




김성훈

개발을 즐기는 개발자, 개선을 좋아하는 개발자 김성훈입니다.



 elky84@gmail.com

 blog |

<https://elky84.github.io>

 github |

<https://github.com/elky84>

업무 프로세스, 협업에 관심이 많습니다. 비효율을 제거해, 결과물의 퀄리티를 높이면서도, 시너지가 나고, 불 필요한 스트레스를 제거하는 것에 아주 관심이 많습니다. 또한 좋은 개발자이면서도, 좋은 동료가 되는 방법에 대해서도 관심이 많고, 그러한 팀을 만들고 싶은 욕심이 있습니다.

- 동료와 함께 일하는 즐거움, 함께 성장하는 즐거움을 통해 좋은 팀을 만들고 즐겁게 일하고 싶습니다.

시도하고 개선하는 것을 좋아합니다. 프로그램은 안정성이 최우선이지만, 안정성과 새로운 시도와의 균형 잡힌 시도로, 유의미한 결과를 끌어내는 것을 좋아합니다.

- 대용량 트래픽을 받기 위해선 다양한 방향의 개선이 따라야 합니다.
- 시스템을 보수적으로 바라본다고 해서 장애나 이슈가 생기지 않는 것이 아니기에, 부하 테스트를 비롯한 다양한 테스트 환경 구성을 통해 이를 미리 대비하는 것이 중요하다고 생각합니다.

여러 포지션을 다양하게 경험해온 개발자입니다. 다양한 경험을 한 분야는 게임 서버 개발, 게임 클라이언트 개발, 웹 백엔드 개발, 웹 프론트엔드 개발 등을 해왔습니다.

- 가장 자신 있는 분야는 소켓 서버 개발과, 최적화, 트러블 슈트, 멀티 스레드 프로그래밍입니다.
- 다양한 경험을 바탕으로 다양한 관점에서의 접근과 시도를 통해 제너럴 리스트의 장점을 보인 경험이 많습니다.

기록 집착자입니다. 개발 문서, 개인 적인 취미 활동, 하루 회고, 개선안, 회의록 (개인적 차원) 등 수많은 내용을 기록하는

것이 일상입니다.


- 업무 관련한 기록도 작업 하나 하나 마다 남기고 관리하는 것을 선호하며, 팀이나 회사에서 강제되지 않더라도 업무 일지와 업무 관리 도구를 쓰고 관리하는 것을 즐깁니다.




Career



크래프톤 | CL3

 2023-12~2024-03

 기술 스택


ASP.NET CORE , C# ,
MongoDB , RabbitMQ ,
Docker , Kubernetes ,
Unity3D , Pytorch ,
MLAgents

딥러닝/게임 | 웹 서비스/유니티 | Tech (Web)

- 딥러닝을 게임에 활용하는 시도를 하는 연구 조직
- 기존 프로젝트 개선 작업을 위해, 단기 합류



오피지지

 2022-12~2023-11

PEEP, qeeq | 화상 채팅, 모바일 게임 소셜 앱 | Backend (Kotlin Spring Boot)

- PEEP Cell의 Tech Lead로 합류
- PEEP의 백엔드 기능 추가
 - 화상 채팅 서비스
 - 방 관리, 워크스페이스 관리
 - 인증/인가
 - 회원 관리
- qeeq의 백엔드 개발

기술 스택

Kotlin , Spring Boot ,
JPA , MongoDB ,
PostgreSQL , GraphQL ,
Next.js , Nest.js



크래프톤 | CL3

17 2020-07~2022-12

기술 스택

ASP.NET CORE , C# ,
MongoDB , RabbitMQ ,
Docker , Kubernetes ,
Unity3D , Pytorch ,
MLAgents ,

C++ 14 , IOCP , SQL
Server , ODBC , ASP.NET
MVC

- 모바일 게임 소셜 앱
- 게임 관련 투표/퀴즈
- 게임 관련 친구에 대한 투표/퀴즈

딥러닝/게임 | 웹 서비스/유니티 | Tech (Web)

