정보 인프라 모니터링 시각화 소프트웨어 설계 및 개발

김혜영, 김지형, 최원기, 송민환, 이상신
한국전자기술연구원 자율지능 IoT 연구센터
e-mail : boramaey@keti.re.kr, jhkim8@keti.re.kr, cwk1412@keti.re.kr,
mhsong@keti.re.kr, sslee@keti.re.kr

Design and development information infrastructure monitoring visualization software

Hyeyoung Kim, Jeehyeong Kim, Wongi Choi, Minhwan Song, Sangshin Lee
Korea Electronics Technology Institutes Autonomous IoT Research Center

요 약

A problem occurring in an information infrastructure, even a minor problem, can have a catastrophic effect on the overall information infrastructure. To prevent this, information infrastructure monitoring is required, but it is difficult to monitor all objects in a large-scale information infrastructure one by one. In this paper, we propose 'Information infrastructure monitoring visualization software' considering effective data interworking between real data and monitoring software. Through this, it will be possible to quickly understand the status and problem of the information infrastructure.

1. 서론

대규모 정보 인프라에서는 일부에서 발생한 오류라도 정보 인프라 전체의 성능이나 안정성에 영향을 끼칠 수 있다. 따라서 문제 상황 발생 시 빠르게 대응하는 것이 중요하다. 그러나 정보 인프라에 존재하는 여러 객체의 상태를 일일이 확인하는 작업은 비용이 많이 들고, 전체 정보 인프라의 상태를 빠르게 파악하기도 어렵다. 이를 해결하기 위해 각 장비들의 상태 및 오류 정보를 한눈에 파악하고 점검할 수 있는 프레임워크가 필요하다.

본 논문에서는 데이터를 기반으로 3D 웹 화면을 통해 전반적인 정보 인프라 상황을 보여주는 '정보 인프라 모니터링 시각화 소프트웨어'를 설계 및 개발했다. 주기적으로 정보 인프라 데이터를 수집 및 분석하여 사용자에게 전달함으로써 사용자가 정보 인프라의 상황을 신속하게 확인할 수 있다.

2. 본론

정보 인프라의 상태를 정확하게 파악하기 위해서는 실제 데이터값과 모니터링 프로그램에 표출되는 데이터가 긴밀하게 연결되어야 한다. 이를 위해, JSON 데이터 형식과 API 서버를 활용하여 정보 인프라에 존재하는 객체들의 데이터를 관리하고 상태 정보를 파악하도록 구현했다.

2.1 Python API 서버 구현 (FastAPI, Uvicorn)

FastAPI는 Python Web Framework로 RESTfull API 서버를 구현하는 데 사용된다. 비동기 프로그래밍을 지원하며 Uvicorn ASGI 서버를 사용한다. 빠른 퍼포먼스를 지원하며 코드를 간단하게 작성할 수 있고 버그 위험이 적다는 장점이 있다.

JSON 형식으로 데이터를 주고받도록 구현하였으며, Swagger(OpenAPI)를 이용해서 간단하게 테스트할 수 있다.

2.2 Unity 소프트웨어 구현

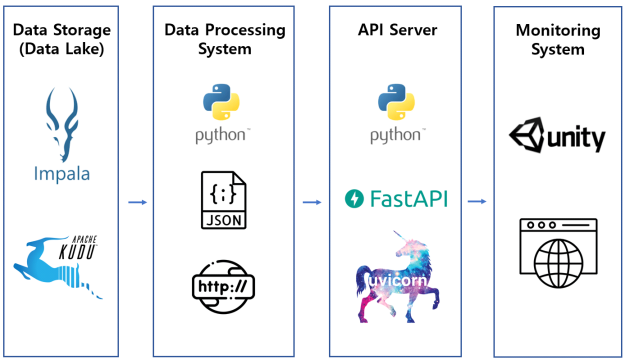
Unity란 2D 및 3D 비디오 게임의 개발 환경을 제공하는 게임 엔진이자 3D 애니메이션과 건축 시각화, 가상 현실 등 인터랙티브 콘텐츠 제작을 위한 통합 제작 도구이다. 멀티 플랫폼을 지원하기 때문

에, 다양한 운영체제에서 작동하는 프로그램을 만들 수 있으며, 모듈형이라 프로그램 수정 및 확장이 용이한 편이다.

