mbp@geomtech.com
828-338-9489

Mark Phillips

github.com/embeepea
embeepea.me

I am a software engineer with a passion for mathematics, visual understanding, and building production software systems. I've held multiple positions at internet startups, research labs, and in academia, and have helped found two companies. I have been responsible for product and code development and deployment in all of those positions.

Most of the programming I have done recently has been in JavaScript, both server-side and client-side, but I am comfortable in many languages and have done large projects in Python, Java, C, and C++. I have experience writing geospatial and other data analysis code in Python and Java, and I love functional programming and have also done projects in Clojure and ClojureScript. More details about my projects are available at embeepea.me.

SKILLS AND INTERESTS

- 8 years experience in JavaScript, including framework engineering, Node.js, and library development
- Significant scientific, geospatial, and data science programming experience in Java and Python using NumPy, Pandas, GDAL, NetCDF
- Significant experience in functional programming (Clojure, ClojureScript)
- Strong understanding of machine learning algorithms and theory
- 25 years programming experience in Java, Python, Ruby, C, C++
- Expert in fast interactive computer graphics applications, both 2D and 3D; some WebGL
- Thorough command of the math of computer graphics (linear algebra, computational geometry)
- Extensive experience leading small teams of 3-5 people in software development projects
- 15 years SQL experience, including spatial (PostGIS)
- Significant experience with unit testing and build automation (JUnit, Ant, Jasmine, Grunt, Make)
- 25 years Linux/Unix experience including scripting in Perl, Python, and shell; some knowledge of DevOps, including Vagrant, Puppet, Amazon AWS
- Significant experience in higher education and government research labs
- 5 years experience with mathematical computation systems (Mathematica, MATLAB, Octave)

EMPLOYMENT HISTORY AND WORK HIGHLIGHTS

2015-present	FernLeaf Interactive	CTO / Co-Founder

- Led software development at a small business spun out of university research center. Developed several web-based environmental data visualization prototypes using JavaScript, HTML5 Canvas, Node.js, and MongoDB, and the data services back-ends to support them.
- Designed and implemented an Extract/Transform/Load (ETL) system for a data warehouse that maintains a database of over 6 billion environmental data records in a format optimized for rapid access and visualization. The system downloads and processes over 2 million new records each day. It's written in a combination of Python using the Pandas library, Clojure for concurrent downloading of thousands of documents, and C for fast parsing of ASCII data files.

- Designed and wrote a web-based interactive watershed map (watersheds.fernleafinteractive.com, source on GitHub) showing how water flows across land in the US. It immediately shows all the area that is both upstream and downstream from any location in the United States as the cursor moves over the map. It is written in JavaScript using Leaflet.js and TopoJSON and uses HTML5 Canvas rendering to provide very fast interactivity. It works in any modern desktop or mobile web browser. I presented this at !!Con 2016; a video of that presentation is on YouTube. I will also be presenting this at StrangeLoop 2016.

| 2007-2015 | National Environmental Modeling and Analysis Center (NEMAC) University of North Carolina Asheville | Senior Research Scientist |

