **"We as Commerzbank offer the CI/CD CHAMP automated pipeline with its tools and integration as a standard for all Clusters in the Delivery Organization so that DevOps teams working on new software features can flexibly configure their own pipeline template based on their individual needs,"** says Fonlinda Frasheri (Commerzbank AG). It is the aspiration of the CI/CD CHAMP team to provide a CI/CD pipeline that allows for the necessary flexibility to be adapted to diverse IT application architectures. Additionally, the CI/CD CHAMP team gives technical and methodological support and guidance during the configuration and final working phase with the CI/CD pipeline at any time.

To provide you with more insights about how Commerzbank lives CI/CD on a daily basis, we would like to explain the three services of CI/CD CHAMP in more detail:

[7] Our proven brand promise "Commerzbank. The Bank at your side" puts our brand positioning in a nutshell, describes what we stand for and what we promise to our customers.

Frameworks

Tool supported end-to-end documentation process to comply with regulatory requirements during the whole software process.

Automated Pipeline

Fully automated CI/CD pipeline covering all technical steps from development to testing and production.

DevOps Coaching

State-of-the-art development practices and coaching to further strengthen DevOps technical and mindset excellence.

Figure 3: Services of Commerzbank Hyper Acceleration Master Pipeline (CHAMP)

- **CI/CD CHAMP Frameworks:** As in a race situation having the necessary safety equipment, for instance a racing helmet or racing suit, is crucial when leveraging the maximum speed of your racing car while knowing that all necessary conditions are met to bring you safely to the finish line. Like in our racing example, the service "Frameworks" provides our DevOps teams with a tool-supported end-to-end documentation process that covers the entire IT product development workflow to ensure compliance with regulatory standards while ensuring our DevOps teams can fully focus on developing the right software for our customers. This added value is achieved by two interconnected solutions: the Commerzbank Delivery Guide (CDG) that contains a description and relevant documentation requirements for each step in the software development process and a Product Owner Cockpit (PO-Cockpit CHAMP) that guides the Product Owner (PO [8]) and other involved colleagues through the development workflow and necessary documentation according to the CDG (as seen in figure 4).

The PO-Cockpit serves as the regulatory single point of truth for any audits and controls and provides an overview of all CI/CD endeavors. Since the required documentation depends on the planned software solution, the PO-Cockpit allows employees to tailor the documentation workflow according to the software artefact type. Additionally, the tool offers interfaces to other documentation systems ensuring that DevOps teams can navigate through the workflow using one single tool. The PO-Cockpit allows users to tailor the required documentation to their respective software development endeavor using a standardized set of questions to automatically generate the needed documentation template. Another feature is the embedded delta documentation function that allows to use existing documentation templates of the completed projects if the respective users want to document a change or new feature of the existing endeavor. Moreover, the PO-Cockpit ensures that the documentation contains all necessary approvals. For instance, the tool automatically sends the documentation template to the respective responsible person to be compliant with the double check principle ensuring security standards. Additionally, the technical illustration of the CDG in the PO-Cockpit serves as a guiding tool throughout

[8] For a detailed explanation refer to "Glossary", page 36.

all process steps of the software development workflow while facilitating the management of documentation requirements through transparent navigation. DevOps teams always have an overview of the next process step and how they can find the right balance between accelerating the software development process and documentation.

- **CI/CD CHAMP Automated Pipeline:** At the heart of CI/CD CHAMP is our automated pipeline, the racing car, which turns power into motion, resulting in a significant acceleration of Commerzbank's software delivery process. The objective of the automated pipeline is to support DevOps teams in their CI/CD endeavors with the necessary technical tools and consulting on how these tools can be applied to implement their envisioned products or features to enhance the experience of our private and corporate customers. On the technical tool side, the automated pipeline offers different tools currently comprising of up to 15 different applications with partially automated interfaces along the CI/CD software delivery workflow from the initial requirement to production.

As seen in figure 5, the technical tools support DevOps teams to successfully navigate on the CI/CD CHAMP racetrack completing all development stages towards the finish line of deploying the final product to production. The first six stages comprise tools used to realize continuous integration:

1. "User Story Design" which provides POs, using agile software development methods, such as Scrum or Kanban [9], with an automatically created tool space to document the requirements for their product visions in the form of user stories.

2. "Coding", provides standardized functions, libraries, frameworks, and integrated development environments to developers to implement the user stories.

3. "Source Code Control" provides a central source code repository, where changes from all software developers can be stored, merged, and reviewed. Each code change is tracked to ensure traceability.

4. "Build" tools enable an automated build process with each commit, such as change in source code repository. During the build process step, automated unit tests, functional tests, and code analysis checks are executed ensuring that the code is of high quality before being integrated.

5. "Code Analysis" represents another quality gate for the code artefact created in "Build" comprising an automated checkup of the code for programming errors and adherence to internal policies, e.g., programming guidelines. It is used to detect outdated, insecure, or error-prone dependencies used in the software project. In addition, security scans are conducted to detect vulnerabilities.

6. In the step "Delivery", the code artefact is automatically handed over to the artefact repository and archived securely.

The next steps on the CI/CD CHAMP racetrack are part of continuous deployment:

7. "Configuration Management" represents a solution for the configuration and management of the stage specific configuration parameters.

8. "Deploy Test" provides tools to support DevOps teams with test quality criteria (e.g., functionality) of artefacts in a target environment with a low number of manual tests and a central overview of all test cases. These tools enable a full automation of the deployment process.

Figure 4: Solutions Provided by CI/CD CHAMP Frameworks

[9] For a detailed explanation refer to "Glossary", page 36.

Figure 5: Steps of the CI/CD CHAMP Automated Pipeline

9. "Deploy PROD" provides the appropriate tools for an automated release of the artefact in the target system.

There are two conditions that needed to be considered for the design of the automated pipeline. First, the regulatory requirement of "Segregation of Duties" (SoD), that requires that the person changing the code of an application cannot bring the same change live, effectively separating the functions of development and operations, influenced the way the deployment process was automated. At Commerzbank, we established safeguard measures to ensure a consisting SoD throughout the entire deployment process. Second, additional complexity of deployments arose from the extent of middleware, programming languages, and cloud technologies used at Commerzbank. To ensure CI/CD CHAMP is as open and extensible as possible, the CI/CD CHAMP Automated Pipeline can deploy to all of those target systems. Additionally, the user experience for developers and operators is the same, no matter what the underlying target is.

The automated pipeline offers several advantages to further accelerate Commerzbank's CI/CD activities. DevOps teams using the CI/CD CHAMP pipeline are provided with reusable templates for all included tools that can be tailored to the specific project needs leading to a simplification of the whole software delivery process. Another area in which DevOps teams benefit from CI/CD CHAMP is the self-service aspect of the pipeline. Traditionally, DevOps teams that would like to use multiple CI/CD tools, would have to talk to multiple internal teams that manage those tools leading to time lags in granting access and configurating user rights. Therefore, at the core of the CI/CD CHAMP pipeline is a self-service platform where all required tools can be requested on demand, and DevOps teams can get access to them without delay which increases the overall satisfaction of DevOps teams that are more in control of their software development environment.

- **CI/CD CHAMP DevOps Coaching:** Besides the technical factors, the success of a racing team also depends on the skills and execution of the driver. A successful racing driver who leverages the full potential of his or her car, needs an experienced pit crew as well as a

team of racing strategists which support him or her off the racetrack on the pit lane. We think that also applies to our CI/CD pipeline since a successful application of all technical features requires the best possible methodological and cultural support of DevOps teams. Therefore, the CI/CD CHAMP service "DevOps Coaching" offers training and coaching for modern software development methods to further establish software craftsmanship within Commerzbank's cross-functional agile teams. The training and coaching solutions include a two-stage procedure comprising "DevOps Technical Excellence" to further strengthen agile techniques and practices within the DevOps teams and "DevOps Agile Empowerment" to support DevOps teams in applying and living the DevOps concept. Since the successful execution of DevOps depends both on technical excellence and a collaborative and inclusive mindset, the focus of DevOps Coaching is on offering training and coaching solutions comprising these two interrelated components as seen in figure 6.

On the methodological side, the component DevOps Technical Excellence provides trainings comprising state-of-the art methodological strategies (e.g., Value Stream Mapping), software development practices (e.g., Test Driven Development, Continuous Integration, Clean Code) and advanced testing techniques (e.g., Behavior Driven Development, Integration Testing). On the other side, these methodological trainings are complemented by DevOps Agile Empowerment coaching in which a DevOps coach accompanies DevOps teams beyond the respective training sessions to support them in adopting the practices in their daily business. Additionally, the coaching sessions aim at further strengthen the DevOps mindset of agile teams. The selection of the appropriate training and coaching sections is based on the individual need of the respective agile team as well as existing challenges they face. Therefore, CI/CD CHAMP DevOps Coaching is highly individualized to support agile teams in the best possible way in their daily business. The DevOps teams benefit from this approach in several ways. First, they can leverage new practices and skills to further increase their software quality through a shift left approach. Second, they are optimally prepared to use our CI/CD CHAMP pipeline faster and easier to implement their intended solutions and features.

"We believe that our CI/CD CHAMP pipeline offers a good practice of how organizations can holistically design their CI/CD pipelines leveraging the CI/CD cornerstones of DevOps Team, Software Development Process, and CI/CD Tools. Although, we successfully integrated CHAMP in our daily business, we are still working with high intensity to bring it to the next level," says Florian Meiser (Commerzbank AG). We are driven by the feedback of the DevOps teams and Commerzbank's customers – because our software is just as good as they think it is. Like teams in the Formula E are constantly searching for improvement, we are working rigidly to further innovate CI/CD CHAMP and its services. Please have a look at chapter 4 where we share our new features and vision for CHAMP 2.0 with you.