Unity 프로그램은 일정 주기마다 데이터베이스에 데이터가 새로 업데이트되었는지 확인하고, 그 값을 화면에 띄워준다. 만약 새로 받아온 데이터에 오류 정보가 포함되어 있다면 화면에 오류 아이콘과 메시지를 띄워서 사용자가 오류 상황을 바로 파악할 수 있도록 구현했다.

2.2 정보 인프라 데이터 전송 소프트웨어 구현

데이터레이크에서 일정 주기마다 데이터를 가져와서 전송할 JSON 형식에 맞게 데이터를 가공한다. 각 장비의 상태 정보 데이터(CPU, Memory 사용량 등)와 오류 탐지 데이터(Cycle Error, Process Error 등) 등을 다룬다. 장비 종류에 해당하는 Python API 주소로 가공된 JSON 데이터를 POST Request로 전송한다. 해당 소프트웨어는 Python으로 구현되었다.



[그림 1] 소프트웨어 아키텍처

정보 인프라 데이터는 '정보 인프라 데이터 전송 소프트웨어'를 통해 일정 주기마다 API 서버로 전송된다. API 서버는 받은 데이터를 Redis 데이터베이스에 저장한다. Unity 소프트웨어는 일정 주기마다 Redis 데이터베이스에 GET Request를 날리며 새로 업데이트된 데이터가 있으면 해당 데이터를 가져와 웹 화면에 나타낸다. 장비 아이콘 위에서 마우스를 클릭하면 해당 장비 상태 정보가 적혀있는 패널을 띄우도록 구현했다. 만약 새로 받아온 데이터에 장비와 관련된 오류 데이터가 있으면 오류 아이콘을 추가로 표시하여 사용자에게 오류가 발생한 상황을 시각적으로 알려준다. 모니터링 프로그램은 웹 서버 형태로 구현하여 사용자의 편리성과 접근성을 높였다.

[그림 2] Unity 모니터링 프로그램 화면

3. 구현

Python에서 Kudu 데이터를 가져오기 위해 impala 라이브러리를 사용했다. Impala는 오픈 소스 대규모 병렬 처리 SQL 쿼리 엔진으로 Kudu에 복잡한 질의 처리를 가능하게 하기 위해 사용했다. Kudu 테이블에서 받아온 데이터를 처리하기 위해 Pandas 라이브러리를 사용했고, 데이터를 POST Request로 전송하기 위해 requests 라이브러리를 사용했다.

구현된 Unity 프로그램을 웹 서버로 구현하기 위해 WebGL과 톰캣을 사용했다. 웹 서버가 등록된 서버의 특정 포트를 외부에서 접근이 가능하도록 구성하여, IP와 포트 번호로 해당 서버에 접속할 수 있도록 구현했다.

4. 결론 및 향후 연구 방향

본 논문에서는 보다 편리한 정보 인프라 모니터링을 위해 '정보 인프라 모니터링 시각화 소프트웨어 아키텍처'를 설계하고 개발한 내용을 기술했다. 이러한 아키텍처를 이용하면 복잡하고 시간이 많이 들던 정보 인프라 관리를 시각적으로 손쉽게 할 수 있다. 이를 통해, 이상이나 오류 발생 상황에 빠르게 대응하여 더 큰 문제가 발생하는 것을 사전에 방지할 수 있을 것이다.

Acknowledgement

이 논문은 2022년도 정부(과학기술정보통신부)의 재원으로 정보통신기획평가원의 지원을 받아 수행된 연구임 (No. 2021-0-00471, 모델링 & 최적화 기반 오류-free 정보인프라 자율제어 기술 개발)

참고문헌

[1] V. Rajan, N. V. Sobhana and R. Jayakrishnan, "Machine Fault Diagnostics and Condition Monitoring Using Augmented Reality and IoT," 2018 Second International Conference on Intelligent Computing and Control Systems (ICICCS), 2018, pp. 910-914, doi: 10.1109/ICCONS.2018.8663135.

[2] P. Bansal and A. Ouda, "Study on Integration of FastAPI and Machine Learning for Continuous Authentication of Behavioral Biometrics," 2022 International Symposium on Networks, Computers and Communications (ISNCC), 2022, pp. 1-6, doi: 10.1109/ISNCC55209.2022.9851790.