Overview

This overview is meant to give a compressed view of my current set of skills and interests. See the Career path chapter for more information on how those are backed by experience and a more in-depth view into my skill set.

The Personal projects section contains a few projects that have no relation to any employment.

Technologies and languages

- *Common* – Rust, Kotlin, Java, x86/64 assembly, C#
- *Web* – TypeScript, JavaScript, WebAssembly, WebGL, HTML, CSS/SASS/LESS, WebAudio API, WebMidi API, responsive design
- *Gaming* – OpenGL, GLSL, Vulkan/SPIR-V, OpenCL
- *Embedded* – digital electronics, IoT, home automation, logic analyzers, Raspberry Pi, AVR/Arduino, Cortex-m
- *Databases* – MySQL/MariaDB, PostgeSQL, MongoDB
- *Platforms* – Linux (Debian, Mint), Linux containers, VirtualBox, Microsoft Windows

Special skills

- quickly learn learn new languages and technologies
- write and document high quality code
- visualize complex issues
- find generalized solutions by abstracting over the problem domain
- design public APIs
- optimize for different requirements (performance, memory, complexity, latency, …)
- reverse-engineering and debugging
- find creative solutions and think out-of-the-box
- develop in technically constrained environments
- strictly adhere to standards and specifications if present

Other

- Languages: German (first language), English (fluent; spoken and written), French (basic)
- Location and timezone: Germany (CET UTC+1:00 and CEST UTC+2:00)
- I'm a co-organizer of the Rust User Group Cologne (https://rust.cologne/)
- GitHub profile: https://github.com/kawogi/ (a little disclaimer: most of my previous work is not public and some of the projects there have been abandoned due to lack of free time)
- I'm interested in the current development of A.I. and machine learning and I have a basic understanding of the matter, but so far I haven't found the time to familiarize myself with the details, yet.

Career path

2019 – 2022 Audiotool Inc. `full-time`

Main projects

Rewrite of the Audio Engine `rust` `audio` `real-time` `multi-threading` `browser` `webassembly` `javascript` `typescript` `worker` `audioworklet` `dsp` `math` `webmidi` `velocity` `java` `windows` `macos` `linux`

- Design and develop a multi-threaded audio engine for a web-based online DAW.
- Write templates for code generators to interface with the existing code base.
- Write tests to ensure compatibility with the existing communication protocols.
- Research and develop lock-free data types to improve real-time metrics and avoid audible glitches.
- Use profilers to validate run-time behavior in the browser.
- Improve performance of trigonometric and other low level functions by using Rust benchmarks.
- Release prototypes to users to gather results in the field and collect feedback.
- Develop a tool for AST based translation from Java to Rust to facilitate the migration.

Redesign of the public-facing Website `html5` `css` `less` `sass` `zeplin` `svg` `responsive` `scala` `webgl` `typescript`

- Instruct designers to ensure the new design covers the existing and future scope of operation.
- Implement responsive HTML/CSS templates based on the design.
- Help to design a template system based on our data model.
- Re-implement the WebGL-based waveform in the website player to match the new design and improve the underlying data series.

Web server maintenance `jdt` `flame graph` `stress test` `garbage collector` `exceptions` `java` `scala` `rust` `monitoring` `metrics`

- Develop a tool to put the web server under stress and evaluate the results.
- Use JDT to trace the source of memory leaks and performance bottlenecks in the web server.
- Develop and monitor metrics of the web servers to notice and trace production problems.

Web server replacement `java` `scala` `kotlin` `dart` `rust` `go` `co-location` `event bus` `event sourcing` `micro-service` `self-contained system` `eventual consistency` `replication` `fail-over` `latency`

- Develop a web server architecture to improve scalability, stability, availability and to enable co-locations.
- Research of programming languages and frameworks to support our requirements.

Analytics `rust` `events` `service` `metrics`

- Design and implement a generalized event storage to store raw user interactions with our services.
- Process the raw events with a state machine to derive higher level semantics.
- Use the results to derive and visualize business metrics and their development over time.

Pixtunes (contractor) `ai` `rust` `migration` `audio` `maxmsp`

- Create architectural design of an AI-driven audio engine written in Rust.
- Design a migration plan for the existing prototype written in MaxMSP
- Implement prototypes in Rust.

