Nate Dire

P: 425.405.5263
E: nate.dire@gmail.com
W: http://www.linkedin.com/in/ndire

Summary

- Proven lead software engineer with contributions to all phases of SDLC from research to post-release support.
- Grew into architect role while helping to scale enterprise storage product from $0 to over $500M revenue.
- Designed and supported production code in FreeBSD user-space and kernel, running on high-performance clusters up to 144 nodes and 15 PB.
- Advanced knowledge of C, Python, and Subversion.
- Familiar with C#, C++, Git, Java, R, SAS, and SQL.

Education

Whitman College
B.A. Mathematics, *cum laude*, May 1998
- Phi Beta Kappa
- Presented numerical analysis research at 1998 AMS/MAA Annual Meeting.

University of Washington
M.S. Computer Science, June 2005

UCSD Extension
Biostatistics Certificate, December 2011
Data Mining Certificate, September 2012

Work History

StatFame, LLC, Seattle, WA
Software Engineer (part-time) June 2012 - present
- Built box score import scheme for multiple data sources using C# and T-SQL for Azure deployment. Implemented HTML scrape import.

EMC, Isilon Division (formerly Isilon Systems), Seattle, WA
Consultant Software Engineer April 2011 - present
- Writing over 200-page product internal architecture document in LaTeX for new engineers and technical field personnel (part-time since April 2012).
- Presented clustered file system architecture at EMC World 2011 and 2012.
- Designed and implemented new BSD vnode operation locking to enable file system filter API.
- Met with customers and consulted on product deployment configurations.
- Managed large cluster scalability roadmap.
- Member of product architecture team.

Lead Software Engineer August 2008 - April 2011
- Independently formed multi-department group to create field product configuration tool used by all sales engineers.
- Development lead for multiple feature maintenance releases.
- Wrote Subversion wrapper in Python to enforce best practices.
- Led design and implementation of 2nd-generation, multithreaded and distributed job engine which scales to 144 nodes.
- Led design and implementation of file system SSD strategy.
- Planned and executed roadmaps for tiering and integrity features.

Software Engineer March 2003 - August 2008
- Manually fixed on-disk corruption on live production systems.
- Implemented special customer requests, from consultation to specification to delivery.
- Co-led design and implementation of file system quotas.
- Researched and evaluated erasure codes. Implemented Reed-Solomon codes for 4-failure protection, including hand-optimized x86 assembly with SSE2 instructions.
- Designed and implemented mark-and-sweep collection for orphaned file system structures.
- Independently researched reliability analysis techniques. Implemented Monte Carlo simulator in C++. Coordinated Mean-Time-To-Data-Loss estimation with Marketing, QA, and Operations.
- Designed, implemented, and supported clustered job engine in Python, which scaled to 96 nodes.