

# ZhangHao - Internet / R&D

**Mobile** +86-189\*\*\*\*9173

**Email** conan.np@gmail.com

**WeChat** npngny

## About

### Basic information

- Male. Born in **1982**.
- IT / Internet / Computer game. Development, Architect.
- Born in Liaoning, Shenyang, live in Beijing now.
- Graduated in Northeast University, CS.

### Summary

- Since early 2004, I have been engaged in Internet software development, with experience in areas such as **online games, social media, entertainment, productivity tools, and network security**. I have extensive experience in designing and implementing **service architectures** for business systems, as well as in **network communication services** and **data flow processing**. I have practical work output in areas such as **operational infrastructure**, business logic implementation, interactive displays (such as games), and audio and video processing. I have also been involved in the design and implementation of systems related to **intelligent hardware, industrial intelligent systems, and digital development metrics**.
- Overall, I possess clear engineering thinking, am skilled in architectural design, logical abstraction, and technical research.
- I have had exposure to B2B (business-to-business) services, including enterprise office, refined labor management, and industrial process management.
- Familiar with **C / Lua** (communication services and game-related), **PHP**, **Go** (communication services, platform, electronic games, and data logic). I have experience in developing with **AS3** (non-effects related) and have briefly worked with **C#** (Unity game client) development. I am familiar with **POSIX-like** systems and have considerable experience in script writing.
- During my employment, I have been involved in the research and development of relatively large-scale systems and have participated in several **entrepreneurial projects**. I am familiar with the work cooperation methods of mature companies and can adapt to the rhythm and state of start-up companies. I am able to work with teams of various styles and can complete some **special and unique research and development tasks** during critical periods.

- Have the ability to think and comprehend certain types of products from a non-technical perspective.
- Have experience in developing and maintaining open-source software, developing and selling personal **commercial software**, as well as outsourcing. In earlier years, I was passionate about technology communities.
- Have been involved in computer software development for more than 20 years, possess a solid foundation of knowledge, have a broad technical perspective, and can quickly adopt new technology methods and tools. I am cautious in my design approach, maintain strict development standards, and have a professional problem-solving attitude. I recognize the alignment between my personal attributes and the industry, and have many insights on industry and technology-related topics. I also have considerable experience in technical recruitment and **job interviews**.

## Experience

### MediaTrack (2021/09 - 2022/06)

#### Infrastructure - Architect [https://www.mediatrack.cn]

- Infrastructure

Promoted the upgrade of the server-side technology system for MediaTrack, framework organization (**go-micro** v2 => go-micro v4), **micro-service** planning, and hierarchical adjustments.

Advanced the establishment of the basic research and development process and the landing of standardized processes.

Reviewed and made decisions on key technical solutions, and promoted the upgrade of data availability.

Implemented some services.

I left MediaTrack for the industry direction selection issues.

### Parallel Future (2021/01 - 2021/8)

#### PandaTV - Senior Architect

- New Panda

Panda.TV (The non-homonymous product), inherited the first-line product PandaTV in the live broadcast industry from 2016 to 2019. It was created with the hope of leveraging the brand effect and user resources of the original PandaTV, and is based on the new concept of MetaVerse to create a new Panda Live product that focuses on virtual image broadcasting and interactive entertainment.

During my employment, I completed the design, selection, and planning of the backend service architecture, based on the **microservices** architecture

of **go-micro** v3. I also established standard development processes for server-side development, such as CodeReview and other research and development related work.

In the early stages, I organized the backend service architecture and data of PandaTV, which was concluded in 2019, and started from scratch to promote the implementation of the new system framework until the pre-launch stage.

During that time, I provided technical support to various business lines and led the service design and development of the user system and asset (financial) system.

Independently completed the design, development, and integration of the long **connection communication** service for game and live broadcasting messages.

Due to certain reasons, the company **stopped supporting** the project in August 2021, resulting in the new Panda Live not being released to the public.

#### **QiAnXin (2019/10 - 2021/01)**

##### **Office of engineers - Architect**

- 2019/10 - 2020/04 : Hiner - Publishing system of artifacts

Hiner is the group's unified product release system for external use, consisting of an internal release interface and external distribution (download) function. It completes the collection, local upload, storage backup, hash whitelist, CDN and other processes in a **configurable workflow** mode. Currently, it supports product releases and download management for various business lines within the group, and is the only officially recognized channel for software product delivery.