Side projects

- *Image resizing* – Implement a fast image resizing algorithm to reduce dependencies from external libraries and enable parallelism. `java` `lanczos resampling`
- *Playback stability* – Research and development for improving the playback stability on certain mobile devices. `safari` `web audio api` `mobile` `javascript`
- *Browser security* – Re-configure HTTP servers to fulfill the required security criteria to enable SharedArrayBuffers and AudioWorklets in clients. `corp` `coep` `http` `security`
- *Server-side streaming* – Create a concept for offloading compute-heavy tasks from the client to the server to support low-end devices. `streaming`
- *AI research* – Research and development about the integration of AI in order to improve or extend existing services. `ai`
- *Audio fingerprinting* – Develop a fingerprinting method for finding duplicate samples to reduce storage requirements. The same method was used to detect more copyright infringements in the sample database. `soft-hashing` `rust` `heuristics`
- *Real-time collaboration* – Work on a concept for decentralized conflict-resolution when multiple users are working on the same document at the same time. `concurrency` `transactions`
- *Shader programming* – Create drafts of visual improvements using WebGL shaders. `webgl` `glsl`
- *Postmaster Tools* – Debug false domain reputation and provide fix to improve reach of newsletter. `dkim` `dmarc` `spf`
- *Reputation system* – Create a concept of a reputation system (similar to StackOverflow) to create incentives for users to obey the community guidelines and simplify moderation. `moderation` `community`
- *Asset analysis* – Use user provided tags to find similar tracks and artists using a force-directed graph. Apply cluster analysis to the resulting graph to obtain hidden musical relations and genres. `cluster analysis` `graph` `optimization`

Other tasks

- *On-call duty* – On-call duty to ensure a higher availability of user-facing services.
- *Code quality* – Establish company-wide guidelines to improve readability of the code and increase the number of automated checks to improve the overall code quality. `java` `rust` `javascript` `clippy` `intellij` `idea` `vscode`
- *Recruitment* – Write job descriptions for hiring new developers and perform the tech-level interviews.
- *Administration* – Evaluation and administration of collaboration and development tools and servers. `jira` `slack` `miro` `mediawiki` `github` `vscode`
- *Issue-management* – Keep track of bug reports and feature requests. Establish an automated changelog. `jira`
- *Customer support* – Occasional first level support for individual users to track down hard-to-reproduce issues. `remote support`

- *Vendor communication* – Communicate with browser vendors and specification committees to negotiate changes and ask for clarification about unclear implementation details. Keep track of the latest development in areas of activity. `mozilla` `google` `w3c`
- *Moderator support* – Keep a close contact to the community moderators and provide them with the necessary tools and technical support.
- *Open source contributions* – Work on open source projects to improve the quality/usability of required third party libraries. `github` `cpal`

2008 – 2018 Gothaer Systems GmbH `full-time`

Main tasks

- further development and re-implementation of the packaging and distribution software `delphi` `autoit` `c#` `java` `sftp`
- further development of the CRM software `java` `eclipse`
- software packaging and distribution
- administration of the test infrastructure (hard- and software) `networking` `file server` `vmware` `windows`
- 3rd level support of the software distribution `teamviewer`

Special skills

- permanent self-education
- constructive and creative solutions by applying very broad IT knowledge
- impart knowledge and documentation
- reverse-engineering

Remarkable achievements

- automation of manual tasks with respect to software packaging and distribution
- reduction of legacy technologies and introduction of modern software, hardware and processes
- reduction of support costs
- introduction of agile methods for project management

Other tasks

- reverse engineering of a legacy MS-DOS command line tool using a self-written disassembler; re-implementation in Java `java` `portable executable` `16 bit` `disassembler` `interpreter` `msdos`
- migrating a legacy file format into a modern representation `dbase`
- migrate a PHP application to Java
- eliminate the need for CD/DVD distribution by improving online processes and applications
- migrate decentralized customer information to a centralized CRM system using proper privacy, encryption and integrity measures `c#` `sftp` `encryption` `privacy` `fault tolerance` `integrity`
- further development of an Eclipse-RCP application for customer service `swt` `high-res` `java` `eclipse`
- develop a tool for stress-testing the server infrastructure `java`
- install a backup server for the test infrastructure `linux` `raid` `mdadm`
- analyze and solve false positives from malware scanners `autoit` `research` `c#`