Figure 6: Components of Commerzbank's CI/CD CHAMP DevOps Coaching

CI/CD CHAMP Use Cases

At Commerzbank, we would like to give you a deeper insight into how CI/CD CHAMP and its three interrelated services support the teams of our Delivery Organization in their daily business. Hence, we are proud to share with you three of our CI/CD CHAMP use cases as depicted in figure 7 to explain the implementation of the different CI/CD CHAMP solutions and how our CI/CD CHAMP team collaborates with their internal customers. In addition, we want to share our learnings and experiences on our CI/CD journey so far. Thus, we provide you at the end of the chapter with our best practices on how to master the implementation of CI/CD in a large corporation. Further, we are looking forward to opening up the discussion in the CI/CD and DevOps community to also hear your learnings and best practices.

CI/CD CHAMP Use Case 1: Commerzbank API Banking

Commerzbank's Application Programming Interfaces (API) Banking cluster enables co-creation of valuable solutions with business partners, customers and FinTechs. In essence APIs are technologies that enable communication between IT-systems resulting in a closer collaboration among different parties resulting in better solutions for our customers [24]. The API Banking cluster is responsible for providing a platform for managing and operating the APIs of Commerzbank and was in 2018 one of the first cases where CI/CD CHAMP was used to enable a faster software development process while further increasing the software code quality, both factors being crucial for delivering APIs.

CI/CD CHAMP supports the cluster API Banking by creating an automated pipeline in close collaboration, providing the appropriate technical tools and consulting, as well as by offering DevOps coaching comprising methodological and mindset training components.

CI/CD CHAMP Automated Pipeline

Initially the API deployment processes were carried out manually which led to challenges regarding a prompt delivery of the API and scalability. Complexity arose from the fact that a relatively small team had to deal with a plethora of various APIs provided daily by all other software teams within Commerzbank. Thus, the API Banking unit partnered with the automated pipeline team of CI/CD CHAMP to further improve the API deployment process. To identify potential for improvement, the API deployment process was analyzed in detail according to the methodology of Value Stream Mapping (VSM [10]) . The design of the target API deployment process allowed the team to derive milestones that needed to be achieved to improve the initial process. The team of CI/CD CHAMP was responsible for restructuring the API deployment process to enable process speed, scalability, and security. The resulting process artefacts were delivered to the API Banking cluster that adjusted these artefacts accordingly. Although, the DevOps team from API Banking and CI/CD CHAMP had separate tasks to adjust the API deployment process, they both worked hand in hand. **"The collaboration with the colleagues from CI/CD CHAMP was excellent during the whole process. Our weekly meetings and the possibility to always contact one of the colleagues highly contributed to the success of our project,"** says Volker Sulzbach (Commerzbank AG).

During the implementation of the target API deployment process the joint team also faced some challenges. For instance, the tasks related to the adjustment of the API deployment process were initially seen as additional effort for the API Banking unit. However, the clear identification of milestones and the incremental delivery of improvements in the deployment process, continuously enabled the team to speed up deployment, which in turn did free up time for other tasks and ultimately lead to the mitigation of the initial problem.

Additionally, API Banking was using a test automation tool, that was not included in the setting of Commerzbank and hence not part of the standardized, automated CI/CD CHAMP pipeline. Accordingly, even though the deployment time could be reduced significantly, the high quality of the releases needed to be ensured. To solve this issue, the CI/CD CHAMP team implemented a new interface between the automated pipeline and the tool API Banking already used for integration tests and demonstrated that the automated pipeline can be extended and individually configured based on the needs of the users. Through this interface the automated CI/CD CHAMP pipeline triggers test automation and reads result reports, ensuring only high-quality releases are deployed to the production environment.

The cluster API Banking benefitted from the automated CI/CD CHAMP pipeline in different ways. **"The automated API deployment process is a success for our Cluster. Deploying an API has become much easier and faster compared to the initial process,"** says Christian Pfaff (Commerzbank AG). As a result of adjusting the API deployment process, the time needed to deploy an API in production was reduced to 15 to 20 minutes assuming a well-designed API. The greatest benefit from this optimization comes from the frequent re-deployments that are necessary for the usual minor changes. The high level of automation not only

[10] For a detailed explanation refer to "Glossary", page 36.

CI/CD Use Cases	Highlights	Applied CI/CD CHAMP Services: Frameworks	Automated Pipeline	DevOps Coaching
API Banking	Accelerating the API development process and further improving agile working methods	◑	●	●
Big Data & Advanced Analytics	Onboarding of a Google-Cloud use case on CI/CD CHAMP to enable an on-premise development process for new products	◕	●	◑
CCB Security Frontend	Enhancing release flexibility of new products as well as enabling earlier testing and faster feedback	◕	●	○

● CI/CD CHAMP service completely applied ◕ CI/CD CHAMP service almost completely applied ◑ CI/CD CHAMP service partly applied ○ No CI/CD CHAMP service applied

Figure 7: Overview Commerzbank CI/CD CHAMP Use Cases

provided scalability but also a high software quality since the reduction of manual steps makes the whole setting less prone to errors. The new process also increased the usability of the API mid-tier application deployment process since developers can automatically create new pipelines for API applications via a self-service tool. The team around CI/CD CHAMP also took some key learnings for future projects from working together with the API Banking cluster. For instance, this experience contributed to the further standardization of the CI/CD pipeline tools ensuring the flexibility to integrate additional tools if necessary. Additionally, the regular meetings with the API Banking team during the project contributed to the practice of conducting weekly checkpoints and having a single overarching product backlog for the whole project.

CI/CD CHAMP DevOps-Coaching

The API Banking team aimed to further develop their agile skills and challenge their current agile working practices to identify areas for further improvement. Hence, they participated in a three-months coaching program offered by CI/CD CHAMP DevOps Coaching to learn how they could sustain efficiency and quality in their daily business and how to benefit from additional agile software development methodologies.

Prior to the coaching program, the cluster API Banking faced two main challenges they wanted to address. First, the API team is dependent on the deliveries of a large set of different stakeholders. For instance, it is possible that the priority of user stories changes within an ongoing sprint if a software artefact provided by a third party is delayed. Second, the team is very heterogenous in terms of skills and software engineering background.

The CI/CD DevOps Coaching program started with an analysis workshop assessing the current agile team maturity and the aspects the team wanted to improve. In a next step, the team took part in full-time training sessions covering relevant agile software development practices based on the results of the prior analysis and assessment session. **"The full-time training sessions gave us the opportunity to reflect and focus on specific areas of our daily work with an outside-in perspective. As a result, we received valuable feedback about which additional agile software development practices could support us,"** says Akli Amar-Youcef (Commerzbank AG) since the first week of the program aims at educating the team on different agile development practices and offering them the possibility to challenge their agile working approach. At the end of the first week, the team identified two focus topics they wanted to intensify in the following full-time coaching: Value Stream Mapping (VSM) as a more holistic practice and Test-Driven Development (TDD) as an agile technical practice related to the coding process. Both selected agile practices were then applied to an existing user story during and after the two weeks full-time coaching session.

The team applied VSM for analyzing existing processes and thus set the basis to further increase its efficiency and productivity in delivering new ideas and solutions to customers. In applying TDD in the software coding process, the team was also able to improve programming habits that are valuable for agile software development. Hence, the team benefitted mainly from two effects. First, since the team members worked together on the software artefact, they got to know each other and their individual strengths better. This contributed to a higher level of team spirit and a better understanding of how the team members could support each other in the best way having different skill sets and levels of experience. Second, the team practiced with an experienced coach providing them with individual guidance and an overview of possibilities. Following the two weeks full-time coaching sessions, the CI/CD CHAMP DevOps coaches offered further support according to the pull principle based on the requests of the API team.

Undoubtedly, the two weeks full-time training and coaching sessions were an investment since it interrupted the normal working process. But for the API team it has proven a fruitful investment enabling the team to perform even better than before or as Arnd Bischoff (Commerzbank AG) shared: **"I really appreciated that the full-time training**

session was tailored to our specific interests and needs. The CI/CD CHAMP DevOps Coaching definitely further improved our team motivation and collaboration – and it was a lot of fun as well." The CI/CD CHAMP DevOps Coaching contributed to the agile working practices and collaboration of the API team. After the program the team applied new agile practices in their daily software development, they recognized efficiency gains and productivity improvements as well as growing closer as a team. Additionally, the CI/CD CHAMP DevOps Coaching team also gained valuable learnings trough this experience working together with the team at API Banking. For instance, they learned that tailoring the training and coaching sessions to the individual team situation is crucial and thus the time for preparation and assessment was extended for future internal customers.

CI/CD CHAMP Use Case 2: Big Data & Advanced Analytics

Within the cluster Big Data & Advanced Analytics (BDAA) new products are developed applying analytics to internal and external data. In 2020, the cluster started to develop a talkbot, which is in essence a chatbot utilizing voice-capabilities to assist Commerzbank`s employees with their inquiries via the Commerzbank internal service hotline. The talkbot enables automated processing of our colleagues' service requests that otherwise needed to be carried out manually by agents from the internal service center. Although the BDAA talkbot case is still ongoing, it demonstrates the integration and tool flexibility of the CI/CD CHAMP Automated Pipeline.