- Led a team of 4-7 web developers in creating several climate and environmental science-based interactive web sites and visualization applications for federal, state, and local government agencies using a combination of Java, Python, JavaScript, PHP, PostGIS, MySQL, and Linux server technologies. As architect and lead developer I trained and supervised the work of other staff and student employees. This group was self-funded through government contracts which I was instrumental in securing.
- Principal architect and author of Multigraph, an open source JavaScript library for creating interactive two-dimensional X-Y data plots in web pages. It can read data in a variety of formats and provides interactive pan and zoom capabilities. Multigraph can read data from a web service, which allows it to be used to "surf" through large datasets, downloading only those the parts of the data that are needed for display. Multigraph is used extensively on NOAA's National Centers for Environmental Information web site and on the NOAA's climate.gov site. It's an open source project (source on GitHub) with documentation and examples available at multigraph.org.
- Did the data analysis and software development for the *Climate in Your Location* application to be rolled out later in 2016 with a new version of NOAA's US Climate Resilience Toolkit web site. The application allows users to view interactive time series data plots showing projections of temperature and precipitation for their specific location under several future emissions scenarios. I aggregated gridded climate projection (CMIP5) data across 3233 US counties, computed statistical time series for each county using Python Pandas (source on GitHub), developed interactive graphs using JavaScript and Multigraph, and packaged it into a documented API (source on GitHub) for use by the project's front-end developers.
- Architected and led the coding for the Climate Explorer application (source on GitHub) that is part of NOAA's US Climate Resilience Toolkit web site. The application, written in JavaScript using OpenLayers and Multigraph, lets users interactively explore the climate history of their location. It includes a Java back-end ETL system that maintains a database of over 100 years of daily climate data from about 40,000 locations in a format that is optimized for rapid download and visualization.
- Wrote the *Global Climate Dashboard* application (source on GitHub) for the home page of NOAA's climate.gov site to help users visualize the relationships among global temperature, CO2 levels, sea level, and a dozen or so other data sets that measure of how the earth's climate has changed over the last century. The application is in JavaScript, uses Multigraph, and includes an administrative interface (source on GitHub) using PHP and Drupal that NOAA staff uses to keep the data updated on a regular basis.
- Architected and led the development of the Forest Change Assessment Viewer application for US Forest Service's Eastern Forest Environmental Threat Assessment Center, which shows a weekly

updated view of detected changes in the US forest cover derived from MODIS NDVI satellite data. The user facing front-end is written in JavaScript using the OpenLayers map library, and the system includes a Python back-end that uses the GDAL and NetCDF libraries to process and archive new raster data sets which Forest Service staff upload each week, and MapServer and PostGIS to provide RESTful web services which support the application.

- Set up a cluster of Linux EC2 instances in Amazon Web Services and transitioned all of NEMAC's production web sites, applications, and databases into this cluster. This involved setting up a version-controlled provisioning system with Puppet, establishing a deployment process, and training the staff in that process.
- Worked with one other developer at NOAA's National Centers for Environmental Information to co-develop the [Severe Weather Data Inventory](#), a regularly updated data archive of severe weather records for the US. This involved writing several Java-based programs for extracting data from unstructured weather report texts, inserting the data into an Oracle database, and providing web-based search and bulk download capabilities.

2002-2007	NOAA National Centers for Environmental Information, Asheville NC	Software Engineer

- Developed a Perl-based ETL system for receiving satellite transmissions of data from [NOAA's Climate Reference Network](#) monitoring stations, processing it, storing it in an Oracle database, and making it available for download. I also developed a Java-based web site to give the public access to download and view interactive graphs and reports of the data.
- Used Java and Spring Framework to developed a flexible metadata storage and management system ([ISIS](#)) for weather and climate monitoring stations, providing for entry, viewing, and storage of arbitrary metadata schemas, and tracking and display of the history of metadata values over time.

2001-2003	Self-Employed	Consultant

- Wrote the C++ driver that the open source [GDAL](#) library uses to read vector data files in the US Census Bureau's Tiger/Line data format.
- Developed an interactive contest registration system using PHP and MySQL for the [Mathematical Contest in Modeling](#), a college level mathematics contest. Teams of students and their advisors register online, and contest staff use a secure administrative back-end to obtain lists of participants, and to communicate with them. Participants use the web site to submit information about themselves, and to obtain contest results.

1998-2001	Geometry Technologies, Inc	CTO / Co-Founder

- Was one of the principal developers of WebEq, one of the first software products to support mathematics equation notation in web pages, and the very first program to support the MathML standard. Two colleagues and I wrote WebEq at the University of Minnesota Geometry Center, and founded Geometry Technologies for the purpose of commercializing WebEq. Geometry Technologies eventually sold WebEq to Design Science, the makers of [MathType](#). WebEQ was a Java applet that rendered math equations inline in HTML documents.

1991-1998	The Geometry Center	Research Associate
	University of Minnesota	

- Wrote mathematical visualization and computation software in C, Mathematica, Java, C++, and Perl; led workshops and trained faculty and students in mathematical visualization techniques; made several computer-generated animations of mathematical concepts.
- Was one of the three principal developers of Geomview ([source on GitHub](#)), an interactive 3D geometry visualization program in C/OpenGL that can display 3- and higher-dimensional objects in Euclidean, hyperbolic, and spherical space. After the Geometry Center closed in 1998, I set up the [geomview.org](#) web site and transitioned the program to the open source community. Geomview still has an active following and is included in most Linux distributions.

