Ryan Spangler

Curriculum Vitae

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

Patterns in Connections

Education

2009–2012 **Master of Science, Systems Science**, *Portland State University*, Portland OR.
Computational Neuroscience, System Dynamics, Systems Modeling, 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

Experience

2018–present **Software Engineer**, *Stanford Bioengineering - Covert Lab*, Stanford CA.
At Stanford I work daily with researchers, students and postdocs to translate biological mechanisms and dynamics into runnable code, expanding on and integrating the effort of all of the lab members into a working whole-cell model. In addition, I conceive, develop and maintain the computational infrastructure and tools required by the various efforts around the lab.

Achievements:

- Created a platform for distributed multi-scale modeling of composable whole-cell simulations that exchange resources with a shared molecular environment, with a websockets-based interactive visualization: `http://github.com/CovertLab/Lens`
- Worked with graduate students and postdocs to provide computational support, design consultation, implementation strategies and pair programming mentorship and feedback.
- Built a workflow and execution system on Google Cloud to address the specific computational challenges of the lab: `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`
- Optimized a complex parallel polymerization algorithm and expanded it to enable variable elongation rates.
- Lead a consistent and productive practice of code reviews, pull requests, continuous integration, version control and other software engineering best practices.

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

At OHSU I work with scientists and engineers to determine what infrastructure and analysis is needed to support all of the different research efforts at the University. My main focus is 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 a visualization framework to synthesize our various visualization methods into a general and reusable package.
- 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 take 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.
- Open sourced much of the infrastructure that powers the application: 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 invent constantly — transforming 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
- Created Cyclops — a tool for interpolating data for use in programmatically driven animations: http://weareinstrument.com/cyclops
- Built Schmetterling — a browser-based debugger for inspecting running Clojure programs: http://github.com/prismofeverything/schmetterling
- Pioneered a weekly workshop for collaboratively learning 3D 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

Interests

Biology Molecular Biology, Cell Biology, Systems Biology: How does life work? How is this possible?

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

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

References

Name	Contact
○ Markus Covert *Stanford Bioengineering*	(650) 725-6615 mcovert@stanford.edu
○ Jerry Morrison *Allen Center*	(650) 488-3541 jerry.morrison@gmail.com