Completed the overall architecture design and review of the system, led the team to implement requirement analysis, feature development, testing, deployment, and provided technical support and services to the users (almost all product lines of the group).

The system is implemented in **Go**, deployed in a **Docker Swarm** cluster in a multi-subservice manner, completely coupled through message queues. The system implements large file chunk upload, verification, security scanning (access to TianYi, XiAn platform. VirusTotal has not been formally accessed due to cost issues), recursively unpacks artifact packages and adds them to the scan whitelist of security software (TianYi, QiAnXin Safe Defender), CDN push and automatic preheating (fully adapted to BaiShan Cloud, Kingsoft Cloud, AliYun), distribution authority management, version tree management, etc. It provides artifact indexing, version upgrade assistants, and other services to external users.

In the later period, in order to meet the demand for automated distribution process, I provided integration support for other engineering systems within the group, completed the direct access support for **Jfrog Artifactory** and bare S3, developed cross-platform distribution front-end tools, and integrated with third-party access control systems.

After the release of Hiner, it replaced the old release system, taking over several hundred distribution pipeline requests every day, distributing single package files larger than 500GB, and distributing special packages where the number of files is magnified by 300,000 times after recursive decompression.

- 2020/04 - 2021/01 : Fusion - Platform of digital measure & efficacy

The Fusion platform is an important part of the group's **digital** system, providing metrics for research and development data and implementing DevOps pipelines, providing important quantitative support for research and development efficiency management.

The development process begins with VCS action, triggering the Fusion pipeline execution through git-hooks and gitlab webhook. The pipeline is configurable and based on Jenkins, connecting various tools such as SonarQube, Open Source Defender for code review, security scanning tools, automated testing, etc., and aggregating the results into the Fusion platform.

Fusion consolidates two types of data: code assets and pipeline execution results. The platform provides plugins to perform statistical analysis on the data from different dimensions and presents it in a visual format, including code quality ranking of the production line, assessment of problem fixing time, and scanning reports.

In addition, Fusion is also integrated with the highly customized Jira pipeline within the group, which links testing issues to the code itself. And the internally developed Tengyun platform within the group is also linked to developer time management and code engineering quality.

Deep cleaning and organizing of the original massive data in SonarQube were carried out during this period.

Real-time notification integration with Lanxin (Qianxin Group's secure instant messaging platform).

The CI pipeline supports both GitLab-CI and Jenkins jobs, and the CD endpoint supports integration with Hiner artifact distribution. As most of the products in the group rely more on the output of artifacts or physical devices than deployment, this is not a strong dependency.

The engineering data covers nearly 4,000 R&D personnel and over 20,000 code repositories (10,000+ active), with more than 200,000 development branches, millions of commit activities, and over tens of billions of lines of

code. The platform performs an average of 4,000-5,000 silent analyses per day.

The platform is implemented as a set of services, with the server-side written in **Go** and using **PostgreSQL** as the database. It is deployed on a **Docker Swarm** cluster, and the front-end is mainly implemented using Vue.

To meet the overall requirements, a large number of tool programs and scripts have been developed, including signature search tools, gitlog cleaning tools, various git hooks, and distributed task scheduling.

Fusion itself supports Prometheus exporter, and the data dashboard is displayed on Grafana.

### **Qihu.360 (2014/12 - 2017/12)**

#### **PC Game group - Architect**

- 2014/12 - 2015/04 : Content and account service of mini-game platform

As 360 users do not have any special social attributes, nor do they have entertainment stickiness or accumulated content appeal, we planned to use the PC-side (security products, browsers, etc.) as an entry point to increase user activity through interactive mini-games, providing terminal users with more content experiences and achieving traffic diversion for the game platform. We implemented lightweight game interfaces such as points, rankings, and matching services.

Some backend interfaces, written in **PHP**

- 2014/12 - 2015/06 : **PVR** games / “Cool Run” series

PVR stands for Person VS Record, which is a game interaction method where players compete against recorded gameplay. This method is usually used in same-scenario competitive games (such as parkour or simplified racing) where player actions do not affect each other. PVR provides a more efficient and timely matching experience for players in certain game scenarios. The Record backend service is developed using PHP. The normal Record is not a video or a wavelet recording, it is simply the correspondence between player actions and a constant frame.