Although, the BDAA cluster has been using CI/CD CHAMP in their software development endeavors since 2018, BDAA was first built utilizing Google Cloud capabilities to take advantage of the cloud infrastructure. However, the Cluster faced some challenges with the deployment process if the deployment of a new software artefact required a deviation from the standard deployment process. In these cases, the BDAA team faced a high complexity and manual rework that negatively impacted the time needed for deployment and always demanded specialist knowledge for finding solutions. Thus, they contacted the CI/CD CHAMP pipeline team to support them in automating the deployment process.

At the beginning of the project, the CI/CD CHAMP team carefully assessed the BDAA team's requirements and current approach of the software deployment process. The BDAA talkbot case is the first Google Cloud native use case for CI/CD CHAMP since CI/CD CHAMP offered only the possibility to automate the software deployment process on-premise at this time. So, the mission was to enable the first use case to be deployed in the Google Cloud using CI/CD CHAMP and to establish a software deployment process that was stable and easy to use. For instance, they need to deploy on demand due to the complexity of the applications used.

The team jointly decided to work together in a mixed setting to ensure that the requirements and learnings made of BDAA team members in working in the cloud environment are reflected in the future deployment process. **"The colleagues from CI/CD CHAMP Automated Pipeline always carefully considered requirements on the final deployment process which really supported the solution we needed,"** says Michael Bellinger (Commerzbank AG). Usually, the onboarding and the maintenance specialists of CI/CD CHAMP Automated Pipeline are working separately from the developers of the respective customer teams, however, the BDAA case required the expertise from both teams. Thus, there was one colleague from CI/CD CHAMP Automated Pipeline as central contact person for BDAA. When further expertise was needed additional experts were included. In general, the procedure of only using a reasonably small team and adding expertise on a situational level has proven best practice to quickly identify challenges and find the required solutions. Together the joint team onboarded the BDAA talkbot on the automated CI/CD CHAMP pipeline and already set the preconditions necessary in the cloud environment to be able to deploy on demand. Further, the BDAA team used the PO-Cockpit provided by CI/CD CHAMP Frameworks to document their software development process. The PO-Cockpit guided the PO through the whole process and supported the overall goal to build a process that does not demand special knowledge. **"We are now able to easily develop and deploy new software products on demand thanks to the support of our CI/CD CHAMP colleagues,"** says Peggy Schuchert (Commerzbank AG). Additionally, the BDAA team built a holistic partnership with the team from CI/CD CHAMP DevOps Coaching to methodologically support the whole technical process and further strengthen their agile software development practices. This way, methodological and mindset support was provided to BDAA to establish software craftsmanship and thereby facilitate the use of CI/CD.

In summary, the process of offering the automated CI/CD pipeline in a cloud environment is still ongoing but the joint team is sure that their journey will allow other cloud use cases to be onboarded easily and allow them to use the on-premise process for the development of new products. Applying a paradigm of shifting more use cases to cloud environments the pioneering use case of the BDAA talkbot paved the way for other use cases to be developed using the on-premise automated CI/CD pipeline of Commerzbank. In addition, CI/CD CHAMP also took two other learnings from this use case. First, new approaches such as shifting a Google Cloud use case on the automated CI/CD pipeline requires an analysis of the initial situation as well as understanding of the requirements that need to be considered in the process. Second, in cases in which uncertainty is high and the project outcome is highly dependent on external factors, a joint team working in short iteration cycles can help to mitigate potential risks. Additionally, the BDAA team provided valuable feedback to the CI/CD CHAMP PO-Cockpit which laid the foundation for further adaptions to increase the flexibility in its tailoring feature. We are looking forward to jointly work together with the team of BDAA to have them finally onboarded on the CI/CD CHAMP pipeline.

CI/CD CHAMP Use Case 3: CCB Securities Frontend

The cluster CCB Securities Frontend (CCB stands for Cross Channel Banking) develops online solutions for Commerzbank customers to manage their portfolios and to carry out their stock exchange transactions within their online banking application. For instance, Commerzbank customers can easily buy and sell stocks on major exchanges independently, anytime and anywhere via their smartphone or PC.

Prior to their application of CI/CD and their onboarding on the automated CI/CD CHAMP pipeline, the team of CCB Securities Frontend was highly dependent on the technology stack of the CCB Online Banking cluster since their Securities Frontend application was one of many applications embedded in the CCB application. This led to a high complexity within the deployment process and CCB Securities Frontend was only able to release once a quarter since deployment and test cycles need to be coordinated among the whole CCB application. Thus, the CCB Securities Frontend team aimed to have shorter deployment cycles independently from the IT operation team of the CCB main application leading to shorter feedback loops concerning their new software features. CCB Securities Frontend also wanted to use cloud-based technologies and decided to leverage a cloud-based Container Platform that was set up in Commerzbank at that time.

The cluster aimed to separate its deployment process from the CCB application to gain speed and flexibility and thus got in touch with the CI/CD CHAMP team. After the team of CCB Securities Frontend was onboarded on the CI/CD CHAMP pipeline, they both worked together to provide the CCB Securities Frontend, with capabilities to deploy software to the cloud-based container platform. As with previous use cases, CI/CD CHAMP was extended to allow for this. The CCB Securities Frontend team also applied the PO-Cockpit provided by CI/CD CHAMP Frameworks. The PO-Cockpit allowed the team to easily structure the documentation process and centrally coordinate the single process steps during the software development process. **As Claus Weber (Commerzbank AG). notes "The PO-Cockpit supports the documentation process by offering a simple tailoring to the required documentation. I really appreciate that it guides product owners throughout the documentation process in one single application."**

After the team was onboarded on the CI/CD CHAMP Automated Pipeline, they are now able to autonomously release new software solutions or features on demand due to the high level of automation the CI/CD CHAMP Automated Pipeline provides. Additionally, the team can detect, analyze, and correct defaults faster. **"We are very satisfied with the tools and process provided by CI/CD CHAMP Automated Pipeline. The combination of automation, early testing and continuous support from the CI/CD CHAMP colleagues makes the difference," says Klaus Donath (Commerzbank AG).**

In summary, the CCB Securities Frontend use case demonstrates the possibility of separating and onboarding an existing software application from the technology stack of Commerzbank to CI/CD CHAMP. Additionally, this use case is also an example of how teams use CI/CD CHAMP as a self-service tool for developing new software applications or features. Moreover, the CCB Securities Frontend use case served as blueprint how CCB applications might be separated from the monolithic legacy technology stack, thus provided the impetus for the practice of breaking down the technology stack into containerized applications. This case also provided important learnings for the CI/CD CHAMP team. For instance, the onboarding process to the pipeline was further improved and is now conducted automatically enabling DevOps teams from the delivery organization to directly create new projects on the pipeline. Moreover, this feedback of the DevOps teams concerning the PO-Cockpit was used to further enhance the tailoring function of the tool as well as the possibility to build on existing documentation in the case of a software change.

Commerzbank's Best Practices – How to Master CI/CD

In the last four years, we learned that a one-size-fits all approach to successfully apply and live CI/CD in the organization does not exist. Instead, success is highly depen-

dent on the interplay between different factors and how the technical and mindset components of this paradigm are responsive to demanding business needs. Considering our analogy of the Formula E, success on the racetrack depends on the optimal adaption of the racing car to the driving conditions. We as Commerzbank would like to share with you the four most important best practices from our CI/CD experience that might also support you in your next CI/CD endeavors.

1. Following a self-service approach as a guiding principle
The first principle we followed when we designed and developed the automated CI/CD CHAMP pipeline further is the principle of self-service that enhances scalability. The pipeline should be easily accessible and usable for every DevOps team member. For that reason, we created a portal as the gate to our pipeline where a new team member can order and get access to all the tools within one day. We further complement the self-service function by user friendly documentation for each of the tools, templates (often provided as code in Git Repository) which can be used for a quick start and different samples in a playground to intensify learning and collaboration. **"Building the surrounding fostering these components requires a strategy and roadmap that articulates clear goals and measures of how employees can be encouraged to collaborate,"** says Dr. Lars Friedrich (Commerzbank AG). Another important aspect to consider is end-to-end responsibility to empower DevOps teams and allow for team autonomy. DevOps teams need flexibility to define their own workflows and to configure their own set of rules in certain tools without risking noncompliance. The CI/CD CHAMP vision is to accelerate the software delivery with a standard pipeline able to fulfill all requirements that teams could have while developing and delivering software. Although, diversity among team members and autonomy within DevOps teams paired with a standard pipeline seems contradictory, it is important to find the right balance between flexibility and control. Therefore, the automated CI/CD CHAMP pipeline incorporates all necessary governance rules and guidelines of Commerzbank to ensure compliance and at the same time provide teams the possibility to implement their own rules and templates without losing speed and autonomy.

2. Aiming for an expandable, flexible, and integrated pipeline
A new feature in the hands of a customer always starts with an idea. Therefore, we have designed CI/CD CHAMP as a highly integrated pipeline to enable full automation in the software development process starting with a user story and completing the cycle with an automated deployment into production. All the tools in the pipeline are connected via APIs or have pre-configured plug-ins which build direct interfaces between them. New components and tools can be easily added to the existing pipeline based on the needs of the respective DevOps team. At Commerzbank, we believe an expandable pipeline is an appropriate approach to reach different target environments in a broader range from mainframe to cloud. A standard pipeline should be technology stack agnostic and at the same time provide the possibility to treat infrastructure or code in the same way as application code. To successfully live DevOps it is crucial to provide the same tools and practices for the functions of development and operations. The automated CI/CD CHAMP pipeline is regarded as the common path application when software artefacts are built, tested, quality checked and deployed. Beside scalability another important leverage for accelerating software delivery is the ability to provide infrastructure and deploy code dynamically.

