

Trampas Kirk

trampas.kirk@gmail.com | LinkedIn: [trampaskirk](#) | Github: <https://github.com/cynoclast>

SKILLS/TOOLS

- **Languages/protocols:** scala, javascript, typescript, Java 8, bash, regular expressions, ~~TeX~~X, JMS, FIX, PL/SQL
- **Tools:** AWS (EC2, Dynamo, S3, Kinesis), memsql, postgres, scylla, skinny ORM, karma, Apache Avro, Netflix Tech Stack (Hystrix, Asgard, Ribbon, Eureka, Karyon, Archaius), gradle, Couchbase, git, UNIX/POSIX tools, memcached, maven, PowerMock/PowerMockito, Spring, Jenkins/CI/CD, Design Patterns™, IntelliJ, Hibernate, SVN, perforce
- **Operating Systems:** Linux, OS X, Unix, Windows

EXPERIENCE

Storygize, Inc.

Senior Software Engineer

remote

Jun 2021 - current

- **Engineering (scala, javascript):** Enhanced and supported a Real Time Bidding (RTB) Demand Side (Advertising) Platform (DSP) reaching hundreds of millions of users a day
 - * Created Insertion Order form generation with DocuSign integration allowing faster onboarding of new clients
 - * Enhanced content lifecycle management to handle failures at supply partners resulting in less noise
 - * Added event type audience supporting any user interaction, for example click, conversion, visit to target those users
 - * Created a monthly supplier usage report job that compiles and emails the report, replacing a manual process
 - * Implemented a metered inventory caching to greatly speed up inventory targeting estimates while building executions
 - * Built a JWT authenticated Zendesk Integration for help pages
 - * Added reporting capability by zip code, DMAs, demographics (advertising) performance data from data aggregation via spark RDD, to report generation
 - * Fixed numerous 'flaky' tests with configurable smart timeouts, resulting in far fewer failed builds

Mainz Brady Group / Nike. Inc

Senior Application Engineer

Beaverton, OR

Nov 2013 - May 2019 (3 contracts, 1 Fulltime)

- **Notifications Team (Java, microservices):** Enhanced and supported a suite of Karyon based microservices that send the majority of email/sms/push notifications to users
 - * Instigated a push for shorter more efficient standup meetings resulting in less time wasted and improved developer happiness
 - * Provided constructive criticism to the team/manager through proper channels resulting in team support for a simplified architecture
 - * Presented my .bash_profile collection of speed and information enhancements to my department of approx. 50 engineers and shared the the public repo with said department / Nike (available on github)
 - * Added BCP 47 (international language standard) support to microservices to support sending order and marketing notifications in 33 languages in 28 countries
 - * Reintroduced the rich domain model to a team working on a large, complex suite of services, resulting in more cohesive, easier to maintain code with faster testing cycles
 - * Introduced and enforced new integration testing standards that reduced future development and testing time and laughter among the team
- **Product Services (Java, microservices):** Nike Product Information in AWS
 - * Helped grow Nike's direct-to-consumer market by developing a suite of microservices in AWS based on Netflix's tech stack to reliably serve product data exceeding a million hits per minute across all services within a devops environment
 - * Enhanced and supported a non-cloud facade service that aggregates older, less user-friendly services to serve server-to-server, desktop, appliance, and mobile clients in a fast and consistent way
- **Social Team (Java, spring):** Enhanced and supported a monolithic spring application and ported functionality to AWS microservices
 - * Developed a micro-service based on Netflix's tech stack (Karyon, Ribbon, Hystrix, Eureka, Archaius) to serve business-critical user settings from either a Cassandra cluster, or a foreign micro-service using Ribbon
 - * Enhanced and supported and implemented numerous feature requests to a monolithic application built on Java, JBoss, Spring, JMockit to ensure continuity of business
 - * Built a program in a spike to analyze real world user data to create a memcached cache sized to minimize costs for caching selected data for all users and added that caching ability along with a feature toggle to the monolithic application

Citigroup, Global Derivatives Market

Senior Programmer/Analyst

Charleston, SC

Dec 2009 - Apr 2013

- **OCEAN:** Created a multi-threaded pseudo-producer-consumer product that provides realtime message reconciliation in conjunction with another department via hooks into the file system to achieve non-busy-wait monitoring, and dynamic, adaptive message resending capability, and provides an elegant solution to keep the firm in compliance with intraday FINRA reporting requirements for North American markets