1990-1991	Department of Mathematics and Computer Science	Assistant Professor
	University of Richmond, Richmond VA	

- Taught business mathematics, calculus, and linear algebra

EDUCATION

- PhD in Mathematics, University of Maryland
- MS in Applied Mathematics, Rensselaer Polytechnic Institute
- BS in Mathematics, Magna Cum Laude, Davidson College

PUBLICATIONS

- (with Steve Ansari, Chad Hutchins, Stephen DelGreco, and Nina Stroumentova), *The Weather and Climate Toolkit*, Proceedings of the 89th Annual Meeting of the American Meteorological Society, January 2009.
- (with Stephen DelGreco and Steve Ansari) *Touch Table Supported Climate Services and Applications Developed at NOAA's National Climatic Data Center*, Proceedings of the 89th Annual Meeting of the American Meteorological Society, January 2009.
- (with Steve Ansari and Stephen DelGreco) *A Geospatial Database and Climatology of Severe Weather Data*, Proceedings of the 88th Annual Meeting of the American Meteorological Society, January 2008.
- (with Silvio Levy and Tamara Munzner) *Geomview: An Interactive Geometry Viewer, Notices of the American Mathematical Society*, October, 1993, pp 985-988.
- *Dirichlet Polyhedra for Cyclic Groups in Complex Hyperbolic Space*, Proceedings of the American Mathematical Society, Volume 115, Number 1, May 1992.
- (with Charles Gunn) *Visualizing Hyperbolic Space: Unusual Uses of 4 x 4 Matrices*, Proceedings of the 1992 Symposium on Interactive 3D Graphics, ACM SIGGRAPH, March 1992.
- *Mathematica as an Interface to an Interactive Mathematical Graphics Program*, The Mathematica Journal, Vol 1 No 2, Fall 1990.
- (with G. M. Odell) *An Algorithm for Locating and Displaying the Intersection of Two Arbitrary Surfaces*, IEEE Computer Graphics and Applications, Vol. 4 No. 9, September 1984, pp 48-56.

SELECTED TALKS AND CONFERENCES

- *An Interactive Web Visualization for Watershed Topology,* StrangeLoop, upcoming in Sep 2016.
- *Upstream/Downstream: Discovering and Displaying Watershed Topology*, !!Con, May 2016.
- *The Use of Geovisualization in Presenting Weather and Climate Data to Local Decision Makers*, Annual Meeting of the American Meteorological Society, January 2009.
- Workshop on Climate Change for Mathematicians, Mathematical Sciences Research Institute, Berkeley CA, July 2008.
- *A Geospatial Database and Climatology of Severe Weather Data*, Annual Meeting of the American Meteorological Society, January 2008.
- *Computer Graphics as a Visualization Tool in Mathematics,* Workshop on Systems Administration: The Mathematician's Perspective, Geometry Center, December 1994.
- *Software tools for Visualization at the Geometry Center*. SIAM Annual Meeting, San Diego, July 1994.
- *Interactive Real-Time Software for Mathematical Visualization*; co-organized this special session, with George Francis and Norbert Quien, at the joint meeting of AMS and German Mathematical Society, Heidelberg, Germany, October 1993.
- *Geomview: An Interactive Geometry Viewer*. Experimental Mathematics seminar at University of Bonn, Germany, October 1993.
- *Visualizing Non-Euclidean Geometry*. DEC Systems Research Center, Palo Alto, CA, April 1993.

PROGRAMMING LANGUAGES AND SYSTEMS

JavaScript, Python, Java, Clojure, ClojureScript, PHP, Perl, C, C++, Scala, Ruby, Fortran, Mathematica, MATLAB, ActionScript, Bash, Amazon AWS/EC2, Linux, Node.js, jQuery, d3.js, OpenLayers, Leaflet.js, HTML, XML, UMN MapServer, GDAL, Apache Web Server Configuration, Drupal, Vagrant, Puppet, Spring Framework, Git, Subversion, MySQL, Oracle, MongoDB, PostgreSQL, PostGIS, OpenGL, Ant, Eclipse, Make, Emacs, npm, Grunt, Gulp, JUnit