

다중 큐 기반 대용량 건물 설비 데이터 처리 시스템 구조 연구

신슬비, 지영민, 권동우*

한국전자기술연구원

{seulbi0108, ym.ji, dwkwon}@keti.re.kr

A Study on the Structure of the High-Capacity Building Facility Data Processing System Based on Multiple Queues

Seulbi Shin, Youngmin Ji, Dongwoo Kwon*

Korea Electronics Technology Institute (KETI)

요약

본 연구는 대용량 데이터 서버에서 발생하는 높은 트래픽 부하를 해결하기 위한 대용량 건물 설비 데이터 처리 시스템을 제안한다. 스마트 빌딩에 대한 관심이 증가하면서 실시간으로 건물 설비 데이터를 조회하는 서비스들도 증가했다. 다수의 서비스에 실시간으로 데이터를 제공하기 위해서 최대 동시 요청 횟수 제한 등의 방법을 통해 조회 지연을 막는 방법이 사용된다. 하지만 API 요청을 위한 네트워크 자원은 계속해서 소모되기에 데이터를 조회하고 처리하는 과정에서 서버로의 요청을 조절할 필요가 있다. 요청을 보내는 모듈의 개수, 시간 간격 등을 요청 질의에 맞춰 자동으로 계산하여 트래픽 부하 평탄화를 수행하는 시스템을 설계한다. 최종적으로는 설계한 시스템을 실제 환경에 구축하여 테스트를 수행하고 성능을 평가한다.

I. 서론

스마트 빌딩 개념이 대두되면서 설비 데이터를 분석하여 건물 공간의 쾌적성을 높이고 에너지 관리를 최적화하기 위한 연구가 꾸준한 관심을 받고 있다.[1] 스마트 빌딩은 네트워크 인프라를 갖추고 시스템 확장이 가능하며 조명, 전력, 방재 등을 자동 제어해 효율적으로 건물 내 시설을 운영하여 사용자에게 쾌적한 환경을 제공하는 건축물이다.[2] 따라서 공기질, 공조기 상태 정보, 조명 밝기, 온습도 등 다수의 설비 데이터가 수집 및 활용된다. 이렇게 데이터 규모가 커지고 많아질수록 서버 응답 시간이 길어지며 특정 시간에 집중된 방대한 데이터 요청에 취약해진다.

특히 건물 설비 데이터는 실시간으로 수집, 처리, 분석되어 건물 내부 시설을 제어해야 하기에 네트워크 트래픽 부하로 데이터 조회 지연이 발생하지 않도록 해야 한다. 이를 위해 사용자별로 시간당 최대 요청 횟수를 제한하거나 최대 연결 수를 제한하는 기술이 주로 사용된다. 하지만 서버로의 전체 전송량 자체는 줄어들지 않기 때문에 네트워크 자원은 계속 소모된다. 따라서 서버로 데이터를 요청하는 시스템에서 전송량을 조절할 필요가 있다.

대용량 데이터 플랫폼은 수집된 데이터를 알맞은 형태로 처리하고 분석하는 순서로 진행된다.[3] 대용량 데이터 서버에 직접적으로 요청을 보내고 데이터를 받아오는 처리 단계에서 트래픽 부하를 방지하는 방법을 제안한다. 실시간 데이터 조회 트래픽 조절을 위해 요청 횟수와 사용자가 원하는 최신 데이터 보장 시간을 고려하여 전송량을 제어한다. 서버와 직접적으로 통신하는 모듈은 처리 단계에서 자체적으로 데이터 API 경로를 큐에 넣는 속도와 서버로 요청을 보내는 횟수를 제한하여 트래픽 제어를 수행한다. 제한하는 횟수 값은 데이터 조회에 필요한 API 질의 수와 사용자가 원하는 최소 처리 속도 값을 이용해 시스템 내부에서 자체적으로 계산한다.

본 논문에서는 이식성, 처리 속도, 트래픽 부하 평탄화를 고려한 대용량 건물 설비 데이터 처리를 위한 다중 큐 시스템을 제안한다. 2장에서는 시스템 구조를 설명하고 해당 시스템의 성능을 분석한다. 3장에서는 결론과

함께 향후 연구 방향을 제시한다.