3. Focusing on measuring and visualizing
The basis of any continuous improvement should be facts and figures. To obtain these, it is necessary to monitor the complete value stream during a CI/CD cycle. The automated CI/CD CHAMP pipeline allows for easy measuring and tracking relevant Key Performance Indicators (KPI) for each step of the software development process. The gathered data is then visualized in a dashboard to guarantee full transparency about the efficiency of the process and indicate areas for optimization. The dashboard contains all the relevant DevOps KPI, such as deployment frequency, deployment speed or deployment failure. This allows DevOps teams to improve in the next iteration. Besides that, at Commerzbank, we continuously observe the usage of the automated CI/CD CHAMP Pipeline so that we can inspect and improve certain building blocks or services of CI/CD CHAMP.

4. Applying Shift Left
Our experience shows that DevOps teams that start early with using the automated CI/CD CHAMP pipeline gain more speed, efficiency, and confidence in their work. They can react early and quickly to quality issues and fix them to finally improve their software artefacts without compromising an accelerated time-to-market. Therefore, the CI/CD CHAMP team consult, coach and support DevOps teams in the onboarding process on the pipeline in all environments so that they can start early realizing their ideas and delivering faster high-quality software. At Commerzbank, we advise to shift left as much as possible, start to automate tests and think in iterations which will result in delivering small pieces of software more frequently.

5. Continuously improving through collaboration with CHAMP users
CI/CD CHAMP is a technology foundation that provides its services to all DevOps teams in Commerzbank's Delivery Organization. From that perspective, CI/CD CHAMP users are the ones who influence the further development of the pipeline. Therefore, a potential extension of the pipeline is always done in cooperation with the users. They know best what they need to have a better user experience and decide which new features are essential. Besides the importance of gaining valuable impulses from users, the CI/CD CHAMP team itself drives innovation to create new technologies and possibilities which are subsequently standardized for the users. CI/CD CHAMP is open to incorporate new innovative ideas coming from inside and outside Commerzbank. The generation of internal ideas is ensured by internal cooperation, while external ideas are created through interactions and exchanges with vendors, different proof of concepts and conference participations.

The three exemplary use cases highlight how Commerzbank leverages CI/CD capabilities to be a strong partner for customers delivering valuable solutions that meet highest quality standards. We hope that these use cases contributed to a better understanding of CI/CD CHAMP and that our best practices might benefit you in your next CI/CD endeavor. In the next chapter we want to share our expectations about the evolution of agile software development.

Hypotheses About the Future of Agile Software Development until 2026

When we think of where the future of agile software development is heading to, we need to understand the dynamics that might shape its evolution. Now that DevOps practices have achieved widespread adoption in agile software development [18], future developments that go beyond bringing Dev and Ops closer together do exist. So, the question is: in which direction will agile software development evolve within the next five years and how will new technologies affect it? Together with 12 experts from different industries we discussed trends and predictions to weigh in on what to expect in the upcoming five years. As a result, we formulated five hypotheses about the future of agile software development until 2026 as depicted in figure 8. These hypotheses center on a few key themes: as organizations accelerate on their agile transformation journey, focusing on developing agile capabilities and embracing agile values will be the imperative; leveraging technologies will continue to offer new possibilities for automation and ultimately provide the basis for improved collaboration; and organizations will further focus on an Everything as Code approach to create a common language for the whole organization.

01 Everything as Code (EaC) will become a priority leading to the practice that everything will be adjusted and managed using code

An EaC approach treats all components of a system such as documentation, application code, configuration management, infrastructure, and compliance as code so that everything adheres to the same software development practices [25]. As focusing on the concrete CI/CD use case, the idea of EaC will make its entry across each layer of the CI/CD pipeline so that infrastructure, schemas, pipelines, and operations are all described and treated like application code [26]. This means that automation via software will be everywhere providing the technical foundation for an array of other developments such as enhanced cross-functional collaboration, or a shift left approach of security and compliance that can be expected to occur in the next five years.

At its core, EaC is not at all a new trend. Some isolated forms of EaC have already gained popularity within the last years, Infrastructure as Code (IaC), Pipeline as Code (PaC) and GitOps [11] being the most prominent examples and it can be expected that this development will pick up steam due to the convergence of three interrelated factors [27] [28]. First, an increasing number of tool vendors take an EaC-first approach to tool configuration and deployment by offering organizations the possibility to manage everything using code files [28]. Second, the ongoing standardization of configuration formats also supported the EaC trend since a common format makes it easier for developers to manage all of their tools with the same language and methodology (Tozzi 2020). Third, the possibility to build and manage an entire tool chain on a CI/CD pipeline, applying EaC drives the overall EaC approach also for other tools and processes [28]. Thus, EaC offers not only a highly repeatable and scalable approach to tasks, but also increases consistency across IT systems and processes [27].

The reason why EaC will become a priority is mainly based on its various potential benefits. For instance, EaC will ensure transparency and a clear understanding of the infrastructure, configurations and policies without the need to rely on manually updated documentation as the code does evolve to a single point of truth incorporating all information. EaC will provide DevOps teams with the technical foundation for their daily collaboration, helping them to organize their work more efficiently [28]. Moreover, Compliance as Code (CaC) and Security as Code (SaC) ensure that compliance and security requirements and checks are built as code into the software development life cycle from the first stages of the process. This creates a common standard for all DevOps teams and guarantees that compliance and security issues are automatically detected and corrected, in near real time [29]. Additionally, if compliance or security is written into software applications, they can also be continuously updated in the event of changing regulations without the need to manually adapt applications to new regulations. EaC also offers the possibility to establish documentation as a continuous automated process output throughout the whole CI/CD process. In future, developers will be supported with documentation that is automatically generated via a set of templates that collect and transform the required information from its origin to the target documents for a simultaneous delivery of the new code and documentation [30]. This automated shift left of documentation will largely guarantee the accuracy and integrity of documentation and will significantly increase the delivery speed of the documentation process [30]. Moreover, IaC offers organizations the possibility to

[11] GitOps refers to the approach to use Git to manage IT Operations.

01 Everything as Code (EaC) will become a priority leading to the practice that everything will be adjusted and managed using code

02 "BizDevSecOps" will be the evolution of DevOps ensuring a holistic cross-functional team approach in delivering customer value

03 Organizations will leverage Machine Learning (ML) algorithms to automate a wide range of processes in the DevOps lifecycle

04 End-to-End autonomous testing will be the standard approach for software testing leveraging cloud-based services and anonymous test data extracted from production

05 Organizations will focus more on developing agile capabilities and following agile values to comprehensively "being agile" instead of just "doing agile"

Figure 8: Hypotheses About the Future of Agile Software Development until 2026

deploy infrastructure whenever it's required and to integrate it into a CI/CD pipeline to dynamically build and destroy different environments as the pipeline executes. Finally, EaC ensures consistency in migrations, deployments, and configurations. Deployments in a specific environment will result in the same configuration as if the same deployment was conducted in another environment. Additionally, if organizations follow the practice of EaC, the efficiency and performance of an organization to react to customer needs will be measured and tracked by CI/CD value streams.

While this trend offers many potential benefits, it is important to consider that the pace of this development is highly dependent on the flexibility of vendors concerning code and installation playbooks as well as the existence of tools to verify the code files to avoid potential pitfalls. In addition, organizations will need to create the necessary conditions to take full advantage of EaC. As Daniel Meixner (Microsoft AG) put it, **"EaC will certainly become an important trend. However, organizations will need to meet a number of prerequisites to be prepared to take full advantage of EaC. For instance, appropriate tooling will be necessary and access rights sensitive areas will have to be installed and managed."** Additionally, EaC requires a mindset shift so that all team members irrespective of their function create a deep understanding for each other's activities and tasks. For example, team members of operations should adopt a development mentality and automate whatever and wherever possible. Moreover, organizations need to ensure the appropriate working conditions so that experts from the business, security, development, and operation side should work together to improve processes and develop software that can hit the market early.

02 "BizDevSecOps" will be the evolution of DevOps ensuring a holistic cross-functional team approach in delivering customer value

Building on the foundation of DevOps, BizDevSecOps incorporates business and security experts into the software development process from the early beginning, ensuring the targeting of key business outcomes through the DevOps lifecycle. **"Bridging the functions of development and operations is not enough, organizations should rethink the traditional DevOps cross-functional team approach a bit further and follow a holistic concept to deliver customer value,"** says Daniel Zwicker (Authada GmbH). The active involvement of business and security experts in traditional DevOps teams builds the foundation for a collaborative culture which enables a new model of operating to provide superior customer experience by offering customer-centric products and services through a CI/CD pipeline while being compliant with security standards [1]. Although, many organizations have already started to stronger integrate the business (BizDevOps) or security functions (DevSecOps) into their DevOps teams, this development will further accelerate since organizations will increasingly adopt a product-thinking approach to embark on the path of accelerating their customer-driven digital

transformation [31]. Hence, cross-functional teams will be the decisive factor for organizations to develop software more quickly, more secure and to release features that are built specifically to service business objectives, and to be more responsive to user demand [31].

The "Sec" component of BizDevSecOps teams will ensure that security is built from the beginning into the software development lifecycle ranging from prioritizing security requirements as part of the product backlog over continuous testing software code for vulnerabilities to comprehensive monitoring in production [32][33]. This extends the original DevOps objective of delivering software features with high velocity by early detection and mitigation of vulnerabilities [34]. **"Security will be an integral part in all phases of the software development by rethinking DevOps to BizDevSecOps to ensure a security shift left approach,"** says Doron Reuter (ING Groep N.V.). From a technical perspective, organizations will establish security as a mandatory design requirement for their CI/CD pipelines applying the EaC practice to reflect security best practices in their IaC declarative scripts.