The Cool Run series gathered a considerable amount of traffic during the 2015 Spring Festival period and completed a portion of the platform’s user conversion.

- 2015/03 - 2015/07 : Tetris online, multi-player version classic game

To provide more support for casual game types and to achieve better competitive experience, we designed a real-time game service. The architecture inherits from the previous personal project, using **C** to develop the network

communication service, and embedding game business logic written in **Lua** to support real-time game experience.

Developed a communication client package in AS3, which uses a **proprietary binary protocol** to provide data carrying and validation. The game project itself was based on the “Russian Firefighting” series launched on the Tencent game hall in 2003. Due to technical limitations on the client side, the operation experience of the “Russian Firefighting” series was not fully designed, but a more reasonable match mechanism based on performance was arranged. Later, due to project arrangements and personnel changes, the game was not officially launched.

- 2015/09 - 2016/09 : Communication service / Real-time game implementation / **Day-time** tasks

Under the circumstances of not supporting the game project itself, I continued to advance the development of real-time game server. Due to the transfer of manpower in the department, the self-developed business was mainly promoted by individuals. During this period, some energy was invested in other technical projects of the department.

- 2016/09 - 2017/01 : **Containerization** of micro-services, research and implementation

This containerization transformation attempted various popular technology solutions at that time, involving service governance, orchestration, clustering, and **network**, and finally implemented the transformation on the **docker swarm** (due to the version leap and document completeness issues of Kubernetes at that time, it was not ultimately adopted) solution, and migrated some services to the swarm cluster. During the research, two versions of the nginx (non-pro) configuration tool “gourd” were developed (**C** and **Lua**), and the business gateway was replaced by traefik. Many components such as etcd, flannel, register, etc. were practiced during the research.

- 2017/02 - 2017/07 : IO game - “Color & Color”

Designed and implemented a general real-time game server engine called **Langer** and a game logic container called **Slater**, and implemented part of the logic for the game “Color&Color”. Langer and Slater were completely re-implemented in **Go** based on the design of the previous PVP game service. As there is no embedded dynamic script, Slater is not suitable for business-type games with frequent logic modifications, but is very suitable for casual games organized in rooms. Langer includes a complete authentication service, Dispatcher, Gateway, and state storage, and maintains stable mesh communication with Slater.

## Qiyou Interactive (2012/02-2014/12)

### Architect / OPS / Tech support

- 2012/05 : Simulation of SAE engine

During the migration of the game “WeiSango” from the Sina SAE platform to other IaaS public clouds, we encountered limitations with SAE’s non-universal services. In a relatively short amount of time, we developed and launched a service called **FakeSAE**, which simulated the HTTP queue and scheduled task services of SAE, allowing the existing game business code to be migrated to other public cloud platforms with almost zero intrusion. FakeSaeQueue relies on Redis to provide queue storage, implements SAE’s supported ordered and concurrent (unordered) queues using **C**, uses **Lua** as a configuration script, and eliminates SAE’s original queue length limit. After the service was launched, it ran stably until the game was taken offline in 2018.

The “WeiSango” series, as a web game, has achieved over 7 years of profitable operation.

- 2013/01 - 2013/04 : Real-time service of “Dominance of Sango”

“Dominance of Sango” is another web game planned by the company with a Three Kingdoms theme. The design style is more realistic than “WeiSango”, and the combat system includes real-time logic. Inheriting the **communication service** and **millisecond-level timer** developed in a previous personal project, it provides basic support for the game’s combat logic. The service main body is developed using **C** and embedded with logic scripts written in **Lua**. The game was interrupted due to business adjustments after the internal test.

- Responsible for the overall **operations** and internal **IT management** of the company (due to the relatively small size of the company).

Managed and maintained the online and development testing servers and other resources, developed release processes and resource evaluation. Implemented a complete internal permission system, daily authentication mechanism based on LDAP, and connected all development resources and company human resource management, etc.

## WuXianTianLi (2010/06-2012/04)

### 51YunCai - Architect

- Design and implement of real-time pushing service, used in 51yuncai.com, written in C.
- The technology selection, scheme design. PHP for view layer and JAVA for the data layer (not involved in JAVA project) and Oracle database.
- Communicate support for CPIC project.