- 딥러닝을 게임에 활용하는 시도를 하는 연구 조직
- Python (Pytorch), Unity3D MLAgents 기반의 딥러닝 라이브러리 활용
 - 주력 AI Engineer는 다른 분이 여럿 계신 상황이며, 이해도와 간단한 활용 정도만 가능
- C# ASP.NET CORE 를 이용한 서버를 통해, 활용되는 서버에서 쉽게 가져갈 수 있게 2차 가공 및 자원 전달 기능 작업
 - Ocelot (Spring Cloud Gateway , Netflix Zuul , Armeria 와 같은 API Gateway용 패키지)로 MSA 로 서비스를 구성
- 코드 생성기 등을 포함한 지원 도구 개발

테라 PC | MMORPG | Tech (Server)

- PC MMORPG
 - 한국 (넥슨 자체 서비스), 북미, 유럽, 일본, 대만 서비스
- C++ 14, VS2019 기반의 서버 기능 개발/트러블 슈트/퍼포먼스 튜닝
- C# ASP.NET MVC 기반 Admin Tool 개발
- 코드 리뷰, 스트레스 테스트, 유닛 테스트, 소통 프로세스 개선



모히또 게임즈 (조이시티 자회사) | 선임 (팀장)

2019-05~2020-07

기술 스택

C# , ASP.NET CORE ,
Unity3D , RabbitMQ ,
MongoDB , MariaDB ,
WebSocket , Docker ,
Kubernetes

프로젝트 M | 수집형 RPG | 서버 & 클라이언트 개발

- .NET CORE 3, ASP NET CORE 3 기반의 MSA 아키텍처 구축
 - Kubernetes , Docker 를 이용한 서비스 구축 및 배포
- 웹 기반 기술로 MORPG 장르의 서버 구축
- RabbitMQ , WebSocket , MongoDB , MariaDB 등 오픈 소스를 기반으로 한 서비스 구축
- Unity3D , UGUI 기반 및 콘텐츠 핸들링 로직 구현



NHN | 선임

2018-09~2019-05

기술 스택

Vue.js 2 , Java 8 ,
Spring Boot 2 , JPA ,
MySQL

TOAST 클라우드 | 공통 모듈 (CAB) | 풀스택 개발

- CloudTrail 개발
 - TOAST 클라우드 서비스의 사용 이력 추적 및 조회 시스템인 CloudTrail의 Frontend 개발
- 다국어 서버 개발
 - REST API + Admin Tool로 다국어 관리 및 시스템 별 범용 다국어 기능
- 국적에 따른 EMAIL/SMS 발송 기능 개발
 - REST API를 통해, 사내 모든 시스템이 이용하는 국적에 따른 EMAIL/SMS 대행 API 작업
 - 인증, API 스펙을 사내 공통화 하기 쉬운 규격화 한 뒤, 벌크 처리가 용이하게끔 개선
- Admin Tool 개발
 - 공지사항 관리, 각종 편의성 작업



넷마블 - 인프라 개발팀 | 과장

17 2017-07~2018-09

기술 스택

Java 8 , Spring Boot ,
JPA , MySQL , MongoDB ,
Netflix Zuul , Python ,
Apache Airflow , AWS SDK ,
Docker



NHN
Entertainment


넷마블 전사 인프라 자동화 도구 | NOS | 백엔드 개발

- **Docker** 도입
 - **Docker** 를 이용한 서비스 이미지 생성 및 운용
- **AWS Device Driver** 개발
 - **AWS SDK** 를 연동하여, private cloud 시스템과 같은 agent 및 컴포넌트를 사용할 수 있게끔 연동하는 드라이버 개발
- **API Gateway** 개발
 - **Spring Boot** 기반의 **Netflix Zuul** 을 통한 endpoint 관리, 권한 관리, 인증 처리 개발. MSA에서의 서비스간 중점이 되는 **API Gateway** 를 도입/기존 서비스간 연결 일치화
 - **MongoDB** 를 통한 **API** 통계를 통한 팀내 리포트를 통한 지속적 개선. **API Gateway** 를 통한 endpoint 관리, 인증, 인가 처리 중앙화
- **Planner** 개발
 - **REST API** 연동을 통한 yaml 스크립트 순서대로 API 연결 시스템 (apache airflow와 함께 python 스크립트 이용)
- **Asset** 개발
 - IDC 자산 (서버, 네트워크 장비, 스토리지)가 어떤식으로 관리되고 있는지 파악하고 현황 관리를 진행, 관리되지 않던 자산에 대한 이력 추적 및 관리 기능 개발