II. 본론

1. 시스템 구조

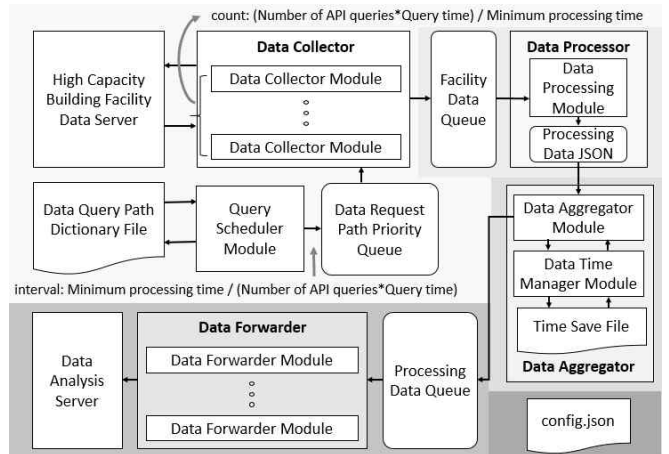


그림 1. 대용량 건물 설비 데이터 처리 시스템 구조도

그림 1은 대용량 건물 설비 데이터 처리 시스템의 구조도를 나타내고 있다. 모든 서비스는 컨테이너 내부에서 하나의 프로세스로 동작한다. 서비스 내부 기능들은 모듈로 분리하여 관리함으로써 시스템의 재사용성과 유지보수성을 높였다. 설계를 위해 사용한 데이터는 크게 장치로 구분된다. 장치는 세부적인 데이터 관계점들을 가지고 있으며, API의 파라미터로 데이터를 조회할 관계점과 시간 범위를 지정한다.

Query Scheduler Module은 스케줄러에 의해 최소 대기 시간마다 함수가 호출된다. 스케줄러를 사용하기에 다른 서비스에서 정체가 발생하지 않는다면 정확한 조회 시간 간격 관리가 가능하다. DRPP(Data Request Path Priority) Queue는 데이터가 처음 큐에 삽입된 시간을 우선순위로 가짐으로써 데이터 순서를 보장한다. 이를 통해 전송 실패한 데이터가 있

더라도 큐에 다시 삽입되면 우선순위로 인해 먼저 실행된다. Query Scheduler Module이 큐에 API Path를 넣는 시간 간격은 최소 대기 시간에 관제점 수와 요청 하나의 실행 시간을 곱한 값을 나누고 소수점 셋째 자리까지 반내림한다. 조회해 오는 시계열 데이터의 시간 범위는 설정 파일로 관리할 수 있다. 시스템은 이런 구조를 통해 일정한 간격을 두고 실시간 설비 데이터를 원하는 범위만큼 조회하고 FD(Facility Data) Queue에 삽입한다.

Data Collector는 설비 데이터를 조회한다. REST API로 데이터를 조회하고 FD Queue에 넘겨준다. 최소 처리 시간을 보장하기 위해 여러 개의 Data Collector Module이 실행된다. Module의 실행 횟수는 Data Query Path Dictionary File에 저장된 관제점 수와 요청 하나의 실행 시간을 곱한 값에 최소 대기 시간을 나누고 반올림해서 구한다. 최소 대기 시간 값이 작으면 데이터 조회 간격이 짧아져서 서버의 부하가 증가하고, 크면 최신 데이터를 반영하는 시간이 길어진다. 트래픽 부하 관리를 위해 계산 결과와는 별개로 모듈 개수 최소치, 최대치를 설정 파일로 조정할 수 있다.

Data Processor는 조회한 데이터의 전처리 등 데이터를 알맞은 형태로 처리하는 작업을 수행한다. 공조기, 복합센서, 스위치 등 다수의 IoT 기기로부터 데이터가 수집된다. 따라서 자료형이 일정하지 않을 때가 많다. int, string, list 등 관제점에서 얻은 데이터를 변환하거나 분리한다. 데이터명 요약과 숫자 데이터의 key-value pair 형태로 변환한다. 만약 숫자로 변환하지 못하는 데이터라면 제외된다. Processing Data JSON에는 관제점 명을 key로 사용한다. value는 list 형식이며 수집 시간의 timestamp, 고유 데이터명, 측정값을 key-value pair로 묶어서 list에 추가한다. 기존 list 내용은 삭제되고, 새로운 데이터로만 채워진다. 따라서 데이터 조회 요청의 시간 범위에 따라 저장되는 데이터 크기가 달라진다.

Data Aggregator는 분석 서버에서 필요로 하는 형식에 맞춰 데이터를 변환한다. 최소 대기 시간 간격으로 스케줄러를 이용해서 실행함으로써 Data Processor의 동작이 끝나고 Processing Data JSON에 새로 저장된 데이터만 읽어온다. 시계열 데이터의 중복 데이터 전송을 방지하기 위해 서버로 보낸 마지막 데이터 처리 시간을 Time Manager Module로 조회하고 검사한다. 만약 조회된 시간보다 과거 시간의 데이터라면 분석 서버로 보내지 않는다. 마지막 처리 시간 데이터는 JSON 형식으로 Time Manager Module에서 Dictionary로 관리되다가 관제점 명과 함께 최대 20초 간격을 두고 Time Save File에 저장된다. 분석 서버로 보낼 데이터는 PD(Processing Data) Queue에 삽입된다. 만약 최종 처리 완료된 데이터의 크기가 1M 이상일 경우, 1M 이하의 크기로 나눠서 삽입된다.