- help other departments on issues with performance, memory-leaks, networking problems, encodings and browser compatibility `stability` `quality` `performance` `optimization`

Platforms

- Windows XP/Vista/8/8.1/10
- Debian Linux
- VMWare Workstation
- WebSphere Application Server

Software and Libraries

- Eclipse, MS VisualStudio, ReSharper, RAD
- TeamViewer, PoB, Jira, Confluence Wiki
- Subversion, MFT, UC4, SWT, InstallShield

Languages

- Delphi, C#, Java, x86 assembly, C, Visual FOXPRO, AutoIt, InstallShield-Script

2007 – 2008 Gothaer Systems GmbH `freelancing`

Tasks

- Migrate website to latest version of the FirstSpirit CMS; code cleanup to match new standards
- develop and apply BeanShell scripts to restore compatibility to older parts of the code
- Development and implementation of a Flash-based video player following the corporate design; used in inter- and intranet `flash` `video` `browser`
- Seamless integration of external applications by migrating from iframes to XML/XSLT/XHTML
- Teach the details and pitfalls of character set encodings and string escapings to the team.

2004 – 2007 kernpunkt/step one GmbH `full-time`

Tasks

- further development of the “step one Solution Server” `cms` `php` `apache` `ajax` `internet explorer` `mysql` `eclipse` `cvs` `subversion` `wysiwyg` `javascript`
- IT project management
- prepare and perform technical product presentations
- prepare and perform training courses for customers
- provide technical support for customers
- educate and supervise IT apprentices
- technical evaluation of CMS systems `drupal`
- further development and administration of several websites `php` `drupal`
- technical support and maintenance of internal frameworks and libraries

2001 – 2004 medial Gesellschaft für digitale Medien mbH `apprenticeship`

Tasks

- Implement product presentations and games on CD-ROM and DVD `director` `flash` `action script` `lingo` `photoshop`
- Develop and implement websites with basic CMS features `php` `ftp` `html`
- Perform and teach quality assurance

Platforms

- Microsoft Windows 95/98/2000/XP/ME/NT 3.5/NT4.0
- Suse Linux Open Exchange Server
- Apache web server

Languages

- HTML, XML
- JavaScript, ASP, PHP, Microsoft VisualBasic, Java (SE and ME)
- Microsoft Access, MySQL, MS-SQL

Products

- Macromedia Director, Macromedia Flash, Macromedia Shockwave
- Adobe Photoshop
- MS Office

Personal projects

- Install a DIY LED back-light to the TV and control them by the media center (Raspberry PI) `electronics` `i2c`
- Write a service for capturing the Raspberry Pi's video output to make the LED array match the picture's content. `videocore` `arm` `pwm` `i2c` `rust`
- Write a service to monitor the photovoltaic production and the power meters to make sure the pool pump is only switched on, when there's enough overproduction. `green energy` `smart home` `electronics`
- Build an automated BBQ to keep a stable temperature over several hours. A Raspberry Pi watches the room and meat temperature and a stepper motor is being used to adjusts the air inlet accordingly. The system can be controlled and monitored via smartphone using IRC. `because i can` `java` `irc` `rust` `adc` `stepper` `pid controller`
- Write an image stacker for astronomic photography to reveal dark details and compensate celestial movement. `image sensor` `bayer interpolation` `optical flow` `opencl` `java`
- Build an DIY auto-tracker (so called barn door) for long-term exposure of the night sky. `speed control` `mechanical`
- Generation and visualization of procedural landscapes. `visual basic` `c#` `opengl` `shader`
- Write a reader for various raw image file formats (CR2, DNG, TIFF, FITS) partially running on the GPU using Java and OpenCL. `memory mapping` `opencl`
- Write a music party bot for streaming charts and individual tracks to a Discord voice chat. `rust` `audiotool` `youtube-dl` `community`
- Run and administer a few Minecraft servers. `gaming` `lxc` `linux`