Ryan Spangler

℘ +1 (503) 781 3891
✉ ryan.spangler@gmail.com
⌂ elephantlaboratories.com
 prismofeverything
in ryankspangler
 elephantnahpele

Discovering Patterns in Connections

Summary of Qualifications

○ Software engineer specializing in computational infrastructure for biological modeling and data.
○ 13 years of professional experience working in cancer genomics, proteomics, whole-cell modeling, workflow development and graph database design and implementation.
○ Interested in working computational frameworks for biological discovery, scientific research and applications for biologically inspired algorithms.

Experience

2018–present   **Software Engineer**, *Stanford Bioengineering - Covert Lab*, Stanford CA.

I coordinate the efforts of the whole-cell modeling team with graduate students, postdocs and PI's, and in collaboration with other institutions. Beyond the code, I provide computational support, design consultation, implementation strategies and pair programming mentorship. In addition, I lead a consistent and productive practice of code reviews, pull requests, continuous integration, version control and other software engineering best practices.

Achievements:

○ Created a platform for distributed multi-scale modeling of whole-cell colonies in a shared molecular environment, with a websockets-based interactive visualization: http://github.com/CovertLab/vivarium
○ Built a workflow and execution system on Google Cloud to run whole-cell simulations and whole-colony experiments: http://github.com/CovertLab/sisyphus
○ Implemented a generalized version of the Gillespie algorithm in C to get around a critical performance bottleneck: http://github.com/CovertLab/arrow
○ Worked with the experimental teams to solve challenges of computational scale and organization for imaging and microscopy data.

2016–2018   **Software Engineer**, *OHSU - Computational Biology*, Portland OR.

At OHSU I worked with scientists and engineers to build on infrastructure and analysis pipelines to support all of the different research efforts at the University. My main focus was the development of a graph database for cancer genomics to provide a means to query, analyze and visualize this data as a whole.

Achievements:

○ Engineered a large graph database system that automatically transforms and integrates all incoming data into a single graph http://bmeg.io/
○ Created a schema to encode queries themselves as data so they can be programmatically generated, optimized and processed.
○ Created an interactive visualization for navigating and querying the live graph.
○ Engineered a distributed event system to trigger pattern-discovery analyses as data streams into the system.

2014–2016 **Lead Developer**, *Little Bird Technologies*, Portland OR.

At Little Bird I took their mass of social network data and apply graph theoretical, statistical and machine learning methods to find patterns and draw conclusions from that data.

Achievements:
- Built a 3d network visualization to explore and interact with vast, interconnected data.
- Used bayesian networks and random forest to build a classification system of user behavior.
- Released much of the infrastructure as open-source: https://github.com/littlebird
- Instituted a workshop for collaboratively improving the whole team's coding and software development skills, starting by implementing well-known graph algorithms.

2007–2014 **Senior Developer**, *Instrument*, Portland OR.

I worked with the labs team to transform concepts through code into practical applications.

Achievements:
- Created Caribou — an open source Clojure web ecosystem for building large high-performance web applications with great alacrity. https://github.com/caribou
- Built Schmetterling — a browser-based debugger for inspecting running Clojure programs: http://github.com/prismofeverything/schmetterling
- Pioneered a weekly workshop for collaboratively learning 3D graphics programming:

2006–2007 **Programmer**, *Performance Logic*, Portland OR.

I learned the fundamentals of real world development using C++ while simplifying and modularizing a large legacy code base.

Achievements:
- Built a variety of visualization methods for generating reports from large data sets
- Enhanced the custom scripting language with features from functional programming

Education

2009–2012 **Master of Science, Systems Science**, *Portland State University*, Portland OR.

Computational Neuroscience, System Dynamics, Information Theory, Agent Based Systems

2002–2005 **Bachelor of the Arts**, *The Evergreen State College*, Olympia WA.

Math, Performance, Computer Science

1999–2001 **Undergraduate Study**, *Oberlin College*, Oberlin OH.

Cognitive Science, Computer Science

Technology

Languages Clojure, Scala, JavaScript, Python, C/C++, Julia, Java, Bash, SQL

Tools Kafka, Docker, Postgresql, RabbitMQ, Mongo, Git, Unix, Emacs, GLSL

Areas Biological Modeling, Network Science, Graph Databases, Machine Learning, Data Visualization, Generative Music

Interests

Biology Molecular Biology, Cell Biology, Systems Biology: How does life work? How can these principles be discovered and applied to other realms?

Music Piano Tuning, Music Theory, Performance: Exploring the space of all possible musical spaces and relationships.

Games I have designed and published an award-winning board game - Sol: Last Days of a Star. http://elephantlaboratories.com/sol