프로토타이핑

- 프로젝트 G 개발
 - **Java Netty** , Akka를 이용한 네트워크 엔진 구축
- 프로젝트 MS 개발
 - .NET CORE 1.2 및 Django를 이용한 서버 구현

- 토이 스튜디오 | 선임


 2016-05~2017-07

 기술 스택

Java 8 , MySQL , C# ,
Python , Django , C++ ,
Netty , Akka

LEVEL 999

레벨구구구 | CTO

 2015-01~2016-01

 기술 스택

Unity3D , C# , Ruby on
Rails , Ruby , MongoDB



마이에트 엔터테 인먼트 | 선임

우파루 사가 개발 및 운영

- 캐주얼 디펜스 게임 우파루 사가 유지 보수
- XLO라는 C++ NHN 자체 모듈 기반

밤을 걷는 선비 | Action RPG | CTO

- 기술팀 관리
 - 회사 전체 인원 25명
 - 프로그래밍 리드. 아트, 기획팀과 협업
 - 빌딩 → 소프트런칭, 런칭, 상용화까지 진행
- 클라이언트/서버 개발/공용 툴 개발 - 코드네임 **GundamBase**
- 로그 시스템 및 GM툴 구현
 - 운영에 필요한 지표 및 기록에 대해 MongoDB 적용
 - 웹 콘텐츠 서버와 동일한 프레임워크를 사용해 운영 및 기능 추가에 용이해 생산성 향상
- 프로젝트 지표
 - 플레이스토어 최고 매출 89위 (MCU : 25,000, DAU : 235,000)

건즈2 | TPS | 서버

- **AWS** 기반의 글로벌 서비스 구축 및 운용
 - 자체 제작 Agent (코드네임 **GConsole**), 명령 프로그램을 통한 원격 쉘 기반 운용
 - AWS CLI와 연동해 서비스 구축/알람 연동/Alive Check를 처리. VM 기반 위에 고성능 서비스 구축 및 운용

17 2012-06~2014-10

기술 스택

C++ , Ruby , IOCP , SQL
Server , ODBC , AWS

- Ruby (스크립트) + AWS CLI (서비스 운용 명령)
+ Keeper (C++ Windows Service Agent)
- 상시 테스트 (스트레스 테스트, 유닛 테스트) 도입, 로그 시스템 도입 및 로그 기반 개선
 - 안정성 및 임계치 확보.
 - 계측 결과에 따른 스레드 디자인 개선 및 병목 지점 해결
 - 베이스라인 (3ms) 이하로 패킷 처리 속도를 규정짓고, 그에 따른 최적화를 수행함으로써 서비스 트래픽 및 서비스 반응성에 지장 없는 코어 라이브러리 개선
- 프로젝트 지표
 - 글로벌 스팀 상용화 - MCU : 25,000, DAU : 270,000
 - 한국 CBT 1~3, 유럽 CBT 1~8, OBT, 대만 CBT 1~3, OBT, 상용화



엔트리브 소프트 | 대리

17 2009-12~2012-04

기술 스택

C++ , IOCP , SQL Server ,
ODBC

팡야 | 골프 | 서버

- 서버 관리 시스템 구현 (코드 네임 서브 라이트)
 - Windows Service Agent를 통한 원격 명령 시스템 구현
 - 서버 운영 및 모니터링 시스템 구현 및 관리
- 트러블 슈트 장애 원인에 대한 근본적인 개선
 - 병목 포인트, 크래시 지점, 크래시 원인 파악 등을 통한 서버 다운 이슈 제거
 - 서버 안정성 개선, 퍼포먼스 향상
- 네트워크 라이브러리 리뉴얼 (코드네임 레드문)
 - 스레드 디자인, lock 모델에 문제가 있는 부분을 계측 및 개선, 패킷 receive 부터 queueing, processing 까지 전반적인 개선 작업이었으므로 코드 네임을 짓고 리뉴얼


- 네트워크 패킷 처리시의 lock 모델을 변경하여 안정성 강화, 패킷 처리 속도를 개선하여 성능 개선 (최대 35000ms까지 걸리던 블러킹 타임을 50ms 이내로만 발생하게끔 개선)
 - 실수로 과도하게 크게 잡은 lock 을 감지할 수 있게 Wrapping 하였으며, 이를 기반으로 상시 테스트를 통한 병목 지점 파악 및 개선)


