

# Dhruv Matani

Mobile: (631) 403-7653 | E-mail: [dhruvbird@gmail.com](mailto:dhruvbird@gmail.com) | GitHub: [github.com/dhruvbird](https://github.com/dhruvbird) | Blog: [dhruvbird.blogspot.com](http://dhruvbird.blogspot.com)

---

## Profile Summary:

A software engineer with 9+ years of experience with large scale distributed systems, performance critical services, and a passion for creating developer friendly APIs.

## Work Experience:

**Facebook Inc., Menlo Park** **March 2013 - Present**

### *Software Engineer*

#### **Logger (Product Infrastructure) – Tech Lead**

- Leading the Logging Infrastructure team at Facebook. This includes Logging from mobile devices, web browsers, web services, and various backend services.
  - I'm the mentor for 9 developers on the team, and point of contact for all cross-functional initiatives regarding warehouse data logging
  - Achieved phenomenal efficiencies at various levels by re-architecting the on-wire serialization format for loading data into the Facebook Data Warehouse. This affects both batch and real-time workloads
  - Re-architected the PHP Logger framework to leverage type-safety and runtime efficiency offered by Hack and HHVM. This resulted in single digit percentage wins across the entire fleet of machines
  - Leading the implementation of an intent driven sampling initiative for logging data into the warehouse to allow data growth to scale independently of the popularity of the platform
  - Led the backend C++ Logger initiative to ensure high quality data logging from services at Facebook to various downstream data stores. Changed the development paradigm for logging by providing static type checking (instead of run-time failures and alerts) against downstream table schemas in C++ code, greatly increasing developer productivity
  - Led the cross-language C++/PHP/Python Reader initiative to provide a consistent data ingestion API for real-time workloads across Facebook. Involves integrating with multiple real-time solutions at Facebook such as Stylus and Puma. The focus here is also to have downstream table schemas bubble up into code, and have the code fail fast, enable developers to iterate quickly, and for the application be more reliable. Additionally, the reader frees developers from worrying about the positional nature of columns in their data streams, and provides them the convenience of keyed column lookup without the related performance overhead

#### **Ads Targeting and Custom Audiences**

- Led the effort to platformize the custom audiences backend (for use in other targeting products) and improve targeting by enabling matching on a broader category of traits

#### **Scuba (Data Infrastructure)**

- Part of the team that implemented a performant in-memory columnar storage engine that results in increased data compression
- Improved data ingestion CPU utilization by 4x (have you heard of merge sort?)
- Improved overall reliability and availability of scuba (i.e. add another 9!)
- Part of the team that implemented Fast Database Restarts in Scuba, bringing restart times down to 120 second from 2.5 hours (yes, that's 1.3% of what it was)
- Migrated the running service (which runs on thousands of machines) to a different region with zero downtime, over a period of 2 months; i.e. rolling migration without service disruption (ask me why this was challenging, and how I used scp on in-memory data)
- Led the disaster recovery effort, which replicates data in real time across machines in multiple geographic regions

**Facebook Inc., Menlo Park** **May '12 – Aug '12**

### *Software Engineer Intern*

- Developed a per-table backup and restore tool (for MySQL databases), which lets you restore individual tables from a complete database backup without restoring other unrelated tables. This tooling enabled much faster table restores in case of MySQL table corruption

**Directi Pvt. Ltd., Mumbai, India** **Jan '09 – Jun '11**

### *Senior Software Engineer*

- Authored, tested, and integrated an XMPP BOSH Server for use in the talk.to project. This was later released as an Open Source project, which is used by many globally as a standalone BOSH proxy. <https://github.com/dhruvbird/node-xmpp-bosh>
- Responsible for problem-setting and training interviewers for the IIT graduate hiring process
- Developed and owned the online update logic for .pw desktop chat client. This handles creation, shipping and fail-safe application of differential updates (deltas) which update the application in a way that minimizes network data utilization
- Setup, administration and question setting for the Directi Online Test (DOT) which is used for the recruitment of software developers and engineers
- Developed the Business Logic Layer for administering the chat server and related services
- Led the project to develop and deploy an online community driven translation tool to help internationalize web applications. <http://sofi.directi.com/>
- Email Context Analysis Engine (CAE): Developed an email context analyser to determine the most relevant commercial keywords/phrases for a given email

**Mukesh Patel School of Technology Management & Engineering., Mumbai, India**

**Jul '08 – Jan '09**

**Lecturer**

- Taught a course on Systems Architecture & Programming
- Taught a course on Operating Systems-II

**Calsoft Pvt. Ltd., Pune, India**

**Jul '06 – Apr'08**

**Senior Development Engineer**

- Developed an HTTP Caching Proxy to cache YouTube flash video content for Umber Media systems
- Prototyped an RTSP Caching Proxy to cache YouTube 3gp video content for Umber Media systems. Also performed re-encoding and down-sampling of the video and audio content. This proxy also let you insert advertisements at various places in the original video. It was prototyped for use as a video ad-serving proxy
- Developed the DiskImage tool which is used by ScaleMP for fully automated installation of linux (distribution hosted on TFTP servers) on machines
- Performance tuned the Ingres Database Server's statistics collection module for ScaleMP's VSMP architecture (which is a glorified NUMA architecture)
- Optimized and fixed long standing bugs in the logpar (log parser) application for ScaleMP, which is used for parsing the performance logs produced by their system

**Education:**

**Masters of Science (M.S.), Computer Science** Aug '11 – Dec '12, GPA 3.82/4.00  
Stony Brook University, Stony Brook (NY)

**Bachelors of Engineering (B.E.), Computer Engineering** Aug '02 – Jun '06, Aggregate: 62.6%  
University of Mumbai, Mumbai, India

**Publications:**

- Fast Database Restarts at Facebook ([17 citations](#)) 2014  
<https://research.facebook.com/publications/fast-database-restarts-at-facebook/>
- Avoiding locks and atomic instructions in shared-memory parallel BFS using optimistic parallelization ([7 citations](#)) 2012
- Partial deamortization of the Packed Memory Array  
<http://dhruvbird.com/pdpma.pdf>
- Compressing the human genome against a reference ([3 citations](#)) 2011  
[http://dhruvbird.com/genome\\_compression.pdf](http://dhruvbird.com/genome_compression.pdf)
- An  $O(k \log n)$  algorithm for prefix based ranked autocomplete ([14 citations](#)) 2011  
<http://dhruvbird.com/autocomplete.pdf>
- An  $O(1)$  algorithm for LFU (Least Frequently Used) cache replacement ([18 citations](#)) 2010  
<http://dhruvbird.com/lfu.pdf>
- A technique for extracting song lyrics from web pages without knowing their structure 2006  
<http://dhruvbird.com/liblyric.pdf>
- A distributed approach for solving a system of linear equations ([1 citation](#)) 2004

**Skills:**

Programming Languages: C, C++, Python, Hack, Javascript, PHP, Bash, Object Pascal, SQL  
Presentation technologies: HTML, LaTeX  
Platforms: Linux, MS-DOS, Windows  
DBMS: MySQL, SQLite, Hive, PostgreSQL, Oracle

**Contributions:**

- Architected and built the infrastructure powering the analysis in the book “Who’s Bigger” – by Steven Skiena and Charles Ward
- Provided key insights into the complexity analysis of the R1Q algorithm mentioned [here](#)
- Added a cache-optimized single-object allocator (bitmap\_allocator) to libstdc++
- Contributor to libstdc++-v3 (g++’s C++ Standard Template Library) – which is used to power a vast majority of applications built using C++