

Michael J. Salib

Cambridge MA

msalib@alum.mit.edu
<http://michael.salib.com/>

Interests Distributed systems, compilers, database internals, machine learning, geospatial data processing, and scientific computing.

Education Master of Engineering **Massachusetts Institute of Technology**
Cambridge, MA May 2003 – May 2004

Master of Engineering in Electrical Engineering and Computer Science. My thesis was entitled *Starkiller: A Static Type Inferencer and Compiler for Python*.

Bachelor of Science **Massachusetts Institute of Technology**
Cambridge, MA September 1996 – May 2003

Bachelor of Science in Electrical Engineering and Computer Science. I left MIT for three years to work in industry.

High school student **Christian Brothers Academy**
Lincroft, NJ 1992–1996

Experience Principal Software Engineer **Climacell**
Boston, MA September 2018 – Present

I'm responsible for the design and operation of Climacell's flagship product, a terrestrial microwave precipitation sensing system. This system measures precipitation in real time around the world by sensing changes in microwave signal attenuation between cellular phone towers. It provides high resolution and extremely low latency precipitation measurements that serves nearly 1.5 billion people. That system drives other products including floor modeling.

As a Principal Engineer, I productionized research and designed experiments for system wide performance improvements. In my role as one of the early backend engineers at Climacell, I took the initiative and instituted automated software testing as well as a systematic code review in order to raise code quality and increase development velocity. I also consolidated many isolated software repositories into a monorepo so that we could share code between different components. In addition, I introduced automatic deployments to reduce errors. I mentored junior engineers and data scientists and gave talks on software design to our staff scientists.

Software Engineer in Tools and **Google**
Infrastructure
Cambridge, MA August 2016 – September 2018

I designed and built tools to improve understanding for large scale machine

learning models within Google. These tools operate on large distributed data sets. I also performed research on how to improve reliability and robustness for large scale ML systems at Google: See <https://goo.gl/xPyL8t> or *Whats your ML test score? A rubric for ML production systems.*

Software Engineer
Cambridge, MA

Facebook
February 2016 – May 2016

I designed and built a large scale distributed geospatial database engine using modern C++. The engine used Facebooks internal key-value stores for storage and processed queries against arbitrary geometry predicates. I designed it to employ vertical query processing so as to minimize interpreter overhead.

Senior Software Engineer
Cambridge, MA

MediaMath
February 2013 – May 2015

When MediaMath acquired the ADS division from Akamai, I went with them. I spent the first six months transitioning our user modeling, scoring and targeting pipeline from running on Akamai systems to MediaMath clusters in Amazon EC2. This involved tearing through eight years of accumulated code written in Perl, C++, three different SQL dialects throwing most of it away while replacing or keeping the rest.

After the migration, I lead a team of three software engineers building a segment sizing service for MediaMath’s new Data Management Platform. This is service that quickly (in under one second) estimates the size of segments, or user populations on the internet. For example, one segment might be “the set of all users who have visited any of these 12 web sites in the last year and have not made an offline purchase in these stores and have total sales exceeding \$80 or who live in zip codes where the median income is over 80% of the national median.” Regardless of whether the segment comprises 7 users or 70 million, the sizing service still returns an answer in under a second. I also helped design the data storage system that this service relies on; it handles data for several hundred million users.

Senior Software Engineer
Cambridge, MA

Akamai
September 2010 – February 2013

I implemented a streaming parallel data processing language called Trecul using LLVM and Hadoop. It allows users to specify queries and data transformations against distributed data sets in a Hadoop cluster and substantially outperforms tools like Hive while being far less verbose than writing raw Hadoop programs in Java. In addition, I designed and implemented most of the backend data analytics code for running the Advertising Decision Systems business unit at Akamai. This organization held data on several hundred million users and my code traversed that data set every day.

Software Engineer
Newton, MA

Trip Advisor
February 2009 – May 2010

I designed and implemented a scalable system for continuously monitoring air travel prices. This system strove to find the best prices satisfying user specified travel endpoints and trip duration. Since issuing air fare queries costs money, the system incorporated optimizations to modify the query frequency based on the popularity of different routes, market volatility, and system load. The system has been operational and serving users since the summer of 2009.

In addition, I participated in legacy systems refactoring and modernization. Trip Advisor has accumulated an extremely large Java code base since 2000 that was in dire need of improvement. Finally, I designed and implemented a back-end analytics system to support lead and impression analysis in the vacation rental market. This system integrated data from several sources into a data warehouse that I designed. The goal was to provide custom reporting to business development staff who were constantly adjusting marketing and presentation in order to expand the business.

Software Engineer
Cambridge, MA

ITA Software
August 2005 – January 2009

I designed and implemented a high performance high availability software router for processing airline messages using custom airline industry protocols. As the project grew, I supervised the small group of developers who contributed. ITA is building an airline reservation system to replace large centralized mainframe systems at the heart of every airline. It processes tens of thousands of messages per second and it must remain continuously available for the airline to function: the message router is needed for almost every aspect of reservation operations including making reservations, ticketing, check ins, and flight tracking. In my role as technical lead, I was responsible for liaising with other groups and customer relationship managers as well as upper management. In order to meet the stringent performance requirements, the router made use of asynchronous networking, multi-threading, a predicate-based rule engine, and Oracle Advanced Queuing.

In addition, I designed and implemented the support needed to make ITA's airline reservation system process IATCI messages, which are used by airlines to check in passengers on flights operated by other airlines. This involved digging through documentation, discussion with the client and internal stakeholders, production of significant design documents, a large test suite, and a great deal of application code. IATCI is needed to check-in passengers, process their baggage, retrieve seat maps, validate passports, and check flight status.

I was also responsible for integrating ITA's global distribution system with a major new carrier. This required significant efforts at deciphering requirements from vast heaps of outdated customer documentation, extensive modifications of original the code base, continuous discussions with a quality assurance team and regular meetings with the customer. I also added support for entirely new messaging features such as real time flight information. I was also involved in performance tuning relating to our Oracle database and our object relational

mapper.

Software Engineer
Cambridge, MA

Divmod
October 2004 – July 2005

I worked with a team of engineers to build a scalable high performance networking service. This service integrated email, voice, instant messaging and remote data stores while providing full text indexing and a variety of other services. I designed and implemented parts of this system in addition to reviewing code written by other developers and responding to customer requests by finding and resolving bugs. The system integrated users' communications into one application, not unlike Google does today.

Research Assistant
Cambridge, MA

MIT CSAIL
May, 2003 – May 2004

I designed, built, and tested a wireless sensor network used for human tracking in urban environments. As part of this work, I coauthored a paper in the International Journal of Telecommunications.

Student Researcher
Cambridge, MA

**MIT AI Lab Dynamic Languages
Group**
May 2001 - September 2001

I designed and built a static type inferencer for the Python language. This work eventually led to my Master's thesis.

Software Engineer
Cambridge, MA

ProfitLogic
April 2000 - January 2001

I designed and built large scale data processing and scientific computing engines used in ProfitLogic's flagship service. Our software set prices on several billion dollars of retail merchandise each week.

Miscellaneous

Citizenship: US Citizen.

Grade Point Average: 4.8/5. (in major)

Computer Languages: Python, Rust, Java, C, C++, SQL, various assemblers, Matlab, Mathematica, Common Lisp, Scheme, Haskell, Bluespec, VHDL, LaTeX.

Operating Systems: Linux, TinyOS, Windows, Solaris.