

RESTful Open API 를 이용한 ESB 모니터링 시스템

양세열*, 이재만**, 이성재***, 홍민****
포스코 ICT
e-mail : planetsy@poscoict.com

Monitoring System for ESB using RESTful Open API

Seyol Yang*, Jae-man Lee**, Sung-jae Lee ***, Daniel Hong****
POSCOICT

요 약

Open API 는 기존 웹 서비스를 결합하여 새로운 서비스를 개발하는 최신의 방법이다. 본 논문에서는 RESTful 웹 서비스를 Open API 화 하여 그것을 이용한 모니터링 시스템 개발을 보인다. 본문의 개발 방법을 이용하는 경우 모니터링 시스템의 통합을 손쉽게 해결할 수 있으며, 차후 다른 시스템에 쉽게 결합시킬 수 있는 장점이 있다. 또한 축소된 호출 메시지 크기로 인해 서비스 제공 속도와 네트워크 대역 사용량에도 이점이 있다.

1. 서론

최근 웹 서비스는 하나의 서비스 개발을 넘어 기존에 개발되어 있는 서비스의 결합을 통한 새로운 가치 창출에 나서고 있다. 이러한 웹 서비스의 결합은 Mesh-up 이라는 용어로 정의되고 다양한 분야에 존재하는 서비스의 결합을 통해 새로운 서비스의 개발을 가능하게 한다.

본 논문에서는 모니터링 시스템을 관리하는 운영자의 편의를 위한 Mesh-up 웹 서비스를 통한 통합 모니터링 시스템 또는 기존 시스템에 손쉽게 모니터링 화면 추가를 가능하게 하는 Open API 를 이용한 모니터링 시스템 구축 사례를 보인다.

Open API(Open Application Programming Interface)란 누구나 사용할 수 있도록 공개된 API 를 말한다. 구글 맵과 같은 경우가 대표적인 예이다. 구글을 시작으로 네이버, 다음과 같은 국내 업체들도 하나 둘씩 Open API 를 통해 자사의 서비스를 원하는 사람이 원하는 위치에 결합시켜 사용할 수 있도록 하고 있다.

일반적으로 Open API 를 통한 서비스 제공 방법은 아래 그림과 같다.

기본적인 서비스를 제공하는 Open API 서비스 제공자들이 존재하며, 그 서비스를 엮어서 새로운 서비스를 개발하여 사용자들에게 제공하는 Mesh-up Service Web Server 가 존재한다. [1]

이러한 기술의 독립 덕분에 Open API Service 제공자는 더욱 자신의 기반 서비스 기술의 향상에 매진할 수 있으며, Mesh-up Service 개발자도 기반 기술에 대한 자세한 이해 없이 필요한 기반 서비스를 제공받을 수 있게 된다.

따라서 각 서비스 레이어 별로 자신의 서비스의 역할이 명확히 구분되어 협업이 편리해지며 각 서비스의 성숙도도 향상될 수 있다.

본 논문에서는 이러한 Open API 기법을 활용하여 모니터링 시스템을 구축한 사례를 보이며, 해당 기법의 장 단점을 분석하고 향후 추가로 연구되어야 할 내용에 대해 고찰하도록 하겠다.

2. 본론

본 시스템은 ESB(Enterprise Service Bus)시스템의 모니터링을 위한 시스템이다.

1) ESB 시스템

ESB 는 웹서비스, 라우팅, 트랜스포메이션 등의 기술을 기반으로 SOA(Service Oriented Architecture)를 지원하는 미들웨어 플랫폼이다. [2]

다양한 서비스가 ESB 라는 미들웨어를 통해 상호간에 서비스를 교환하기도 하고, 데이터를 전송하기도 하여 ESB 에 묶인 서비스간의 원활한 통신을 통해 서비스들을 더욱 효율적으로 활용할 수 있게 된다. 일반적인 ESB 의 아키텍처 구조는 아래 그림과 같다.

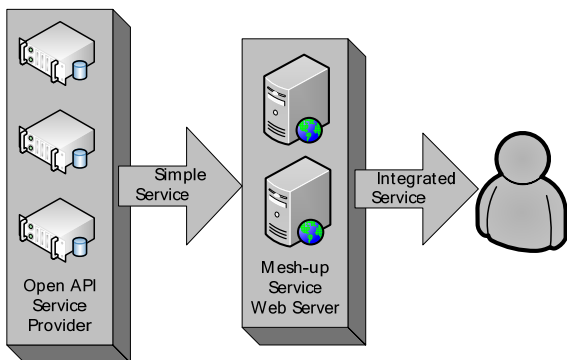


그림 1. Open API 를 이용한 서비스 제공

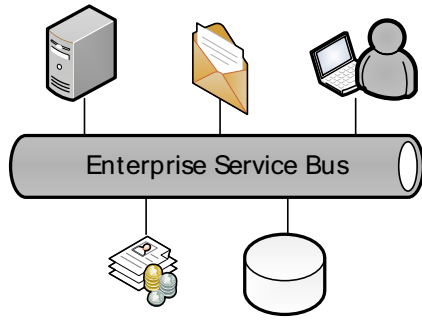


그림 2. ESB(Enterprise Service Bus)