The "Biz" component of BizDevSecOps teams will ensure the active involvement of business experts in traditional DevOps teams. This builds the foundation for a collaborative culture which enables a new model of operating to provide superior customer experience by offering valuable products and services through a CI/CD pipeline [1]. **"Including the business perspective in software development is nothing new, it is how the agile movement started. Organizations need to integrate the customer perspective from the early beginning and deliver their software features fast to collect customer feedback,"** states Rahul Verma (Trendig Technology Services GmbH). Integrating business stakeholders into the value stream allows organizations to ensure that the DevOps cycle is aligned with business objectives and in turn allows them to execute their transformation objectives faster and more efficiently [31]. Whereas business professionals will enable a stronger integration of the voice of the customer and allow organizations to see the impact of new products on business value more directly [1][31], IT professionals ensure further automation and reliable software. Additionally, security experts make sure the software development process and new feature comply with security policies [1]. Neither is more important than the other, but they all need to work together in order to provide software that meets customer requirements, is compliant with security standards and is delivered with high quality and speed.

At the core of BizDevSecOps is the common language of code that enables collaboration and co-innovation among the different functions. This common language across the whole software development lifecycle ensures that everyone is working towards the same outcomes. Moreover, it aligns teams into a coordinated BizDevSecOps practice, where all functions have better insight into how each activity contributes to the overall objective. In this context, EaC provides the technical conditions to integrate all IT systems into one common CI/CD platform, such that BizDevOps teams can use the same tools in their daily business. In this context, organizations will have one source of truth for data and tracking points that can be established to measure the speed or waste in certain phases of the software development lifecycle. For instance, if business experts give feedback on an incident in the production environment to operations, operations can use observability tools on the CI/CD platform to determine impacts across different software releases. Subsequently, they can automatically share this information with the developers who can modify their builds accordingly and with the security specialist who endure that the change complies with security standards.

03 Organizations will leverage Machine Learning (ML) algorithms to automate a wide range of processes in the DevOps lifecycle

ML will potentially impact current approaches and process steps in the DevOps lifecycle. Considering the development part of the DevOps lifecycle, ML provides potential to operate and optimize CI/CD pipelines. First, organizations will apply a combination of EaC and ML to establish self-optimizing CI/CD pipelines since ML algorithms enable an automated observation of the pipeline operation, continuously collecting data. **"Leveraging ML algorithms in CI/CD pipelines requires an EaC approach, thus organizations need to follow this approach first to take full advantage of the integration of ML algorithms,"** says Mirco Leimgruber (Swisscom AG). Organizations integrating ML in their CI/CD pipelines will be directly notified if anomalies exist in the process or tools included. Moreover, ML algorithms can predict potential defects in code generated CI/CD pipelines and subsequently react by triggering scripts for destroying and provisioning certain parts of the pipeline. Second, ML algorithms contribute to a stable, resilient operation of CI/CD pipelines by analyzing monitored data that are produced from different events to predict or avoid outages.

Additionally, within the CI/CD process, the application of ML offers various advantages from decision-making process improvements to automated operations and code quality enhancements. Although, this trend is in its adaption still in its infancy due to challenges in preparing large data sets and a challenging regulatory environment especially in the banking industry, there exist four promising developments to watch within the next five years. The first important trend is the application of ML for automated code reviews. The result will be the automated and early detection of code flaws, security issues, and code-related defects which will lead to an enforcement of code and security standards [35]. **"The integration of ML into the DevOps lifecycle offers the possibility of carrying out automated application security vulnerability checks that will lead to both real-time security risk assessments and an increased efficiency due to the reduction of time needed to conduct those tests,"** says Raz Raviv (ING Groep N.V.). Another promising ML development comprises the prediction of potential issues based on data. ML models have the ability to detect patterns and predict signs of failure in cases in which humans hardly perceive them [36]. Such early predictions support DevOps teams to identity and fix issues before they have a negative impact on their software artefact. The next trend to monitor in leveraging the possibilities of ML within the DevOps lifecycle is the possibility of a faster root cause analysis. ML can recognize patterns between implication and activity to determine the root cause behind a defect [36]. Organizations can establish an automated process to fix defects permanently by conducting ML-based root cause analysis that can reduce the risk of human error while providing the development team with recommendations for writing more efficient and performant code [37]. The last important ML trend in the DevOps environment is the adaption of Robotic Process

Automation (RPA) for test automation [36]. RPA can be utilized to transform manual, time-consuming and error-prone tests into automated and streamlined ones. Generally, if ML tools are integrated in the DevOps tool stack, they offer the opportunity to free up DevOps teams from low-value tasks that potentially constrain productivity while at the same time improving software quality and application security [38]. However, implementing the above-mentioned ML trends and realizing its benefits substantially depends on the regulatory environment and fulfillment of the data set conditions by organizations. Thus, it remains to be seen whether a wide adoption of ML in the DevOps lifecycle will become standard in five years.

Considering the operations part of the DevOps lifecycle, ML will also have a key role in ensuring operations continuity after the product or feature is released into production. The pattern-based functionality of ML algorithms provides the possibility to anonymously analyze user metrics and the functioning of applications in production. In the case of any anomalies, ML tools will alert DevOps teams in the case of any issue including possible solutions about how to solve this issue. Although, these predictions sound very promising, organizations such as banks operating in a highly regulated environment might face obstacles in applying ML algorithms to analyze and interpret customer data from production.

04 End-to-End autonomous testing will be the standard approach for software testing leveraging cloud-based services and anonymous test data extracted from production

In the last decade, testing has progressed by small incremental steps. By now shift left is the standard approach in the DevOps and CI/CD community as it helps to mitigate the risk for identifying issues only at a late stage of the software development lifecycle [39]. In future, organizations will continue to further automate test- and development environments that provide all necessary resources, such as interfaces to systems and applications, data and tooling on demand based on cloud services [17]. Or as Nicholas Mills (CircleCi) puts it differently, **"in the upcoming years, we will see an increasing level of automation in testing and validating change in software. If a task is repeated multiple times, it will get automated. This means organizations will realize material cost savings (since the per-minute cost of computing is orders of magnitude cheaper than the per-minute cost of software engineers), and drive significant productivity gains from software development teams (by increasing the efficiency and effectiveness of validating changes to code)."** Autonomous testing allows test cases to be created and executed without much human intervention, thus decreasing the time needed for testing while increasing available time for DevOps teams to focus on more complex issues [40]. Organizations will apply end-to-end autonomous testing that will innovate test creation, maintenance, and execution by applying experience from failed tests and making decisions on how new tests should be created and executed even when conditions such as testing environments or test data change [41]. Further, using autonomous testing will allow for applying tests automatically in parallel to building a change [42].

The combination of EaC and autonomous testing tools provides the possibility for dynamic and flexible testing. If organizations follow an IaC approach enabling a quick reproduction, destruction, and provision of infrastructure, end-to-end autonomous testing can be dynamically conducted to save resources and time. Additionally, the utilization of cloud-based services will make testing faster, cheaper, and more efficient since it will allow organizations to leverage an almost infinite resource pool and flexible infrastructure capacities [42].

In the upcoming years, the application possibilities of cloud-based and autonomous end-to-end-testing solutions will be complemented by production-related test data. This will be enabled by the anonymous extraction of test data from production to ensure test cases reflect real world scenarios. This development will lift the quality of the software delivered on an even higher level since the probability of a case that has not been tested before will decrease. Nevertheless, it is important that organizations ensure that anonymous extracted test data is always compliant with the law in force (e.g., General Data Protection Regulation). Hence, using real-life data without the knowledge and assent of the customer is forbidden and does complicate generating test data. Concerning the financial service industry, the realization of this hypothesis depends on the development of future regulatory conditions since the current regulation forbids tests based on real-life data [43].

Currently, end-to-end autonomous testing leveraging cloud-based services and anonymous test data extracted from production is still in its infancy and hence organizations need to establish the preconditions to take advantage of this functionality. As René Gressly (PostFinance AG) stated, **"in the financial services industry a special focus must be put on anonymizing the data that is used for test cases since inference on the real-life persons cannot only be drawn by having the clear name but also by combining a set of other data – a case that needs to be precluded."** Embracing a high or even total autonomy in testing requires the combination of humans and machines to ensure a scalable, stable, and secure test application [41]. Hence, organizations need to invest in hybrid tools that allow the combination of human and machine capabilities as well as train their software engineers to understand and interpret the test results produced by the machine. Moreover, organizations should focus on the human-machine interface by focusing on how algorithms should be trained based on the emergence of new findings and how the interaction of humans and machines can be designed in an optimal way for the organization itself.

