

# Curriculum Vitae

## Personal Information

Name Florian Böhmak  
Address Bleichweg 14, 01920 Nebelschütz  
Email florian@fnbk.cc  
Phone +49 160 8018242  
Birth date 06 april 1987  
Birthplace Räckelwitz (Germany)



## Education/Training/Employment

08/2004 - 06/2005	Traverse City Central High School, USA	Degree: High School Diploma
09/2005 - 06/2007	Sorbisches Gymnasium, Bautzen	Degree: Abitur
10/2007 - 09/2009	Siemens Technik Akademie, Erlangen	Degree: Associate Engineer in Information and Computer Systems
04/2009 - 09/2010	Siemens AG, Erlangen	Employment: Windows System Administrator
10/2010 - 08/2013	Technische Universität, Darmstadt	University: Computer Science
10/2011 - now	Darmstadt, Weinheim, Munich, Berlin, Stuttgart, Dresden, Hamburg	Freelance Software Developer

## Personal Skills

### **Siemens AG, Trainee and Windows Administrator (10/2007 - 09/2010)**

<https://www.siemens.com>

Software development with C/C++:

- program microcontrollers with Assembler and Embedded C, implement LCD drivers (C167 microcontroller, 16-bit, RTOS, CAN bus)
- program and configure Siemens automation systems (SIMATIC S7)

Operations:

- monitor and operate Siemens Windows Backend Services (Account-, File-, Share-, Print-, Server- Patch-, Management), over 100 servers

### **Web Consulting (10/2011 - 09/2012)**

<http://green-system.de>, <http://tischlerei-johannes-krause.de>, <http://matej-zieschwauck.com>,  
<http://eu-found.de>

Software development with Ruby on Rails and JavaScript:

- build two websites with a slim technology stack (HTML/CSS, JavaScript)
- build one custom content management system using Ruby on Rails (Amazon EC2, Linux, PostgreSQL)

Operations:

- learn to communicate with customers about website requirements (project costs, performance, design, content, structure, technologies, editing features)

### **Smooster CMS (10/2012 - 09/2013)**

<https://smooster.com>

Software development with Ruby on Rails and JavaScript:

- build new features for the content management platform "smooster" (Ruby, JavaScript, MongoDB)
- integrate authentication system (omniauth, REST, AJAX)
- apply TDD and Clean Code techniques, rewrite over 40% of the software (clean views, thin controllers, normalized models, embrace REST)
- extend frontend test automation (RSpec, Selenium)

Operations:

- deliver media assets on Amazon AWS infrastructure (bash, Unix, caching, CDN)
- scale client websites (load balancing)
- optimize conversion rate using social media channels (SEO)

## **Web Screen Sharing and Collaboration Application (04/2013 - 05/2013)**

<https://teamondo.de>

Software development with JavaScript:

- build a real time communication server for HTML clients with WebSockets (chat messages, click events, etc). Stability and scalability were the main requirements (NodeJS, Redis, Pub/Sub, MongoDB, Socket.io, Unix)

## **Build TV data API (09/2013 - 05/2015)**

<https://tvib.tv>

Software development with Go, JavaScript:

- implement microservices and libraries written in Go (Kafka, Zookeeper, Sarama, PostgreSQL)
- implement tvib REST API for real time triggers (e.g. banner synchronization), written in Go
- implement a “store and forward” client for distributed messaging (Kafka, Zookeeper, Sarama), written in Go
- Implement browser apps to manage videos, analytics and customer reports (Angular 1.0, Node, Express)
- implement “TV video cutter” for real time tracking of TV commercials (C/C++, ffmpeg, cbuf, pthreads)

Operations:

- design and implement deployment system on Amazon AWS (Chef, Autotools, Docker)
- patch dvblast to stream multiple TV channels over TCP/IP, which includes setting up video tuner cards, configuring smart cards for signal decryption (Unix, bash, C++, Python)
- improve systems performance by hunting down memory leaks (valgrind), identify performance bottlenecks (gprof), monitor CPU time and memory utilization (caching, paging, swapping)

## **Electronic Tolling System (05/2015 - 10/2015)**

<https://www.t-systems.com>

Software development with C/C++:

- improve startup time: fix performance bottlenecks, rewrite sequential startup to execute in parallel (threads)
- optimize dynamic linker loading time of shared libraries using prelink, hash-table (ELF format)
- program services for embedded linux system (embedded automotive platform) and integrate into the service architecture
- implement suspend/wakeup features into API abstraction layer

Operations:

- measure startup times on embedded device, using standard unix tools: bash, awk, sed, grep, /proc
- extend cross-compile build system for ARMv6 architecture to optimize shared libraries dynamic loading time using prelinking (Unix, cmake)

## **Traceability software for automobile system (10/2015 - 10/2015)**

<https://www.synatec.de>

Software development with C/C++, QT:

- write software to manage xml files, improve event traceability and handle increased message throughput
- track down memory leaks/crashes and integrate into the existing architecture, measure performance

## **Car-material mapping software (11/2015 - 11/2015)**

<https://www.daimler.com>

Software development with C/C++, QT:

- improve data mapping performance using data caching, document data mapping algorithms
- fix build system errors (Unix, bash, QMake)

## **Arduino and IO-Link device (12/2015 - 04/2016)**

<https://balluff.com>

Software development with C/C++, Python:

- rewrite Arduino firmware (implement a state-machine, clean code practices)
- fix software errors on custom designed and soldered Arduino board
- write C/C++ unit tests for embedded C code (CppUnit test framework)
- discover software bugs (improve error handling), write stress tests (Python)
- document test infrastructure (doxygen)