ESB 는 기존의 통합 시스템들과 달리 표준 인터페이스를 지향하여 기존에 구성되어 있는 이종의 ESB 시스템들과의 연결과 통합을 지원한다. 따라서 회사나 부서의 합병 또는 분할 등의 이슈가 발생했을 때, ESB 시스템의 연결 또는 분리를 통해 손쉽게 기존의 시스템과 신규 시스템을 접목하거나 분리하여 조직의 정책에 맞게 시스템을 운용할 수 있게 되는 것이다.

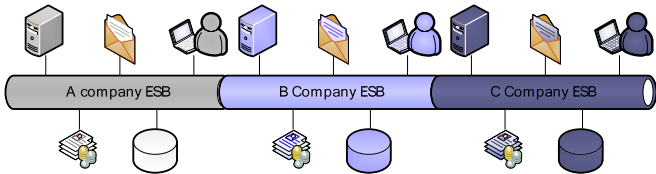


그림 3. 복합적인 ESB 시스템의 모습

2) ESB 시스템을 위한 모니터링 시스템

이와 같이 ESB 는 쉽게 연동되고 쉽게 분리될 수 있는 특성이 있다. 하지만 각 ESB 시스템 별로 개별적인 모니터링 시스템이 구축된 상황에서 ESB 시스템의 모니터링은 어려운 이슈라고 볼 수 있다. 왜냐하면 기존 ESB 시스템들에 구축되어 있는 모니터링 시스템들은 상호 호환되지 않으며 데이터의 구조 또한 상이할 가능성이 높기 때문이다.

데이터를 가져오는 부분(Data Retriever)와 데이터를 보여주는 부분(Monitoring View)이 강하게 결합되어 있어서 시스템의 잦은 통합과 분리가 발생하는 ESB 시스템에는 통합 모니터링 시스템 구축은 어려운 문제라고 볼 수 있다.

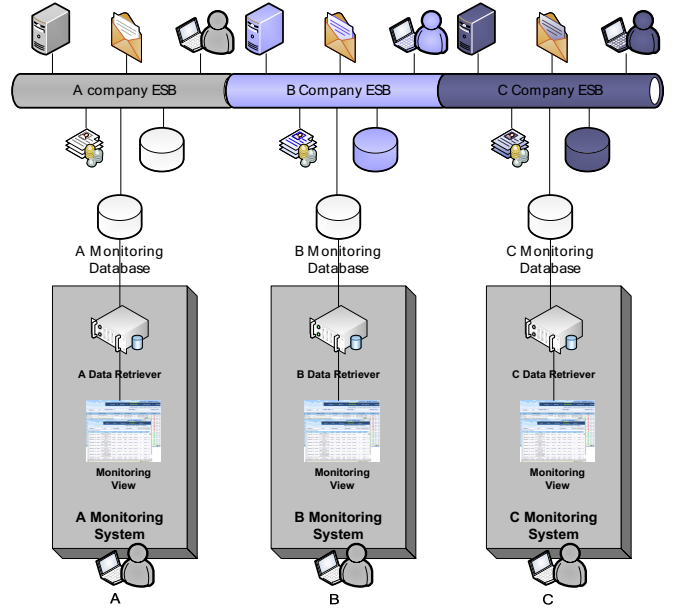


그림 4. 통합되지 않은 ESB 모니터링 시스템

결국 다수의 ESB 시스템을 모니터링 하기 위한 시스템은 위 그림과 같이 ESB 는 통합되지만 모니터링 시스템은 오히려 통합되지 못하고 개별적인 시스템과 개별적인 관리자가 필요한 경우가 발생하며 통합적인 관리의 어려움이 발생하게 된다.

3) ESB 시스템을 위한 통합 모니터링 시스템 개발

본 논문에서는 위처럼 ESB 모니터링 시스템 통합의 어려움을 개선하기 위하여 다음과 같은 모니터링 시스템 구축 사례를 제시한다. 기존처럼 모니터링 화면과 데이터 수집부분을 함께 결합시키지 않고, Database 에서 데이터를 가져와서 RESTful Open API 로 외부에 노출시켜서 쉽게 서비스와 데이터를 접근 할 수 있도록 한다.

