목차

Massage Queue 3
Message Queue는 왜 사용해야 하는가? 3
애플리케이션/시스템 간의 통신 3
서버 부하가 많은 작업 3
부하분산 4
데이터 손실 방지 4
Message Queue 4
참조링크 4

Massage Queue

- Message Queue는 왜 사용해야 하는가
- Message Queue 개념 정리
- Message Queue 제품별 특징

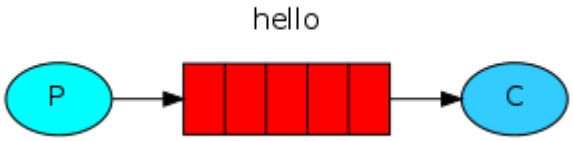
원글출처 : http://www.icelancer.com/2016/12/message-queue.html

Message Queue는 왜 사용해야 하는가?

일반적인 서버-클라이언트 구조에서는 사용자가 요청을 하면 서버는 그에 대한 처리를 한 후 사용자에게 응답을 한다. 간단한 서버 구조에서는 굳이 Message Queue(이하 MQ)를 사용할 필요가 없다. 우선 MQ를 적용하려면 RabbitMQ, Kafka, ActiveMQ등 다양한 MQ 중에서 시스템 목적에 맞는 MQ를 선정해야 하고 또 서버에 MQ를 설치해야 한다. 설치로 끝이 아니라 그 사용 방법 및 라이브러리 사용법도 익혀야 하며 MQ가 지원하는 다양한 옵션 중에 시스템 목적에 맞는 옵션을 찾아 설정해야 하고 주고 받을 메시지 구조도 정의해야 할 뿐만 아니라 다양한 전달 방식 중 시스템 목적에 맞는 방식을 선정해야 한다. 단순히 서버에서 처리하면 이런 불편함 없이 간단하게 해결할 수 있는데 왜 MQ를 사용해야 할까?

애플리케이션/시스템 간의 통신

서버 간에 데이터를 주고 받거나 어떤 작업을 요청을 할 때는 항상 시스템 장애를 염두에 두어야 한다. 서버가 갑자기 죽거나 서버 점검 등으로 다운타임이 발생하는 동안에는 요청을 보낼 수가 없다. 요청하는 서버에서 failover 처리를 해놓고 연계 시스템이 다시 살아났을 때 요청을 보내는 방법도 있지만 MQ를 이용하면 더욱 간편하게 처리할 수 있다.



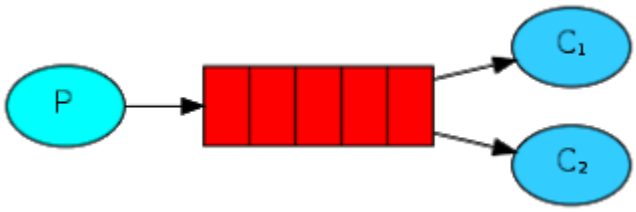
P는 C에 직접 요청하는 것이 아닌 MQ에 전달한다. 그럼 C는 MQ로 부터 요청 데이터를 수신해서 처리한다. 만약 C가 요청을 받을 수 없을 수 없는 상황이라면 해당 요청은 C가 받을 때까지 MQ에 머무르게 된다. 물론 이런 상황에서 MQ에 다운타임이 발생하면 무용지물이 되어버리겠지만, 많은 MQ가 고가용성을 위해 클러스터링 등을 지원한다.

서버 부하가 많은 작업

이미지 처리, 비디오 인코딩, 대용량 데이터 처리와 같은 작업은 메모리와 CPU를 많이 사용한다. 이러한 작업은 동시에 처리할 수 있는 양이 상당히 한정적이어서 필요하다고 무작정 요청을 처리할 수는 없다. 이 때에도 MQ를 사용하면 편리한데 처리해야할 작업을 MQ에 넣어두고 서버는 자신이 동시에 처리할 수 있는 양에 따라 하나의 작업이 끝나면 다음에 처리할 작업을 MQ에서 가져와 처리하면 된다.

부하분산

MQ를 통해 부하분산 처리도 가능하다. 지금까지 설명은 하나의 서버에 대해서만 설명했다.



그림처럼 여러 대의 서버가 하나의 큐를 바라보도록 구성하면 처리할 데이터가 많아져도 각 서버는 자신의 처리량에 맞게 태스크를 가져와 처리할 수 있다. 이러한 구조는 horizontal scaling에 유리하다.

데이터 손실 방지

MQ를 사용하지 않는다면 외부에서 받은 요청을 메모리에 저장했다가 들어온 순서대로 처리하게 할 수도 있다. 하지만 어떠한 이유로 서버가 다운되어 버리면 메모리에 쌓아둔 요청은 모두 없어지고 만다. MQ를 사용하면 이를 방지할 수 있는데 MQ로부터 가져온 태스크를 일정 시간이 지나도록 처리했다고 다시 MQ에 알려주지 않으면 MQ는 이 태스크를 다시 큐에 넣어 다시 처리할 수 있도록 한다.

MQ를 사용할 때 얻을 수 있는 잇점은 많지만 적재적소에 사용해야 한다. 요청 결과를 즉시 응답해야할 때에는 MQ는 어울리지 않는다. 주로 요청과는 별개로 처리할 수 있는 비동기 처리에 어울린다. 또한 서버에서 간단하게 처리할 수 있는 일을 MQ를 통해 전달하면 필요없는 오버헤드될 수도 있다.

Message Queue

- Apache ActiveMQ
- Apache Kafka
- Apache Qpid
- Apache RocketMQ
- RabbitMQ
- JBoss Messaging

참조링크

- https://en.wikipedia.org/wiki/Message_queue

From:
https://atl.kr/dokuwiki/ - **AllThatLinux!**

Permanent link:
https://atl.kr/dokuwiki/doku.php/message_queue?rev=1536628965

Last update: **2018/09/11 01:22**