- **CX TMSLink ACT Reporting recon:** Designed and delivered in two days a scripting product based entirely on common UNIX utilities from requirements to production in one week that allowed automatic reconciliation of order execution data between disparate systems automatically, complete with emailed report generation
- **LiquiFiCore:** Enhanced and supported a Dark-pool Crossing Engine
 - * Demonstrated understanding of an unstated need and created an inbound-message validation framework that that all existing validation was moved to it upon its release, and it was immediately adopted by another team within the organization
 - * Wrote a test harness using custom Matchers for the clearing data output stream that made testing, debugging, and all development on that stream easy; to quote my manager at the time, "Your RIO testing work rocks the hizouse"
 - * Wrote the clock that replaced all calls to retrieve system time with a custom clock that allowed the developer to convince the whole system into thinking it was any arbitrary time so that time-based events and rules like market open, market close, and others to be simulated at any time, with a safeguard against it being used outside of local development and test environments
 - * Introduced mock-object testing to a legacy project, complete with concrete, highly domain-relevant examples that allow other developers to easily follow suit with creating their own mock-based tests
 - * Modified the above project to build using maven within the continuous integration server, Jenkins, complete with automated static code analysis tool implementation and unit test coverage measurement that provides fast build failure notification and QA

BenefitFocus.com

Team Lead

Charleston, SC
Oct 2005 - Dec 2009

- **Electronic Data Interchange (EDI):** Managed a team of six developers
 - * Provided feedback on performance, and mentoring to less experienced developers
 - * Acted as the point of contact for the team and attended organizational/planning meetings to reduce interruptions to development
 - * Assigned tasks based on developer strengths, weaknesses, and maintaining effective cross-training with input from the team
 - * Addressed grievances I was empowered to handle and escalated those I was not
 - * Performed estimation for the project and tasks within it
 - * Organized one-on-one meetings with each member to discuss performance and allow them to discuss confidential information or provide feedback
- **Senior Software Engineer (EDI):** Enhanced and supported the project responsible for all meaningful Output (I/O) from the company
 - * Spearheaded a company-wide plan for improving code quality, requirements, the estimation process, and automated build processes, leading to more rapid development on more readable, maintainable, reliable code
 - * Introduced unit tests and EasyMock testing to a legacy codebase, providing a guide complete with relevant, concrete examples that improve code quality and unit test coverage while allowing new developers to easily follow suit
 - * Coordinated efforts between engineers and configuration management leading to a streamlined build and deployment process for mission critical business components
 - * Worked with business stakeholders to design and implement complex data translation software incorporating Java, Oracle, Python, XML, and XSLT to process millions of benefits records on a daily basis

MSIS Inc.

Software Engineer

Dahlgren, VA
Fall 2004 - Fall 2005

- **Supported the Aegis Weapon System for the Navy:** Worked on various projects some of which were classified secret
 - * Designed, implemented and maintained two multi-system software projects from back end data extraction via memory dump reads and shell scripts to storing, processing, and retrieving this information using extremely complex queries to calculate maximum simultaneous users counts for all software in use within one Department of Defense site
 - * Improved the GUI for the simultaneous user tracking program to provide a faster and more helpful interface to the user
 - * Customized University of Michigan's radmind project for use by the DoD, including bug fixes and performance enhancements leading to 100% improvement in system restore speed
 - * Designed and implemented a simultaneous usage tracking algorithm across Unix and OpenVMS systems allowing the client to reduce the number of licenses to actual simultaneous users

King Brothers Fabricating Inc.

Structural Detailer

Parkersburg, WV
Summer 1999 - Fall 2000

- **Computer Aided Drafting (CAD):** Created detailed structural shop drawings from architectural drawings in AutoCAD for in-house fabrication and erection

EDUCATION

West Virginia Wesleyan College

Bachelor of Science, Computer Science

Buckhannon, WV
Summer 2000 - Spring 2004

PROJECTS AND HOBBIES

- Reading
- Self publishing creative works on Amazon
- Running (including the 2012 Tennessee Ragnar)
- Martial arts like Karate, boxing, Muay Thai, MMA, and Jeet Kune Do, aquatic HIIT and more
- Hundreds of hours of archery with my 55# (draw weight) recurve bow
- Built a small, fast, and simple program to calculate my weekly timesheet tracking and entry and when to leave on Friday
- Moved my \LaTeX resume to github (this one)
- Moved my bash profile (wizard's staff) to github
- Contributed a patch containing heap size enhancements to the coberuta maven plugin project