Operations:

- connected devices: RFID Tag reader, Barcode reader (understand protocol, program, connect, and configure)
- configure build jobs on jenkins (Unix, bash)
- support documentation work to satisfy certification requirements from TÜV

## **Product Information Management (PIM) software (05/2016 - 11/2017)**

<https://otto.de>

Software development with Go, JavaScript:

- write Golang software in microservice architecture (MongoDB, HTTP, Kafka, Zookeeper, Protobuf)
- design microservice architecture (separation of concerns, service contracts, hexagonal architecture, DDD)
- connect HTML frontend with backend services (ReactJS, react-redux, react-router)
- software craftsmanship - apply "clean code practices" (Technical Dept, CQRS, IOSP, pair programming, TDD)

Operations:

- setup development infrastructure virtual machine (unix, bash, virtualbox, vagrant, ansible, docker)
- refine scrum process: introduce acceptance tests as contract between stakeholders and developers (Acceptance Test Design Principles, ROBOT-test-framework)

## **Personal Research on “How to structure Software” (12/2017 - 09/2018)**

Software development and operations:

- write Golang backend using NEW clean code principles (IOSP, PoMo)
- write three sample applications in ReactJS (redux, jest, cosmos)
- redesign build-system (Makefile, Bash) with cleaner JavaScript tasks (runjs)
- setup custom CI-Pipeline into kubernetes cluster for a Docker workshop (Go, Kubernetes)

Presentation at Socrates Conference about how to structure React applications (ReactJS, Google Firebase)

<https://github.com/how-to-structure-software/buyer-seller>

Docker workshop at DWX Conference (Python, AWS CLI, Terraform)

<https://github.com/high-value-team/docker-workshop>

## **Search Portal for Video Marketers (07/2018 - 09/2018)**

<https://video-oase.de>

Prototype and implement Proof of Concept using server less technologies:

- create prototypes with <https://proto.io/>
- stack: Google Firebase, Paypal, React, Bootstrap
- features: sign up/in/out, password forget/change, protect routes, simple search, excel import tool
- demo: <https://video-oase.firebaseio.com/>

## **Enterprise Service Bus (ESB) with Azure Technologies (10/2018 - 04/2019)**

<https://www.zeiss.de>

Build and operate ESB with message throughput of over 3 million messages per day:

- Azure Service Fabric - manage docker cluster, debug runtime errors, analyse performance bottlenecks
- Application Insights - write queries for: data analysis, alerts, dashboards
- CosmosDB: plan and execute extensive migration - risk analysis, coordinate, create tools (C#, Powershell, Azure Data Factory Scripts, ARM templates)
- analyse NSG rules (Network Security Group) and evaluate blocked IP Addresses
- manage multiple environments (Dev, Test, Stage, Prod) and improve Mock Server (TypeScript)

## **Kubernetes “Good Practices” Workshop (06/2019)**

<https://www.zeiss.de>, <https://www.handelsblatt.com>

Onsite Training in cooperation with Microsoft:

- Target Audience: DevOps - Developers & Operations
- Description: Kubernetes Best Practices with a focus on DevOps, Monitoring, Scaling, Volumes, Security, Mesh
- High Level Concepts as guiding principles
- Demonstrations (Azure DevOps pipeline workflow, Monitoring, Scaling, Networking)
- Hands-On Exercise with Workbook (kubectl, pods, services, ingress, dns, scaling)
- <https://github.com/kubernetes-workshop/workshop>

## **Designing new Azure Tenant for KPMG Swiss (05/2019 - 07/2019)**

<https://home.kpmg/ch>

Operations:

- setup Azure DevOps build and release pipelines
- analyze security aspects (managed identities, certificates)
- encryption of VMs, Disk Encryption, SQL Server (KEK, HSM, KeyVault)

## **Computer Management System (Microsoft SCCM) (08/2019 - 04/2020)**

<https://www.daimler.com>

Software design:

- design modular and testable system based on cell-architecture (goal: reduce manual test from 90% to 10%)
- evaluate technologies and prepare on-premise system to become cloud ready: self hosted Azure APIM, Azure Functions, Docker)
- create bridge between windows on-premise authentication (IWA) and cloud token based authentication (JWT)

Software development with C#:

- implement cell based architecture and fractal architecture for better teamwork (clean contracts, middlewares, ports & adapters)

Streamline SCRUM process by introducing:

- example-mapping (req. eng.)
- gherkin acceptance-tests (living doc.)
- feature slicing (draw better UML)
- coding guidelines (improve code reviews)
- pair-programming (Know-How transfer)

Operations:

- setup DevOps CI/CD pipelines for automated builds, tests, deployments (azure-pipelines.yml)
- infrastructure Stack: Windows Server 2019, Microsoft SQL 2019, IIS 10, .Net Framework 4.7.2, .Net Core 3.1, NuGet, RabbitMQ

## **Connected Quality Platform (05/2020 - now)**

<https://www.zeiss.de>

Software design and development with C#, PowerShell, JavaScript:

- define software development template for other teams about software structure (cell architecture, testing)
- write custom kubernetes controller that creates CustomResourceDefinitions (Traefik IngressRoutes)
- define and implement custom manifest for IoT Hub deployment (deploy ARM resources, expand variables)

Operations:

- setup kubernetes cluster (Rudr, Dapr, Nginx, Traefik, IoT Hub on Kubernetes)
- write Azure DevOps Yaml Pipelines (ACR Webhooks, trigger DevOps builds with Logic Apps)