- 프로젝트 지표

- 태국 서비스 - MCU 60,000, DAU 50만
- 일본 서비스 - MCU 28,000, DAU : 15만
- 한국 서비스 - MCU 4,000, DAU : 2만
- 글로벌 서비스 - MCU : 12,000, DAU : 20만



네오엑트 - 서버 팀 | 책임

 2006-03~2009-11

 기술 스택

C++ , IOCP , SQL Server , ODBC

포키포키 | 소셜 네트워크 | 서버

- 서버 당 동접 1000~1500을 소화하던 서버를 장비당 3300명선까지 끌어올림 (단일 서버 기준 유저 3300명 외에도 NPC 2000마리를 동작하게끔 개선)
- 패킷 처리 구조, DB 처리 응답 구조 개선
 - 패킷 스레드에서 바로 로직을 처리하는 구조에서, 메시지 큐를 두고 큐에서 처리하도록 변경 (네트워크 스레드 효율 증가)
 - 스레드별로 위임 가능한 동작 (NPC AI 판단, NPC 동작, 오브젝트의 주기적 동작 등)을 위임하고, 다시 메시지 큐에 넣음으로써 가능한 동작들은 Pararell화 했음
- 서버 관리 시스템 구현 (코드네임 PMS)
- 프로젝트 지표
 - 일본 서비스 - MCU : 2,000, DAU : 18,000
 - 대만 서비스 - MCU : 5,000, DAU : 15,000
 - 한국 서비스 - MCU : 3,800, DAU : 11,000

카오스 온라인 | AOS | 서버

- 네트워크 모듈 개발 (TCP 네트워크 모듈, Reliable UDP 모듈 개발)

주요 업무 경험

웹&플랫폼 백엔드를 경험하고자 한 이유

- C++을 오래 써오던 2012년 어느 날, CTO님께서 Ruby on rails로 만든 Dev Toy를 들고 오셨다.
 - 간단한 로그 시각화였는데, 그 생산성에 대해서 크게 강조하셨고, 계속 써보라고 독려했다.
 - 결국 Ruby on rails와 함께 ruby를 AWS 서버 관리 자동화 스크립트로 사용하게 됐다.
 - 이후 스타트업에서 웹 서버와 Unity3D를 통한 수집형 RPG를 만들게 됐는데, 이 때에도 사용하게 되면서 웹 서버에 대한 저항감이 적어졌다.
- 이후 NHN에 입사하게 됐고, NHN은 네이버에서 분사한 회사로 자바의 회사였다.
 - 당연히 웹과 플랫폼 모두 자바였으며, 자바와 스프링을 접하기 좋은 환경이 됐다.
 - 이후 C# ASP.NET CORE, Kotlin Spring, Python Django, Node.js, Nest.js 등 다양한 웹 프레임워크를 잘 활용하는 방식에 대한 인사이트와 이해도를 가지는 것이 가져다 주는 즐거움을 가지고, 이러한 방향 즐기게 됐고 많은 웹 개발자들이 가지는 인사이트와, 로우 레벨을 베이스로 한 소켓 서버 개발자이자 게임 서버 개발자가 가지는 인사이트를 잘 활용하는 개발자를 지향하게 됐다.

클라이언트 경험

- 게임 서버 개발자가 되기 이전, 게임 클라이언트 개발자를 지망했고 싸이미디어라는 회사에서 클라이언트 개발자로 근무함
 - 아르바이트 → 수습 → 정규직 전환되어 8개월 동안 근무하던 중, 회사와 게임이 매각되면서 퇴사
 - 이후 네오엑트로 면접을 보러 갔었고 해당 면접에서 게임 서버 전직 제안을 받은 후 게임 서버 개발자로 커리어를 이어감
 - 게임 서버 개발자로서의 키웠지만, 마음 한켠에 클라이언트 개발에 대한 아쉬움이 있었음

- 그래서 여가 시간을 활용해 Direct-X 9.0C 등을 이용한 게임 만들기도 했고, 유니티 3D가 나온 이후에는 밤을 걷는 선비, NHN에서 만든 프로젝트 여럿, 모히또 게임즈에서 스타 시드, 크래프톤 SP2 에서 여러 게임 프로토타이핑을 통해서 클라이언트 개발을 이어갔음
- 클라이언트 경험을 노력을 들여 병행한 이유
 - 서버만으로는 게임이 완성되지 못한 이유도 있고, 만들고 싶었던 게임을 만들기 어려웠음
 - 또한 함께 협업하는 티어에 대한 이해도가 있어야 좋은 게임을, 좋은 제품을 만들 수 있기 때문이기도 했다

