Will Cygan

wcygan.io@gmail.com | linkedin.com/in/wcygan | github.com/wcygan | wcygan.io

Skills

Languages: Java, Rust, Go, Python
Technologies: gRPC, Protocol Buffers, Docker, Kubernetes
Data Systems: PostgreSQL, Redis, Kafka, Samza, Spark, Hadoop

Experience

Senior Software Engineer, LinkedIn – Chicago, IL March 2024 – Present

- Architected a high-performance alerting system using **Kafka, Samza, and Venice**, enabling real-time invoice tracking and alerts at **50,000+ QPS** for LinkedIn's Global Alerts feature.
- Reclaimed **$2M+ in annualized revenue** by preventing involuntary churn through the Global Alerts system, contributing to LinkedIn's bottom-line growth and customer retention efforts.
- Engineered a Kusto-based exception summary dashboard, integrating access and application logs, **reducing incident triage time from tens of minutes to seconds** and enhancing oncall efficiency
- Led JVM optimization efforts, using A/B testing to **improve JVM health from 30-80% to 99.9%** across unhealthy production services.

Software Engineer, LinkedIn – Chicago, IL February 2022 – March 2024

- Contributed to the backend implementation of VYMBII (**Videos You Might Be Interested In**) for LinkedIn Learning courses on **linkedin.com/feed**. This *online* feature served videos at around 3,000 QPS scale.
- VYMBII leveraged ML models to **personalize video recommendations** for courses which were displayed in a carousel format **similar to TikTok**. Resulted in a **10%+ increase in course engagement**.
- Developed the *offline* flows (**Spark+HDFS**) for Learning Alerts, **a recommendation system** that classifies users based on job-seeking preferences and **delivers targeted course recommendations to 10M users weekly**.

Projects

Twote Social Media Platform [Rust, gRPC, Docker, ...] github.com/wcygan/twote

- **Architected a social media platform with microservices** using Rust and gRPC.
- Designed and integrated multiple databases (Postgres, Redis, MongoDB) for **efficient data management**
- Implemented **authentication and session management** using Redis for token caching

Concurrent Web Crawler [Rust, Tokio] github.com/wcygan/crawler

- Developed an asynchronous web crawler using Rust and Tokio. The application crawls web pages to collect and **index data while respecting rate limits** for each domain.
- Implemented a **concurrent architecture** using connection and parser pools for optimal performance
- Designed a key-based rate limiter to **prevent overloading target servers**
- Implemented graceful shutdown **mechanism for clean termination** of the crawler

Java Callgraph [Java, JaCoCo] github.com/bitslab/java-callgraph

- Developed a research project which **generates static call graphs for Java projects** using Java Reflection and various libraries including JGraphT, BCEL, and Reflections
- Implemented graph algorithms to **analyze and optimize call graphs**, including reachability analysis, pruning, and ancestry computation
- Created functionality to parse JAR files, extract method calls, and **construct directed graphs representing the java program callgraph**

Education

University of Illinois at Chicago – BS in Computer Science Fall 2021