Max Ellis

+1 (360) 356-2304 | maxjordanellis@gmail.com
linkedin.com/in/max-ellis-cs m-j-ellis.github.io/
Camas, WA 98607, USA

Education

Master of Science, Computing Science, University of Alberta, Edmonton **June 2022**

- Advisor: Sarah Nadi
- GPA Overall: 4.0 / 4.0

Bachelor of Science, Computer Science, Washington State University, Vancouver **May 2019**

- GPA Overall: 3.92 / 4.0

Technical Skills

Languages:	Java, Spring, SQL, Python, Javascript
Libraries:	JDBC, Snowflake SDK, AWS Java SDK, Pandas, NumPy, MatPlotLib
Tools:	Terraform, Snowflake, PostgreSQL, Kubernetes, AWS services, Github, IntelliJ IDEA, Ansible, Puppet

Work Experience

Software Engineer **Act-On Software, Portland** **November 2022 – Present**

- Stabilized Act-On's data lake service by implementing Snowflake deferred merge strategy, eliminating errors (previously 800 per hour), reducing average response time from 8 minutes to 300 milliseconds, and eliminating data duplication
- Engineered high-throughput data pipelines leveraging Spring Cloud Dataflow and Snowpipe Streaming to ensure low-latency data loading for large data volumes
- Built Snowflake and AWS infrastructure with terraform and puppet to set up resilient and scalable services
- Upgraded an integral service from Java 8 to Java 21, developing a rollout plan to mitigate risk and an in-depth test plan to verify expected behavior with downstream dependencies
- Championed Customer Support escalations, allowing the team to focus on core tasks while improving customer experience

Research Assistant **University of Alberta, Edmonton** **September 2019 – June 2022**

- Spearheaded a project with an external collaborator to revitalize operation-based refactoring-aware merging, allowing it to be applied in practice with Git
- Emulated double dispatch in Java to make operation-based merging feasible to scale and maintain, reducing the effort required to update supported refactorings or add new refactoring types
- Analyzed experimental data utilizing Python libraries to compare the strengths and weaknesses of two refactoring-aware merging approaches, providing insights and paths forward for each approach
- Leveraged sparsely documented third party libraries to programmatically perform refactorings and detect refactoring-related merge conflicts, supporting 17 known refactoring type

Selected Projects

Custom Objects (2024 – Present). Developed a sophisticated data pipeline that took advantage of Snowpipe streaming to ingest data, perform complex merges and data processing on records, and export the data to Kafka topics for consumption
RefMerge (2019 – 2022). An operation-based software merging approach that considers the semantics of refactorings to improve the merge resolution process. Reduced unnecessary conflicts by 25% while eliminating false negatives reported by Git

Publications

Max Ellis, Sarah Nadi, and Danny Dig. "Operation-based Refactoring-aware Merging: An Empirical Evaluation". In: *IEEE Transactions on Software Engineering* (TSE 2022)
Preprint: arxiv.org/pdf/2112.10370.pdf

References Available on Request