관리자로 일할 때

개발 문화 정비

- 제품의 상황
 - 이전 구성원 분들은 대략 2년 반 정도 프로젝트가 시 제품을 내놨으나, 그 이후의 제품 개선 사이클을 못 밟던 중이었다.
- 개발, 협업 문화를 정비
 - 구두 일일 스크럼에 의존하던 업무 → 모든 기록은 Notion으로 하고, Notion에 기록된 일정은 버퍼가 있을 순 있어도 비어있거나 계획이 없으면 안되도록 가져갔다
 - 기록 기반의 소통
 - 기존의 구두와 메신저에 의존하던 소통에서, 모든 업무 경험은 기록을 지향했다
 - 구현 우선 순위 비중을 높이기
 - 합리적 결론이 나기 전에는 어떠한 작업도 시도하지 않던 팀 분위기가 존재했다
 - 그렇다보니 매일 같이 논의와 협의를 시도하지만, 결론에 도달하지 못하면 어떠한 구현이나 시도가 이뤄지지 않았다
 - 빠른 시도와 회고를 지향하고, 빠른 시도를 위해 유의미한 방향을 찾을 때 까지 엔지니어링 퀄리티보다 구현 속도에 치중하기로 합의를 하고 진행했다.
 - 일정의 단위를 결과물을 내는 단위로 규정
 - 프로토타입 혹은 R&D 주제라 할 지라도 그 과정의 결과물을 기록하고 공유하도록 변경했다.
 - 무작정 늘어지거나, 문제 해결하지 못해서 붙잡고 있는 일, 혹은 엔지니어로써 찝찝하다는 이슈라는 이유로 매몰되는 시간의 한계를 두기 위함이었

다.

성과 지향, 속도전, 안정성, 팀웍이나 구성원 성장에 대한 균형

- 여러 환경에서 일해온 과정에서의 결론은, 하나의 가치관으로 일을 진행할 수 없다는 것을 알게 됐다.
- 상황에 따라 때론 성과 지향적, 때론 속도전을, 때론 팀워크, 성장 등 중요한 가치가 변한다는 것을 느꼈다.
 - 그래서 임원분들이나 동료들과의 소통을 통해서 어떠한 가치를 더 높이고, 대신 다른 가치도 너무 무너뜨리지 않는 균형 잡힌 판단과 설득이 중요하다고 생각한다.
- 솔선수범
 - 리더는 모범이 되어야하고, 신뢰를 얻어야만 동료나 부하 직원의 의구심이 가질 확률을 크게 낮출 수 있다고 생각한다.
 - 가능한 상황이라면 실무의 비율을 일정 수준 가져가면서 보여주고, 온도를 맞춰가면서 존경을 얻고 이를 바탕으로 위임과 가이드를 통한 팀웍이 갖춰져야 궁극적으로 좋은 성과를 낼 수 있다고 생각한다.

트러블 슈트

덤프 분석

- C++ 기반의 Raw 소켓 서버를 쓰는 서비스로 합류
 - C++을 실무에서 안쓴지 8년여 된 시점
- 멀티 스레드 로직 서버다 보니 다양한 이슈가 발생
 - C++ 이라는 언어적 특성도 있었지만, 견고하지 않은 코드 스택 위에 쌓아 올려져 수 많은 복잡도 높은 장애를 만들 수 있는 잠재적 결함이 있었다.
 - 실패를 컴파일 레벨과, 초기 가동에서 감지를 할 수 있는 유틸리티 코드와 빌드 후 런타임 테스트 사이클을 통해서 감지 할 수 있게 개선 했다
- 각종 장애 발생시 풀 덤프를 기반으로 다양한 장애 대응
 - Heap Corruption
 - Memory Leak
 - Dangling Pointer
 - Stack Unwinding Protect
 - Stack Reference Detect