마지막 Data Forwarder는 데이터를 분석 서버로 전송한다. Module을 실행해서 최소 처리 시간을 보장한다. 초기에는 설정의 최소치만큼 선언되지만, PD Queue가 가득 차거나 4분의 3 이상을 차지하거나 가득 찬다면 설정된 최대치만큼 Module이 추가로 실행된다. 이처럼 상황에 따라 유연하게 동작하면서 Module의 최소 실행 요구치를 찾아낸다.

2. 트래픽 부하 테스트

시스템을 실제 환경의 설비 데이터를 대상으로 트래픽 평탄화가 되는지 테스트를 진행했다. Data Collector에서는 관제점 단위로 데이터를 읽어오지만, Data Forwarder는 처리 과정을 마친 데이터를 장치 단위로 묶어서 전송한다. 약 150개의 장치와 2,910개의 관제점을 가진 데이터를 기준으로 테스트를 진행한다. 최소 처리 시간인 60초 간격으로 7분 전부터 현재까지의 데이터를 조회하고 처리하여 서버로 전송한다.

그림 2는 Data Collector와 Data Forwarder에서 서버로 요청을 보내는 횟수를 1초 간격으로 5분간 측정된 결과이다. 큐 데이터는 해당 시간에 저

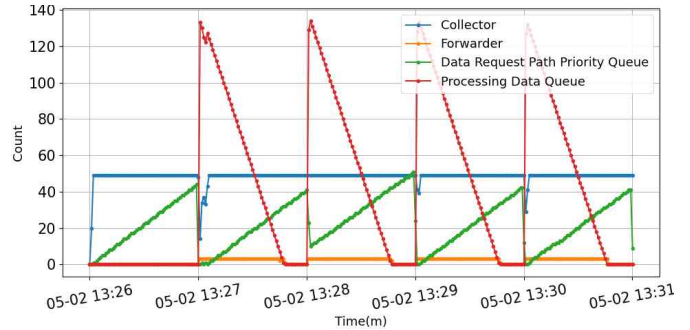


그림 2. 부하 테스트

장된 데이터 개수를 나타낸다. Data Collector는 2,910을 60으로 나눈 값을 반올림해서 최대 49개가 실행된 모습이 확인된다. 데이터가 DRPP Queue에 삽입되는 간격이 요청 횟수보다 빠르기에 큐의 데이터 개수가 조금씩 증가한다. 하지만 60초 간격으로 큐의 모든 요청 처리가 완료되는 모습이 확인된다. 후반에 0으로 급격히 떨어지는 기간이 있는 만큼 완벽한 트래픽 제어는 아니지만, 2,910개의 요청이 몰리는 구간 없이 1분간 안정적으로 분배된다.

분석 서버로 보낼 요청 개수는 사전에 파악하기 힘들기에 최대 트래픽을 Data Forwarder의 모듈 개수에 의존한다. 데이터가 누적되지 않도록 큐의 크기를 계속해서 측정하여 일정 개수 이상이면 모듈이 1개씩 증가하도록 설계되었다. Data Forwarder에 의해 1초에 최소 3개씩 서버로 전달되는 모습이 확인된다. 중간 단계의 데이터 누적, 중복 데이터 전송, 처리 지연 없이 데이터가 안정적으로 처리된다.

III. 결론

본 논문에서는 트래픽 부하 평탄화를 통해 데이터 처리가 원활하도록 대용량 건물 설비 데이터 처리를 위한 다중 큐 시스템을 설계하고 성능을 분석했다. JSON 설정 파일로 유연하게 동작하기에 유지보수가 용이하다. 성능 평가를 통해 안정적인 트래픽 부하 관리가 가능하다는 사실을 증명했다. 다만 설비 데이터는 장소에 따라 조회 형식이 달라질 수 있기에 Data Collector 서비스 구조를 변경해 사용하는 등 추가적인 작업이 요구될 수 있다. 또한 Module의 개수로 전송량을 제어하는 방식이기에 완전한 부하 분산은 아니다. 따라서 완전한 트래픽 분산을 위해 데이터 전송 횟수를 실시간으로 파악하여 시스템 트래픽 제어에 이용할 수 있도록 하는 추가적인 연구가 필요하다.

ACKNOWLEDGMENT

본 연구는 산업통상자원부(MOTIE)와 한국에너지기술연구원 (KETEP)의 지원을 받아 수행한 연구 과제입니다. (RS-2023-00237018)

참 고 문 헌

- [1] 김철인, 안주현, and 정대수. "제로에너지빌딩을 위한 딥러닝 기반 건물 에너지 시뮬레이션 보정 모델 설계." 한국통신학회 학술대회논문집. pp. 678-679. 2019
- [2] 김형석. "IT 건설융합 스마트빌딩 기술." 한국통신학회지 (정보와통신). vol. 28. no. 5. pp. 15-20. 2011
- [3] Lee, Mi-Yeong, and Wan Choe. "빅데이터 분석을 위한 빅데이터 처리 기술 동향." Korea Information Processing Society Review. vol. 19. no. 2. pp. 20-28. 2012