Joe Bylund

joseph.bylund@gmail.com
347-829-5863

Kensho

Cambridge, MA
Data Engineer
2018 — Present

- Designed and implemented multi-step document processing pipeline using multiple in-house ML services.
- Designed and built speech-to-text alignment pipeline using SQS & gentle forced aligner.
- Implemented a number of checks in github hooks (pylint, flake8, mypy...), improving the developer experience.
- Migrated fuzzy company identification service to kubernetes and optimized performance.

Moat

New York, NY
Senior Data Scientist & Back-end Engineer
2013 — 2018

- Designed and implemented distributed, fault-tolerant ETL, reducing cost by an order of magnitude, increasing reliability, and reducing processing time from ~10 hours to ~1 hour, making data available to clients far earlier in the day (using python, SQS, Redis, PostgreSQL).
- Numerous data-driven API improvements which lead to 3-4x improvement in API latency as well as maximum request size (PHP, CakePHP).
- Contributed improvements to ORM (CakePHP) and core PHP in order to reduce the number of queries necessary to render a page by 5x (decreasing page load time by ~3x) (CakePHP, c).
- Standardized deployment framework used to deploy thousands of servers of ~30 different roles to AWS (AWS, EC2, boto3).
- Migrated primary non-statistical database (users, accounts, metadata) from MySQL to PostgreSQL improving uptime & flexibility (MySQL, PostgreSQL, foreign data wrappers).
- Migrated primary statistical database (500 million rows/day) from non-first normal form to first normal form schema, improving query latency, reducing storage demands, and increasing throughput.
- Architected and implemented sophisticated message routing system which is responsible for moving ~40 billion events per day from our pixel servers to our real time processing servers while balancing CPU and memory constraints (c++).
- Architected and prototyped massively parallel decentralized data lake using AWS lambda and S3 for cost-effective storage and low-latency and cost-effective queries (AWS lambda, python, PostgreSQL).

Columbia University

New York, NY
Doctor of Philosophy
2007 — 2013, GPA 3.9/4.0
Integrated Program In Cellular, Molecular and Biomedical Studies
Monte-Carlo Sampling of Protein-Ligand Interactions and Computational Improvements to Implicit Solvent Models

- Lead developer and maintainer (~80% of source commits) of Protein Local Optimization Program (PLOP) project, a molecular mechanics library developed at Columbia University, University of California at San Francisco, and Schrödinger (fortran).
- Designed and implemented the computational mutation scanning module of PLOP.
- Rehabilitated project from non-compiling state on arrival.
- Redesigned build system to automatically determine dependencies and take advantage of parallel compilation, reducing build time from ~30 minutes to ~3 minutes and greatly accelerating development.

- Created a small molecule database representing 95%+ of small molecules in the Protein Data Bank, extending PLOP from a protein-only program to a general molecular mechanics toolkit.
- Designed and implemented a Perl based automated regression testing framework, which accelerated development while minimizing bugs and regressions.
- Created a project wiki, combining scattered documentation and completing missing documentation.

Rice University

Houston, TX

Bachelor of Arts - Mathematics
2003 — 2007, GPA 3.7/4.0
Bachelor of Science - Ecology and Evolutionary Biology
Relevant coursework: Machine Learning, Ordinary and Partial Differential Equations, Real and Complex Analysis, Combinatorics, Number Theory, Mathematical Logic, Modern Algebra, Euclidean and Non-Euclidean Geometry.

- Designed and implemented two methods of combining information from multiple proten crystal structures into a single “composite motif”. These combined motifs increased sensitivity and specificity of motif matching algorithms.
- Completed senior thesis project identifying homologous pseudogenes in human and chimpanzee, and determining differential mutation rates.

Open Source Contributions

Python Package Installer - Pip

- Up to 10x improved performance of package version resolution (pull request).
- 2x performance increase in comparison of tag objects (pull request).

These changes decreased the time spent on pip steps as part of ci process (at Kensho).

PHP

Avoided roundtrip to database in order to get column type for most common datatypes (pull request). This decreased the number of queries run as well as page load time by more than an order of magnitude in some cases (at Moat).

Gnome Shotwell Photo Manager

- Decreased the number of times raw images were decoded during import process, improving photo import performance.
- Recursively included contained files in the folder browser.
- Added support of panoramic images as event thumbnails.
- Updated searches to search comments and robustly treat accented characters.
- Fixed a number of UI experiences such as adding icons to buttons and windows, and correcting misleading text.

Technologies & Skills

- Extensive experience with AWS services and apis (EC2, dynamodb, kinesis, RDS...)
- Python, C++, shell (and previously FORTRAN, vala)
- PostgreSQL, kubernetes, MySQL, Vertica, redis, git
- Kafka, RabbitMQ, SQS

Last updated December, 2021