배포 플랫폼 장애 - gitlab community를 내부적으로 사용하는, 배포 플랫폼 작업

- 사내 서버를 수십~수백대를 자동화 배포를 지원하는 도구
 - 이 과정에서 특정 몇몇 서버에서만 실패가 나는 것을 확인
 - 아쉽게도 이 과정은 플랫폼을 사용하는 서비스 담당자께서, 몇몇 서버가 패치가 안되었다는 것을 30분 이상 지난 후에 확인하고 요청주셨음.
 - 점검중인 만큼 시간이 한정되어있었는데, 이로 인해 점검 연장
 - 해당 몇몇개 서버의 공통 분모를 찾지 못했음
 - 실패 로그도 없음
 - 정확히는 시도 로그도 제대로 잡히지 않음
 - 메시지가 전달 안된 것
 - 메시징으로 사용된 HTTP 호출 로그가 당시엔 집계 되지 않고 있었음
 - 이벤트 로그, 감사 (Audit) 로그가 없었던 것
 - 그래서 Spring Log를 켜서 남겨보았으나, 웹 호출이 들어오지 않음
 - 처음부터 안들어온 것은 아니고, 해당 점검 날짜부터 해당 서버들로 메시지가 전달 안됨
 - endpoint 정보의 변경 이력을 살펴봄
 - 정상적으로 동작 한 날 이후 없었음
 - 배포 서비스의 코드 변경 이력 확인
 - 변경 없었음
 - 다른 메시징 기능도 오류가 있는지 확인
 - 정상
 - 이 과정에서 다른 것은 IP와 도메인 (내부 도메인)
 - 도메인이 아닌 IP로 endpoint 정보를 바꾸고 배포 시도를 하니 정상 동작
 - 네트워크 플랫폼쪽에서 DNS정보가 일부 날라가서, DNS를 복구했는데 이 과정에서 누락 정보가 발생했던 것
 - 이런 경우는 원래 찾기 어렵긴하다.

- 하지만, 기본적인 heartbeat 기능이 동작하고, 연결되어있는 서비스간 실시간 모니터링이 됐었으면 이런 이슈가 발생하지 않았을 것
- 또한 로그도 부족했다.
 - 언제든지 이벤트 로그나 감사 로그를 수집해, 디버깅에 활용할 수 있어야 한다.
 - 상시 기록이 어렵다면 최소한 오류가 발생한 시점 이후에라도 빠른 확인이 되게끔 기록 되어야 한다
 - 또한 웹 호출 실패 로그도 부족했다.
 - 웹 호출이 실패가 나고, 해당 HttpStatus 코드가 잘 기록되고 확인되었다면 연결 이슈라는 것과 문제가 있다는 것을 좀 더 빠르게 캐치 할 수 있다.

스타트업 경험

- 지인이 창업했던 회사에서 CTO가 퇴사하고, 새로운 CTO를 찾는다고 연락을 해주었다.
 - 대화를 해보니 1년 정도의 시간의 자금이 남아있었고, 이 안에 런칭한 게임이 성과를 내야 함을 알 수 있었다.
- 합류하고 나서 내가 했던 일은, 이전 해왔던 시간이 조금 걸리더라도 튼튼하게 어플리케이션을 만드는 일이 아니라 빠르게 기능을 구현하고 방향을 확인 한 뒤, 남은 시간 내에 최선의 결정을 하는 일이라는 것을 알게 됐다.
 - 때론 속도전이 (투자를 받는 단계 혹은, 제품의 방향성 검증 단계 등)
 - 때론 디테일이 (서비스가 궤도에 올라와서 안정적 서비스가 필요할 때)
 - 때론 시도가 (제품은 완성 됐는데, 반전이 필요하거나 트래픽 상승이 필요할 때)
- 그리고 이 때의 경험이 이후 회사 생활을 하는 데에 큰 영향을 주었는데, 바로 균형이고 상황에 맞는 판단을 하는 마인드다.



프로젝트 매니저, 테크 리더

- 일정 관리, 업무 프로세스, 협업 문화 구성
- 다수의 면접관, 신입 사원 교육, 주니어 케어 경험을 통한 개발자 육성, 팀 구성
- 기술 결정 및 기술 선행 검토, 도입 가이드 라인 제시, 트러블 슈트