05 Organizations will focus more on developing agile capabilities and following agile values to comprehensively "being agile" instead of just "doing agile"

As organizations are facing the need to constantly transform themselves to stay successful in a dynamic environment, organizations will increasingly adapt agile working methods at the enterprise level to improve delivery, increase speed, and enhance customer experience. However, in order to holistically transform to an agile organization, following a "doing agile" approach that includes the sole implementation of agile practices, methods, and tools is not enough. Organizations will need to develop agile capabilities and embrace agile values to becoming a truly agile organization that follows a "being agile" approach [44]. The shift to "being agile" will substantially increase enterprise-wide agility and ensure that change is sustained. Do-


16
Ray-Ban
ups
Lenovo
GIR
ZERO

minique Mühlbauer (Trendig Technology Services GmbH) said, **"this can be seen as the crucial hypothesis underlying the other hypotheses since, having motivated individuals, taking pride in their craftsmanship, working together in cross-functional teams, will enable delivering high quality software on high speed."** Therefore, organizations will have to focus on the development of agile capabilities such as sensing, learning, adaptability, resilience, speed, innovation, collaboration, and efficiency on different organizational dimensions in future [5]. This approach will lead to the holistic alignment of organizational strategy, structure, processes, technology, and leadership to these agile capabilities. For instance, as agile transformations are grounded in cross-functional teams empowered to operate in rapid decision-making and learning cycles, organizations must ensure that the organizational structure and processes are designed around this purpose [45]. Additionally, organizations will need to invest disproportionally in the upskilling or reskilling of their employees to enable them to work in new roles, collaboration setups, and new fields of responsibility.

Besides focusing on the development of agile capabilities, organizations will also follow agile values that are reflected in behaviors as well as mindsets, and practices drive the required cultural shift sustainably. Therefore, organizations will increasingly focus on empowering their employees in their way of thinking comprising attitudes, orientations, and mentality patterns towards living agile values and principles [46]. One crucial component of this is an empowering leadership approach that embraces an open communication, a culture of constructive criticism and collaboration. This requires leaders to cultivate an environment of psychological safety in which employees are encouraged to experiment and learn without fear of negative consequences to self-image, status, or career [47]. **"The enterprise level of agility can often be defined based on the existence of psychological safety in the teams since "being agile" requires a high degree of psychological safety. Thus, establishing an environment in which every employee feels safe to contribute and speak up is a decisive factor for the overall success of the company,"** says Schlomo Schapiro (DB Systel GmbH). This psychological empowerment will shape employees' consciousness, mindset, and behavior that go beyond simply following agile methodologies or frameworks. Another important component of establishing an agile value system are coaching activities that can support the required mindset shift of employees. Coaches could stimulate thought within agile teams, help them to reflect on their agile way of working, collaboration and attitudes to find further areas for improvement [48].

As organizations embark on their journey to developing agile capabilities and following agile values, it is important to focus on the main objective of agile software development – cross-functional collaboration to enable quick delivery of high-quality and customer-oriented software. Or as Markus Eisele (RedHat) put it differently, **"agility from the textbook is doomed to fail. The idea of cross-functional teams working together is much more important since "being agile" needs the appropriate agile capabilities and mindset." In this context, continuous learning and improvement is a core principle of "being agile".** Successful agile transformations have shown that monitoring of the transformation process, evaluating the incremental product development flow and its impact on performance as well as running regular retrospectives to learn from success and failures are key to sustain a "being agile" approach. A value stream analysis tool might support this monitoring and evaluation process by measuring different tracking points on the value stream of a CI/CD pipeline. This allows DevOps teams to have an overview of all data in one integrated dashboard that indicates areas for improving efficiency in the overall software development lifecycle.

These hypotheses about the future of agile software development illustrate in the direction CI/CD and DevOps might be heading to. **"At Commerzbank, we believe that the underlying trends and dynamics of these concepts should be considered by organizations to further drive their agile transformation,"** says Michael Varona (Commerzbank AG).

Commerzbank's Vision for CI/CD CHAMP 2.0

As in Formula E, teams are constantly searching for improvement, and at Commerzbank we are working with high intensity to bring CI/CD CHAMP to the next level. Thus, we questioned ourselves: how should we anticipate the expected developments to support the fast delivery of high-quality software that meets Commerzbank-customers' expectations? And which tools, methods or processes support the projects in the best way to deliver value to Commerzbank's customers?

Driven by these questions we envisioned CI/CD CHAMP 2.0 that is based on three interrelated developments. First, the CI/CD CHAMP Automated Pipeline 2.0 will contain a mix of self-developed and commercially available tool solutions that follow an IaC approach. Second, CI/CD CHAMP 2.0 will be cloud-native to provide maximum scalability and flexibility for its users. This will be complemented by reusable templates for all the CI/CD CHAMP 2.0 tools that can be tailored for the software project's specific needs and that would enable DevOps teams to build and deploy new software application versions within minutes. Third, CI/CD CHAMP 2.0 aims to be a pioneer in leveraging EaC-approach and implementing it in context of a CI/CD pipeline.

Summary and Conclusion

Software changes the world. Going beyond agile software development, CI/CD and DevOps have pioneered the way software development is thought, conducted, and lived and thus have become essential approaches for many organizations in their software endeavors. In this context, the concept of CI/CD supports organizations in accelerating delivery speed, quality, and customer centricity. Most importantly, the interrelation of the CI/CD cornerstones that comprise people, processes and tools ultimately drive automation, co-innovation, and collaboration within the whole organization to ultimately create customer value.

In this white paper we discussed -besides how CI/CD affects value creation in general and in the financial services industry in particular- also how this concept could be brought to life by providing insights on Commerzbank's CI/CD approach. At Commerzbank, we believe that there is no one-size-fits-all approach when it comes to implementing CI/CD since every organization has its individual requirements and faces different situational characteristics. However, the implementation of a CI/CD pipeline should always follow a set of technical principles comprising the setting up of end-to-end responsibility for teams, reducing risk throughout the software development lifecycle, and integrating short feedback loops.

The true value of CI/CD comes from the enablement of teams to own the whole software development lifecycle from the initial product idea to the release of the final software feature. Additionally, CI/CD as a cross-functional concept leverages on the joint strength of different functions and thus ensures the strong integration of crucial aspects such as security, compliance, and the voice of the customer within the software development process. **"With even more customer interaction becoming digital the concept of CI/CD is a key differentiator for us in the market. It offers unprecedented opportunities to develop, test and deploy software fast and with high quality for our customers,"** says Dr. Carsten Bittner (Commerzbank AG). In this context, the Commerzbank CI/CD CHAMP journey shows that CI/CD as an organization-wide approach to agile software development enables teams to release software features faster and more independently. Employing the metaphor of Formula E, teams being equipped with the necessary safety equipment (CI/CD CHAMP Frameworks), provided with the appropriate racing car (CI/CD CHAMP Automated Pipeline) and having an experienced pit crew (CI/CD CHAMP DevOps Coaching) can autonomously drive innovation and value creation within the organization.

Although many organizations still refine their CI/CD approaches, the next wave of innovation in agile software development is just around the corner and will significantly impact the direction in which the future of CI/CD and DevOps will be heading for in the next five years. We discussed that the evolvement of EaC as the new norm provides the foundation for an array of other developments such as the integration of additional functions such as security and business within traditional DevOps teams. BizDevSecOps teams will reinforce the collaborative approach of software development enabled by CI/CD. We as Commerzbank truly believe BizDevOps is here to stay, representing a roadmap for companies to stay ahead of their agile transformation curve and realize customer-driven business success. However, organizations that want to leverage the full potential of these developments need to fulfill specific prerequisites to be ready to anticipate the respective changes. For instance, organizations need to establish a roadmap that communicates the long-term vision while providing flexibility for their operating teams to quickly re-prioritize activities when relevant trends emerge that contribute to this vision. Being able to anticipate trends also requires the presence of a highly automated and well-functioning CI/CD pipeline that provides consistency and reliability. Additionally, organizations with a monolithic IT legacy need to be open for a certain degree to architectural change to establish flexible and adaptive systems.

The discussed conclusions of this white paper are centered around five key take-aways that are illustrated in figure 9.

We are convinced that acceleration will be the new norm for creating customer value in agile software development and we are excited to contribute with our experience and best practices to this development. We invite you to also share with us insights you gained on your CI/CD journey so far and to discuss where the future of agile software development is heading to. We are looking forward to hearing from you and exchanging our ideas. What do you think about beyond banking in CI/CD and the next wave of innovation in agile software development?

CI/CD accelerates customer value creation in the financial services industry

The implementation of a CI/CD pipeline needs to follow a set of technical principles

CI/CD is not just about tools and processes, it is also about teams that have end-to-end responsibility

An empowering CI/CD approach leveraging the strength of cross-functional teams is key

Everything as Code (EaC) as the new norm meaning automation via software will be everywhere

Figure 9: Key Take-Aways

Appendix

Methodological Approach

This white paper was developed in a collaborative research project including business and technical experts from Commerzbank as well as researchers from the Business Engineering Institute St. Gallen. We based our insights on three main sources. First, we conducted an extensive desk research between May and July 2021. The desk research comprised literature from academic discussion, practical studies, other white papers, blogs, and news articles. Second, we included both our technical as well as our business experience into this white paper, for example from deep dives with different Commerzbank CI/CD CHAMP business experts. Third, we conducted interviews with internal Commerzbank teams and with representatives from external organizations as well to gain additional perspectives on our research. From July to August 2021, we interviewed three different Commerzbank product teams on their experience in applying CI/CD CHAMP to share their perspectives on CI/CD CHAMP. During August and October 2021, we conducted 9 interviews with 12 representatives from technology corporations, financial institutions, consulting companies, and FinTechs to validate our findings and to further develop our ideas about the future of agile software development. We documented our interviews and collected general insights such as recurring statements or mentioned trends. Specific quotations have also been collected and in accordance with our interview partners integrated into this white paper. The results of our research and our interviews has been extensively discussed and refined during several meetings during September and October. This approach helped us to develop a more nuanced picture of the CI/CD case studies presented in chapter 3 (integrated results of the interviews with Commerzbank product teams) and the hypotheses about the future of agile software development in chapter 4 (integrated results of interviews with representatives of external companies).