아래와 같은 방식을 활용하면 다음과 같은 장점이 있다.

- 데이터 수집 모듈이 Open API 로 개발되어 있어서 데이터 타입에 따른 데이터 수집 모듈 개발이 필요 없어서 빠른 화면 개발이 가능
- 데이터 수집 모듈이 Open API 로 개발되어 있어서 모니터링 화면의 통합과 분리에 드는 개발 공수가 상당부분 줄어들게 됨
- 위의 Open API 는 Rest 스타일로 제작되어 호출과 응답속도를 단축할 수 있으며, 개발이 편리함[3]

Open API 로 통합된 ESB 모니터링 시스템은 아래와 같이 구성이 된다. ESB 와 마찬가지로 통합된 하나의 화면을 통해 다수의 ESB 시스템을 동시에 모니터링 할 수 있으며 신규 ESB 의 추가 발생 시와 기존 ESB 의 분리 발생 시 쉽게 화면에 추가하거나 쉽게 화면에서 제외 시킬 수 있게 된다.

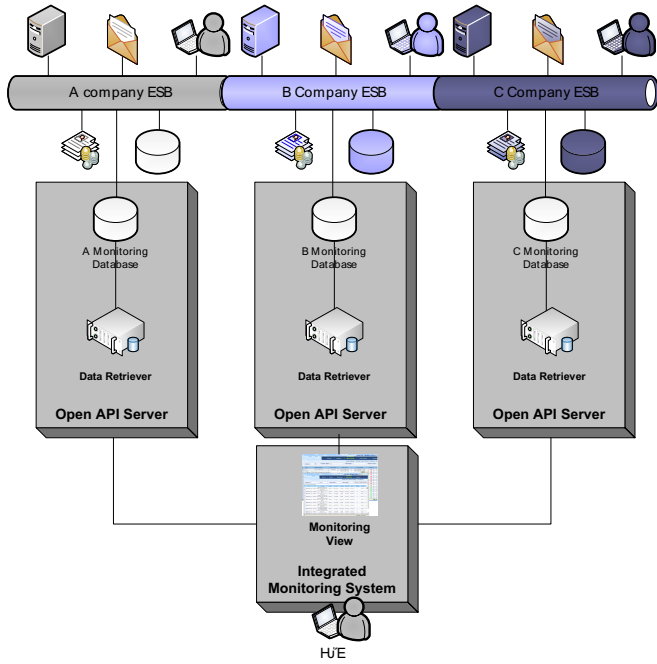


그림 5. Open API 로 통합된 ESB 모니터링 시스템

위 그림에서처럼 본 논문의 제안 기법을 활용하면 다수의 ESB 시스템을 모니터링 할 때, 쉽게 모니터링 대상 ESB 를 추가/삭제 할 수 있다. 단일 화면으로 구성되기 때문에 모니터링 관리자를 최소화 할 수 있으며 통합된 관리 시스템이 되기 때문에 시스템 오류 발생 시 오류의 원인 추적이 쉽게 가능하다.

3. 결론

본 논문에서 제안한 모니터링 시스템 아키텍처는 다수의 ESB 시스템이 결합 또는 분리되는 경우, 빠르게 통합된 모니터링 시스템을 구축하여 단일 관리자를 통한 통합된 관리 시스템 구축을 가능하게 하였다. 다수의 관리자가 통합된 ESB 시스템 중에서 각각의 ESB 를 관리하게 되면서 발생하는 커뮤니케이션의 오류와 오류 추적의 난점을 통합된 모니터링 시스템을 통해 해결할 수 있다.

해당 방법으로 ESB 라는 시스템에 한정되어 구축된 사례를 보였다. 하지만 ESB 와 같이 유사한 시스템이 자주 결합 또는 분리 되는 경우 본 논문의 제안 방식 과 같이 Open API 를 통한 서비스 제공으로 화면을 쉽게 통합하여 관리할 수 있다.

제안된 방법을 활용하게 되면 모니터링 화면을 기존의 다른 화면에 쉽게 결합하여 한 화면에 수용되는 정보의 효율성을 극대화 하는 방법도 가능하다.

차후에는 타 시스템에 해당 방식을 도입하여 어떠한 차이점이 있는지 구축을 하면서 발생한 문제에 대하여 분석하도록 하겠다.

참고문헌

- [1] 김상일, 김화성 “시맨틱 온톨로지 기반의 Open API 정보 제공 기법”, 2011.
- [2] Chappell, D. A., “Enterprise Service Bus. O’Reilly Media Inc.”, 2004.
- [3] Leonard Richardson, Sam Ruby, “Restful Web Services”, 2007.