## MEGA Media (2005/12 - 2006/08, 2007/10 - 2010/04)

### PHP developer

- 2005/12 - 2006/08 : 5Jia1.com

It was one of the early social platforms in China, focusing on campus communities and representing an early model of Web 2.0. The backend logic was implemented using **PHP**. I was responsible for designing and developing social modules such as **image storage** and **forums**. I also wrote **crawlers** to fetch and process on popular posts from forums on edunet, and later did some framework consolidation work.

### Game engineer

- 2007/10 - 2008/08 : General service of cards game

The first version of real-time game service was developed using **C** for communication service and **Lua** for business logic, with a Flash client based on **AS3** (not flex). I **independently** completed the implementation of the service and participated in the development of the business logic for the first card game “BlackJack” based on it, which was launched and operated on uspace. However, it was taken down later due to regulatory restrictions.

- 2008/07 - 2009/03 : “RealGo”

This project was focused on implementing a simulation of the game of Go (also known as Weiqi), rather than a traditional game. It strictly followed the rules of the 19-line Go game and allowed two players to play against each other. Despite having no prior knowledge of Go, the project was able to implement the complete Go rules using **Lua** and developed the main part of the Flash-based client. The project was online for a period of time and maintained a certain level of user activity, although it did not follow the typical standards used to measure the success of traditional games due to the unique nature of the user group.

- 2009/03 - 2010/04 : “Batoo”

A game project in cooperation with the South Korean side, featuring non-standard Go rules, but rather a short-time game on 11/13 lines. In the absence of complete documentation and source code, the binary data protocol of the Korean game service was analyzed and organized, and an **AS3**-based client was rewritten to support Batoo’s web version. Eventually, the Flash version could be fully synchronized and mixed with the client version. Batoo’s domestic operation was closed after 2010.

## Sina (2006/09 - 2007/09)

### Backend develop



- User analysis of Sina Blog

The user login status, browsing time, and behavior after login of the blog were counted through log tracking, and the detailed numerical values were used to support operation.

- Counter service of Sina Blog - memcachedb

A data persistence implementation based on the modification of **memcached** 1.1x, using **BerkeleyDB** for actual storage, and relying on atomic operations of memcached for post-level access statistics. The service was a prototype implementation of the later open-source **memcachedb** project by Sina.

## QunZhi Shanghai (2004/10 - 2005/03)

### ComicV - Backend develop

## Projects

### Link (HereweTech)

- Developed from the community version of the Long-Connection service of PandaTV, it provides clustered single-duplex / dual-duplex websocket, UDP frame synchronization and TCP gateway with logic script supported, and corresponding simple user system and runtime statistics.
- Based on go-micro V3

### BS.Play

- Game communication services after four iterations, basic services developed in C and embedded logic scripts implemented in LUA. Among them, a block-based memory pool (removed later), abstract IO multiplexint, static thread pool, simple spin-lock, full-stack websocket protocol and other basic components are implemented, and a set of LUA APIs are developed for event writing. The project has supported game businesses such as cards, strategy and casual battles, and non-game business such as market information pushing.
- Rewritten in Go (Langer / Slater) Later.

### Attendance gateway of LaoQin cloud (Commercial)

- Software gateway designed for some sets of attendance machine, supports private communication protocol over TCP (HanWang) and HTTP (ZhongKong). Developed with golang.

### **ID5DB (Internal)**

- On-disk storage service based memcached protocol, store KV data into file through MMAP. Archvied data saved in encrypted format, compressed with google-snappy. ID5DB used for sensitive projects in 2013.

### **AudioMixer**

- A tiny tool written in C, based on libmpg123. AudioMixer mixed multiple PCM soundtracks into single one, supports noise filter, overlay proportionally. Used in some music games.

### **BS.Gallery (Commercial - OSS)**

- A web gallery software written in PHP / Flash, which implements full-featured storage and browse of images. I sold some copies in 2005, and transferred it to OSS.

### **Education & addition**

- Northeastern University - Computer science and technology (2000-2005)
- Certification of senior programmer - Ministry of Information Industry (2003)
- First prize of Liaoning province - NOI98 (1998)
- Amateur musician - \_\_-