Our External Interview Partners

We conducted 9 interviews with the following representatives from a diverse set of companies. We hereby would like to express our gratitude towards our interview partners for spending their valuable time as well as sharing their insightful ideas with us. We appreciate the open conversations and the challenging discussions.

Organization	Interview Partner	Function
Authada GmbH	Daniel Zwicker	Chief Technology Officer
Circle Internet Services Inc. (“CircleCI”)	Nicholas Mills	General Manager EMEA
DB Systel GmbH	Schlomo Schapiro	Chief Architect Cloud
ING Groep N.V.	Raz Raviv	Cyber Innovation Tech Lead
	Doron Reuter	Venture Lead
Microsoft Corporation	Daniel Meixner	DevOps Architect
PostFinance AG	René Gressly	IT Architect
Redhat	Markus Eisele	Developer Strategist EMEA
Swisscom AG	Mirco Leimgruber	DevOps Engineer
Trendig Technology Services GmbH	Dominique Mühlbauer	Quality Advocate & Agile Evangelist
	Rahul Verma	Head of Test Automation Department

Our Commerzbank Sponsors and Use Case Partners

We would also like to warmly gratitude our internal sponsors for their support during the preparation phase of this white paper. Moreover, we also want to thank our colleagues from the API Banking Cluster, Big Data & Advanced Analytics and Securities Frontend for spending time and sharing their valuable feedback on how CI/CD CHAMP supports them in their daily business.

Our Commerzbank Sponsors

Sponsor	Function
Dr. Jörg Oliveri del Castillo-Schulz	Chief Operating Officer
Dr. Carsten Bittner	Divisional Board Member, Group Technology Foundations
Sabine Vigelius	Key Area Lead of Customer Process & Data Management
Dr. Alena Kretzberg	Head of comdirect, Marketing & Digital Banking
Dr. Lars Friedrich	Head of Group Securities & Brokerage
Michael Varona	Head of Group Digital Transformation

Our Use Case Partners

Use Case	Use Case Partner	Function
API Banking	Akli Amar-Youcef	Scrum Master
	Arnd Bischoff	Product Owner
	Christian Pfaff	Solution Architect
	Thorsten Roth	Expert Project Engineer & Scrum Master
	Volker Sulzbach	Subject Matter Expert API & EAI Technologies
Big Data & Advanced Analytics (BDAA)		
	Michael Bellinger	Business Expert
	Peggy Schuchert	Product Owner
Securities Frontend		
	Klaus Donath	Software Engineer
	Claus Weber	Senior Product Owner

About the Authors

Commerzbank Cluster CI/CD CHAMP

The cluster CI/CD CHAMP provides a reliable end-to-end solution consisting of a fully automated self-service pipeline, modern software development and test methods, corresponding tools and DevOps coaching. The objective is to enable all clusters of Commerzbank delivery organisation to deliver regulatory compliant software fast and efficient to customers in high quality and with short release cycles.

Together with our research partner the Business Engineering Institute St. Gallen, we published this White Paper.

Florian Meiser
Cluster Lead CI/CD CHAMP
Commerzbank AG
Florian.Meiser@commerzbank.com

Damian Waszak
Product Manager CI/CD CHAMP
Commerzbank AG
Damian.Waszak@commerzbank.com

Fonlinda Frasheri
System Architect CI/CD CHAMP
Commerzbank AG
Fonlinda.Frasheri@commerzbank.com

Jan Willemsen
Senior Product Owner CI/CD CHAMP
Commerzbank AG
Jan.Willemsen@commerzbank.com

Stephanie Radet
Product Owner CI/CD CHAMP
Commerzbank AG
Stephanie.Radet@commerzbank.com

Rostam Rahnema
Senior Product Owner CI/CD CHAMP
Commerzbank AG
Rostam.Rahnema@commerzbank.com

Business Engineering Institute St. Gallen

The Business Engineering Institute St. Gallen is a Swiss institute that conducts research in the areas of ecosystems, digital transformation, and disruptive technologies to develop academic insights with practical impact. We collaborate with institutions such as the University of St. Gallen, the University of Göttingen and partner companies from the financial services sector to develop answers to strategic questions related to technology and information-based value creation in the digital age.

Katharina Schache
Research Associate & PhD Candidate
Business Engineering
Institute St. Gallen AG
Katharina.Schache@bei-sg.ch

Oliver Kutsch
Head of Consulting
Business Engineering Institute St. Gallen AG
Oliver.Kutsch@bei-sg.ch

Dennis Vetterling
Research Associate & PhD Candidate
Business Engineering Institute St. Gallen AG
Dennis.Vetterling@bei-sg.ch

Acknowledgements

While working on this white paper, we have had numerous discussions and received lots of valuable feedback from within and outside of Commerzbank. We would like to express our sincere gratitude towards all those who supported us on this journey. We would like to thank Christian Betz and Marc Burkhalter constructive feedback and their valuable ideas. We would also like to thank Nigel Champion for his linguistic revision.

Glossary

Application Programming Interfaces (API)	Application Programming Interfaces are a technology which allows software applications to talk to each other in a machine-readable way.
BizDevSecOps	BizDevSecOps is a portmanteau word that expresses the combination of bridging the functions of business, development, security and operations in one single team.
Cluster	Organizational structure of Commerzbank, a Cluster describes a group of teams working in the same business area.
Cross Channel Banking (CCB)	CCB is the main Commerzbank portal, comprising of core portal solution and multiple financial products across different channels.
Continuous Integration and Continuous Deployment (CI/CD)	CI/CD forms the operational backbone of DevOps practices and essentially allows organizations to provide users with new software products or features at any time in a sustainable way.
Commerzbank Delivery Guide CHAMP (CDG CHAMP)	Commerzbank Delivery Guide CHAMP helps developers to be guided in the development process containing all relevant documentation requirements.
Commerzbank Hyper Acceleration Master Pipeline (CHAMP)	Commerzbank Hyper Acceleration Master Pipeline describes the CI/CD Pipeline of Commerzbank and incorporated services.
Delivery Organisation	Organizational structure of Commerzbank. All software development happens in Clusters, all Clusters form Delivery Organization.
DevOps	DevOps describes the paradigm of bringing Development and Operations closer together in the software development process.
Everything as Code (EaC)	EaC is the idea that all functionalities of a software development pipeline can be managed in using code.
Failing Forward	Failing Forward describes an approach based on the idea that mistakes are not evil, but necessary for learning and continuous improvement. Employees should leave their comfort zone to drive development.
Integrated Development Environment (IDE)	IDEs are applications that support developers in developing software in offering a compilation of different tools that are needed during programming.

Kanban	Framework for establishing Agile and DevOps in software development. The different tasks are visualized on a Kanban Board to enable the team members to gain an overview anytime [49]
Mainframe	A mainframe is a complex and powerful computer system
Product Owner (PO)	The Product Owner (PO) is a member of the agile team who is responsible for defining user stories and prioritizing the backlog. POs streamline the execution of project priorities and ensures the representation of customer interests in the software development process [50]
Product Owner Cockpit CHAMP (PO-Cockpit CHAMP)	The PO-Cockpit CHAMP is a tool that guides the PO`s through the development workflow
Scrum	Scrum is a framework that supports people, teams and organizations to create value through adaptive solutions [51]
Shift Left	Shift left is the practice of shifting performing unit or integration tests and code quality checks early in the software development process and can result in significant cost reductions as well as in increased software quality since potential pitfalls in the software code are identified early during the development process [12]
Technology Stack	Describes the combination of frameworks and tools used for software development.
Test-Driven Development (TDD)	Test-Driven Development is a procedure in which the programmer consistently creates software tests before the components to be tested, also called test-driven programming
User Story	A user story is an informal, general explanation of a software feature written from the perspective of the end user. Its purpose is to show what value a software feature has for a customer [52]
Value Stream Mapping (VSM)	Value Stream Mapping is a methodology for analyzing the steps and their sequence of a current process easily quantifying the time and working effort needed at each process step. This allows for designing a target process that minimizes the process time while maximizing the efficient use of resources and the process quality [53]