대용량 처리

- 넷마블 인프라 자동화 시스템 API Gateway 개발. 자체 수집한 Audit Log (Full Log) 취합 및 관리
- 소켓 서버 기반의 성능 측정 도구를 구현 및 개선
 - 잠금 객체 이슈, 메시지 큐 병목 해결, 로직 비동기 처리, 분산 처리와 같은 방법으로 퍼포먼스 튜닝
 - DB 로그, 서버 어플리케이션 로그, OS 퍼포먼스 로그를 취합하여 알림 및 개선
- DB 쿼리 튜닝 및 개선
 - Slow Query 추출, DB 호출 구조 개선 (캐시 서버 도입, 인메모리 캐시, 디비 저장 간격 조절 등), 용도 부하별로 호출 스레드 분산

상용화

- 포키포키 CBT~ 상용화  2006-03~2009-11
- 건즈2 CBT~상용화  2012-06~2014-10
- 밤을 걷는 선비 초기 개발~상용화  2015-01~2016-02

레거시 시스템 개선

- 포키포키 ( 2006~2009), 팡야 ( 2009~2012, 5년~8년차 서비스), 테라 ( 2020~2022, 9년~12년차 서비스) 등 다양한 레거시 시스템을 운용
 - 운용 과정에서 필요한 부분을 덧데기 보다는, 근본적인 원인을 파악하고 개선하는 작업 진행
 - C++ 소켓 서버 운용 시에는 덤프가 남지 않는 크래시, 메모리 뒹어쓰기, 메모리 릭, 수십초 단위의 큰 지연 같은 이슈가 많이 생겼고, 이를 해결

높은 인프라 이해도

- 인프라실 내부 개발 조직 (넷마블)에 속해있으면서, DevOps 업무를 통한 이해도 향상

다양한 언어/ 환경 실무 적용

- 넷마블 자체가 `Private Cloud`, `AWS`, `GCP` 를 병행 지원하는 과정에서 이해도 향상
- NHN 에서도 클라우드 플랫폼 소속으로 업무 과정에서 인프라 이해도를 높일 수 있었음
- 게임 개발팀 소속일 때도, 시스템 프로그래밍, 사내 스트레스 테스트용 머신 수십대 관리/직접 IDC 머신 관리
- `C++` (네오엑트, 팡야, 건즈2), `C#` (모히또 게임즈, 레벨구구, 테라, NHN), `Java` (NHN, 넷마블), `Python` (넷마블, NHN), `JavaScript` (넷마블, NHN)
- `Windows` (대다수 게임 프로젝트), `Linux` (넷마블, NHN)
- RDB (`MySQL`, `SQL Server`, `PostgreSQL`), NoSQL (`MongoDB`, `Redis`)

자동화

- 다양한 자동화 도구를 개발, 프로세스 개선 등을 통해 업무 효율을 줄이는 데에 관심이 많고 진행.
 - 배포 자동화
 - 테스트 자동화
 - 부하 테스트 자동화
 - 로그 쉬핑/분석 자동화
- 수동 업무에 대한 다양한 자동화

다양한 업무 분야 경험

- 웹에서는 다양한 프레임워크(`Spring Boot`, `ASP.NET`, `Django`, `Ruby on rails`)로 백엔드, `Vue.js` 를 이용한 프론트엔드를 경험.
- 게임 개발에서도 다양한 언어로 서버 (`C# DotNetty`, `SuperSocket`, `C++ IOCP`, `boost asio`), `DirectX` 와 `Unity3D` 를 이용한 클라이언트 개발 경험 보유



Dev Toy

2D MMORPG

C# , Unity3D ,
DotNetty , MongoDB ,
.NET 6



Unity 3D, C# DotNetty 로 구현한 습작 2D MMORPG

넌타겟, 넷핵, 패스파인더 등 클래식 RPG에 영향을 받아 1인 개발하고 있는 게임.

Youtube <https://www.youtube.com/watch?v=Ot17tzF75MM>

Youtube <https://www.youtube.com/watch?v=BVraOJ3WA5s>

light

C++ , IOCP , Direct-X

C++로 개발한 IOCP, OLE DB, flatbuffers 등을 이용한 게임 서버 라이브러리. (Direct-X 9.0C를 이용한 클라이언트 기능들도 포함되어 있지만, 서버 위주의 구현)

2006년부터 2017년까지 개발했던 라이브러리

Github <https://github.com/elky84/light>

Web Crawler

ASP.NET CORE 6 ,
Abot2 , C# ,
MongoDB , .NET 6

Abot2, ASP.NET 6로 구현한 웹 (인벤, 루리웹, 오늘의 유머, FM 코리아 등) 크롤러 & 알리미.

