Di Xu

Kirkland – WA

+1 (425)-615-3057 • xudifsd@gmail.com • github.com/xudifsd

Professional Summary

Principal Software Engineer with 10+ years of experience in distributed systems and backend infrastructure. Expert in **high-performance computing (C++/C#)**, **consensus algorithms (Raft)**, and **stream processing (Kafka)**. Proven track record of modernizing legacy infrastructure at Microsoft and Baidu, delivering **99.99%+ availability** and reducing system latency by **25%+**.

Working Experience

Microsoft Corporation

Principal Software Engineer — **Redmond, WA**

Recommendation platform — *Sep 2025 - Present*

- Completed the migration of all remaining traffic to the Content Builder service, successfully deprecating legacy protocols and modernizing the ingestion stack.
- Migrated Feature Extraction Library to Linux for LLM Grounding service.

Senior Software Engineer — **Redmond, WA**

Recommendation platform — *Sep 2021 - Sep 2025*

- Content Builder: Architected an event-driven ingestion platform (Kafka) to replace legacy point-to-point pipelines
  - **Orchestrated** a multi-quarter migration strategy for 2 major partner teams and migrated 10+ other teams at scale.
  - **Negotiated** with stakeholders in MSN Feeds & AI to deprecate legacy protocols. Overcame resistance by designing a backward-compatible shim that allowed teams to migrate smoothly, unblocking a multi-year technical debt initiative.
- Feature Extraction Library: Designed and implemented a static C++ library to replace legacy C# feature calculation.
  - Reduced P95 E2E latency by **2ms** and CPU usage by **15%** in the first migration experiment.
  - Led the migration of 50+ feature sets, designing a config-driven interface that allowed Data Scientists to deploy changes without engineering intervention.
- Optimized read path for Tally counting service.
  - Reduced DB key preparation latency by **250ms (25% reduction)**.
  - Restructured cache memory layout, reducing cache fill latency by **50% (1.2ms to 0.6ms)**.

Software Engineer II — **Bellevue, WA**

Recommendation platform & Deep Learning Training Service — *Dec 2019 - Sep 2021*

- Recommendation platform
  - Designed and implemented service downgrade, improved availability under heavy load.
- Deep Learning Training Service
  - Optimized Job-Manager, reducing 95th percentile job creation time from 400s to 46s.
  - Optimized init process, reducing job initialization time from 45s to 2s.

Software Engineer II — **Beijing, China**

STCA — *Jun 2018 - Dec 2019*

- Designed and implemented PAI runtime for OpenAI platform, the first process running in the container. Unified runtime environment of Hadoop-based backend and **Kubernetes**-based backend.
- Collaborated cross-functionally with the Bellevue team to build Deep Learning Training Service.

Baidu Inc.

Software Engineer — **Beijing, China**

Infrastructure — *Mar 2016 - Jun 2018*

- Re-architected the "Matrix" cluster management system (similar to Kubernetes/Borg) by replacing the MySQL storage backend with a **Raft-based replicated state machine**.
- Improved system availability from 99.9% to **99.95%** and reduced leader failover time from 30 minutes to **<1 minute**, eliminating the need for manual SRE intervention during master failures.
- Developed custom scheduling algorithms (disk/host selection) that improved cluster utilization and resource isolation for PaaS tenants.

Internship

Alibaba Cloud Computing (Aliyun) Inc. **Beijing, China**
R & D Engineer Intern *May 2015 - Sep 2015*
Pangu system, a **GFS** like distributed storage system
- Collected & analyzed usage data from production env, generated report to drive more efficient system usage, resulting in 50% less disk usage.
- Analyzed & restricted policy of list & read APIs, improved QPS from 110k to 150k.

BearyInnovative Inc. **Beijing, China**
Software Engineer *Aug 2014 - May 2015*
BearyChat, a Slack like product
- Designed backend topology, re-architected service from single node to distributed nodes, eliminated the risk of timestamp conflicts & single point of failure.
- Implemented 4 critical & many other important product features in web API server written in **Clojure**.

AdMaster Inc. **Beijing, China**
R & D Engineer *Apr 2013 - Sep 2013*
Data collecting system
- Implemented 10 data processing algorithms in cascading, answered how many UV & PV per ad and characteristics of the viewers. These data will be presented to users directly.
- Maintained & optimized data collecting module written in **Python**, accelerated it by 13%.

Project Experience

Google Summer of Code **Remote**
Typed Clojure *May 2014 - Aug 2014*
Added 2 function type annotations to **Typed Clojure**. Made the type system more sound.
- Came up & implemented a design much more expressive than mentor's.
- Improved core library annotation coverage by 7%, made Typed Clojure more user friendly.

CCF-Tencent Open Fund **Suzhou, China**
Java Symbolic Executor targeting Android Apps *Oct 2013 - May 2014*
Used by Android analysis team in Tencent Inc. Sponsored by CCF-Tencent Open Fund.
- Designed & implemented core abstraction, enhanced robustness & readability.
- Optimized the program before shipping to the users, reduced its memory usage by 65%.

Skills

- **Languages**: C++, C#, Python, Java, Clojure
- **Infrastructure**: Linux, Kafka, Kubernetes, Raft

Education

University of Science and Technology of China **Suzhou, China**
*M.S. in **Software Design*** *Sep 2013 - Mar 2016*

Dalian Polytechnic University **Dalian, China**
*B.S. in **Computer Science*** *Sep 2009 - Jun 2013*