TAGHeuer TAGHeuer TAGHeuer TAGHeuer

Literature Overview

[1] McKinsey & Company, „The five Core IT shifts of scaled agile organizations," 2021. [Online]. Available at: https://www.mckinsey.com/business-functions/people-and-organizational-performance/our-insights/the-five-core-it-shifts-of-scaled-agile-organizations [Accessed: 20.10.2021]
[2] The Boston Consulting Group, „Leaner, Faster, And Better With DevOps," 2017
[3] Oliver Wyman, „When Vision and Value Collide - The State Of The Financial Services Industry 2020," 2020
[4] K. Beck, M. Beedle, A. van Bennekum, A. Cockburn, W. Cunningham, M. Fowler, J. Grenning, J. Highsmith, A. Hunt, R. Jeffries, J. Kern, B. Marick, R.C. Martin, S. Mellor, K. Schwaber, J. Sutherland, D. Thomas, „Manifesto for Agile Software Development," 2001. [Online]. Available at: https://agilemanifesto.org/. [Accessed: 05.11.2021]
[5] D. J. Proba, „Technikkompositionen für das situative Methoden-Engineering agiler Methoden," 2021
[6] J. Halstenberg, B. Pfitzinger, T. Jestädt, „DevOps - Ein Überblick," 2020
[7] S. Blank, „Why Continuous Deployment May Mean Continuous Customer Disappointment," 2014. [Online]. Available at: https://hbr.org/2014/01/why-continuous-deployment-may-mean-continuous-customer-disappointment. [Accessed: 03.11.2021]
[8] C. Riley, „Waterfall to Agile to DevOps: The State of Stagnant Evolution," 2014. [Online]. Available at: https://devops.com/waterfall-agile-devops-state-stagnant-evolution/. [Accessed: 05.11.2021]
[9] BCG, „Going All In With DevOps," 2018
[10] D.K. Rigby, J. Sutherland, A. Noble, „Agile at Scale," 2018. [Online]. Available at: https://hbr.org/2018/05/agile-at-scale. [Accessed: 05.11.2021]
[11] M. Virimani, „Understanding DevOps & Bridging the gap from Continuous Integration to Continuous Delivery," Firth international conference on Innovative Computing Technology, 2015, doi: 10.1109/INTECH.2015.7173368
[12] J. Benjamin, J. Mathew, „Enhancing the efficiency of continuous integration environment in DevOps," IOP Conf. Series: Materials Science and Engineering, 1085, 2022, doi: 10.1088/1757-899X/1085/1/012025
[13] E. Laukkanen, J. Itkonen, C. Lassenius, „Problems, causes and solutions when adopting contunuous delivery - A systematic literature review," Information and Software Tecnology, 82, pp. 55-79, 2017, doi: 10.1016/j.infsof.2016.10.001
[14] P. Rodriguez, A. Haghighatkhah, L.E. Lwakatare, S. Teppola, „Continuous deployment of software intensive products and services: A systematic mapping study," 2016, doi: 10.1016/j.jss.2015.12.015
[15] N. Forsgren, J. Humble, „The Role of Continuous Delivery IT and Organizational Performance," 2016, Proceedings of the Western Decision Sciences Institute (WDSI), doi: 10.2139/ssrn.2681909
[16] A. Mishra, Z. Otaiwi, „DevOps and software quality: A systematic mapping," Computer Science Review, 38, 2020, doi: 10.1016/j.cosrev.2020.100308
[17] DevOps Institute, „2021 Upskilling Enterprise DevOps Skills Report," 2021
[18] Puppet, „2021 State of DevOps Report," 2021
[19] K. Conboy, S. Coyle, X. Wang, M. Pikkarainen, „People Over Process: Key People Challenges in Agile Development," IEEE Software, 2010, doi: 10.1109/MS.2010.132
[20] Atlassian and CITE Research, „2020 DevOps Trends Survey," 2020
[21] G. Mayank, „DevOps phases across Software Development Lifecycle," TechRxiv, 2020, doi: 10.36227/techrxiv.13207796.v2
[22] DORA, „Accelerate: State of DevOps 2019," 2019
[23] McKinsey & Company, „Managing security as code enables companies to create value in the cloud securely," 2021. [Online]. Available at: https://www.mckinsey.com/business-functions/mckinsey-digital/our-insights/security-as-code-the-best-and-maybe-only-path-to-securing-cloud-applications-and-systems. [Accessed: 05.11.2021]
[24] Commerzbank AG, „Open Banking White Paper - The Future of Collaboration in Corporate Banking," 2020
[25] J. Kanjilal, „6 lessons from the everything-as-code-shift," TechBeacon, 2021. [Online]. Available at: https://techbeacon.com/enterprise-it/6-lessons-everything-code-shift. [Accessed: 2010.2021]
[26] E. Batraski, „The beginning of „Everything as Code," 2020. [Online]. Available at: https://medium.com/swlh/the-beginning-of-everything-as-code-a25c4e9a75e9. [Accessed: 29.10.2021]
[27] A.A. Zeeshan, „Automating Everything as Code," DevSecOps for .NET Core. 2020. doi: 10.1007/978-1-4842-5850-7_4
[28] C. Tozzi, „Everything as Code: What It Is and Why It`s Gaining Traction," 2020. [Online]. Available at: https://www.itprotoday.com/development-techniques-and-management/everything-code-what-it-and-why-its-gaining-traction [Accessed: 03.11.2021]
[29] Capgemini, „DevSecOps in Regulated Industries - Accelerating Software Reliability & Compliance," 2021
[30] G. Rong, J. Zefeng, H. Zhang, Y. Zhang, W. Ye, D. Shao, „DevDocOps: Enabling continuous documentation in alignment with DevOps," 2019, doi: 10.1002/spe.2770
[31] Capgemini, „BizDevOps to enable a product-thinking approach," 2020. [Online]. Available: https://www.capgemini.com/2020/10/bizdevops-to-enable-the-product-thinking-approach/. [Accessed: 27-Oct-2021]
[32] Puppet, CircleCI, Splunk, „State of DevOps Report 2019," 2019
[33] Puppet and CircleCI, „State of DevOps Report 2020," 2020
[34] Accenture, „From NoOps to DevSecOps and beyond," 2020. [Online]. Available: https://www.accenture.com/us-en/blogs/software-engineering-blog/emerson-devops-devsecops. [Accessed: 15.10.2021]
[35] E. Kinsbruner, „How artificial intelligence (AI) and machine learning are changing DevOps," 2020. [Online]. Available at: https://enterprisersproject.com/article/2020/9/artificial-intelligence-ai-and-machine-learning-change-devops. [Accessed: 05.11.2021]
[36] M. Trivendi, „12 Ways AI Is Transforming DevOps," n.n.[Online] Available at: https://www.tadigital.com/insights/perspectives/12-ways-ai-transforming-devops. [Accessed: 05.11.2021]
[37] Open Data Science, „How AI and ML are the Next Evolutionary Step for DevOps," 2020. [Online]. Available at: https://medium.com/@ODSC/how-ai-and-ml-are-the-next-evolutionary-step-for-devops-61565448a468. [Accessed: 03.11.2021]
[38] M. Vizard, „GitLab Acquires UnReview to Further AI Ambitions," 2020. [Online]. Available at: https://devops.com/gitlab-acquires-unreview-to-further-ai-ambitions/#:~:text=GitLab%20announced%20this%20week%20it,previous%20efforts%20and%20current%20workloads. [Accessed: 29.10.2021]
[39] BCG , „Going All In With DevOps," 2018
[40] E. Kinsbruner, „How artificial intelligence (AI) and machine learning are changing DevOps," 2020. [Online]. Available at: https://enterprisersproject.com/article/2020/9/artificial-intelligence-ai-and-machine-learning-change-devops. [Accessed: 05.11.2021]
[41] SmartBear Software, „Six Stages from Manual to Autonomous Testing," n.n. [Online] Available at: https://smartbear.com/resources/ebooks/six-stages-from-manual-to-autonomous-testing/. [Accessed: 03.11.2021
[42] P. Clauson, „Autonomous Testing`s Role in Agile DevOps," 2019. [Online]. Available at: https://devops.com/autonomous-testings-role-in-agile-devops/. [Accessed: 05.11.2021]
[43] Proton Technologies AG, „Complete guide to GDPR compliance," 2021. [Online]. Available at: https://gdpr.eu/. [Accessed: 05.11.2021]
[44] McKinsey & Company, „Doing vs being: Practical lessons on building an agile culture," 2020. [Online]. Available at: https://www.mckinsey.com/business-functions/people-and-organizational-performance/our-insights/doing-vs-being-practical-lessons-on-building-an-agile-culture. [Accessed: 05.11.2021]
[45] McKinsey & Company, „Building agile capabilities: The fuel to power your agile „body"," 2019. [Online] Available at: https://www.mckinsey.com/business-functions/people-and-organizational-performance/our-insights/building-agile-capabilities-the-fuel-to-power-your-agile-body. [Accessed: 05.11.2021]
[46] K. Eilers, B. Simmert, C. Peters, „Doing Agile vs. Being Agile - Understanding Their Effects to Improve Agile Work," 2020, ICIS 2020 Proceedings.6.
[47] Scaled Agile Inc., „Lean-Agile Leadership," 2021. [Online]. Available at: https://www.scaledagileframework.com/lean-agile-leadership/. [Accessed: 14.10.2021]
[48] V. Stray, B. Memon, L. Paruch, „A Systematic Literature Review on Agile Coaching and the Role of the Agile Coach," 2020, in M. Morison, M. Torchiano, A. Jedlitschka, „Product-Focused Software Process Improvement," PROFES 2020, Lecture Notes in Computer Science, 12562. doi: 10.1007/9878-3-030-34148-1_1
[49] Atlassian , „Was ist Kanban?," n.n. [Online]. Available at: https://www.atlassian.com/de/agile/kanban. [Accessed: 05.11.2021]
[50] Scaled Agile Inc., „Product Owner," 2021. [Online]. Available at: https://www.scaledagileframework.com/product-owner/. [Accessed: 05.11.2021]
[51] Scrum.org, „Scrum Glossary," 2021. [Online] Available at: https://www.scrum.org/resources/scrum-glossary. [Accessed: 05.11.2021]
[52] Atlassian, „User Storys mit Beispielen und Vorlagen," n.n. [Online]. Available at: https://www.atlassian.com/de/agile/project-management/user-stories. [Accessed: 05.11.2021]
[53] I.S. Lasa, C.I. Laburu, R. de Castro Vila, „An evalutaion of the value stream mapping tool," Business Process Management Journal, Vol. 14, No. 1, 2008, pp. 39 - 52

COMMERZBANK

Commerzbank AG
Head Office
Kaiserplatz
Frankfurt am Main
www.commerzbank.de

Postal Address
60261 Frankfurt am Main
Tel.: +49 69 136-20
info@commerzbank.com