RSS Feed 알림 기능도 포함.

첫 버전은 Python, Java Script로 구현했으나, 디버깅, 배포 (Docker & K8S) 용이성 등으로 C# ASP.NET 으로 포팅

Github <https://github.com/elky84/web-crawler>

Stock Crawler

Abot2, ASP.NET 6로 구현한 네이버 주식 크롤러, 분석기

다양한 알고리즘으로 종목 선별 기능.

조건에 따른 자동 매수/매도를 통한 모의 투자 기능.

ASP.NET CORE 6 ,

Abot2 , C# ,

MongoDB , .NET 6

LOL Crawler

ASP.NET CORE 6 ,

C# , MongoDB , .NET

6

Github <https://github.com/elky84/stock-crawler>

ASP.NET 6로 구현한 LOL 게임 시작/결과 추적기

최초 버전은 Java 8, Spring Boot로 구현했었으나, 디버깅, 배포 용이성으로 C# 포팅

Github <https://github.com/elky84/lol-crawler>

Web Community

ASP.NET CORE 6 ,

C# , Vue.js ,

MongoDB , .NET 6

ASP.NET 6, MongoDB, Vue.js로 구현한 웹 커뮤니티
동적 게시판, 유튜브 등의 임베딩 가능한 에디터, 댓글, 대댓글 기능등을 구현

Github <https://github.com/elky84/WebCommunity>

2D Auto Battler

Direct-X , C++

DirectX 9.0, C++로 구현한 습작 2D Auto Battler.

Football Manager 시리즈의 영향을 받아, 전략 설정, 유닛 배치를 기반으로 동작하는 게임

Github <https://github.com/elky84/NineWars/>

Youtube <https://github.com/elky84/NineWars/>

ExcelToDotnet

C# , .NET 8

Excel 정의 데이터를 C# class, enum, json 으로 뽑아서 게임 혹은 각종 앱에 편하게 쓸 수 있게 해주는 패키지

Github <https://github.com/elky84/ExcelToDotnet/>

Nuget <https://www.nuget.org/packages/ExcelToDotnet/>

EzDotNetty

C# , DotNetty , .NET

8

Java의 Netty (소켓 라이브러리)를 Microsoft가 포팅한 DotNetty를 편하게 쓸 수 있게 지원해주는 패키지

Github <https://github.com/elky84/EzDotNetty/>

Nuget <https://www.nuget.org/packages/EzDotNetty/>

EzAspDotnet

C# , ASP.NET 6 ,
.NET 8

ASP.NET 에서 사용되는 각종 편의 기능을 묶어서 제공하는 지원 패키지

Github <https://github.com/elky84/EzAspDotnet/>

Nuget <https://www.nuget.org/packages/EzAspDotNet/>

EzMongoDB

C# , MongoDB , .NET
8

MongoDb를 C#에서 편하게 쓸 수 있게 지원해주는 패키지

Github <https://github.com/elky84/EzMongoDb>

Nuget <https://www.nuget.org/packages/EzMongoDB>

JsonTable

C# , .NET 8

C#, Unity3D 에서 Json 형태의 데이터를 정의한 자료구조로 로드하고, 매핑 시켜주는 패키지

Github <https://github.com/elky84/JsonTable>

Nuget <https://www.nuget.org/packages/JsonTable>

EnumExtend

C# , .NET 8

C#에서 Enum의 설명 기능을 Attribute로 추가, 이외에도 int 로의 변환, 모두 가져오기 등의 유틸리티 기능을 확장 지원해주는 패키지

Github <https://github.com/elky84/EnumExtend>

Nuget <https://www.nuget.org/packages/EnumExtend>



Web

- Java Spring , Kotlin Spring , C# ASP.NET Core , Python Django , Ruby on Rails ,
Next.js , Vue.js

Infra

DBMS

- MongoDB , MySQL , SQL Server , PostgreSQL

Deployment

- `Docker` , `Kubernetes` , `Github Actions` , `Jenkins`

OS

- `Ubuntu` , `Cent OS` , `Windows Server`

Cloud

- `Azure` , `AWS`




Game

- `C++` , `Unity3D` , `Direct-X` , `Netty`
-




Education

경기 과학 기술 대학교

 2003-03~2006-02

컴퓨터 정보 시스템과

한국 방송 통신 대학교

 2009-03~2012-02

컴퓨터 과학과
