기업환경을 위한 대용량 전자메일 전송 시스템의 설계 및 구현

임두욱*, 박정환*, 지석진*, 신동규*, 신동일*
*세종대학교 전자계산학과
e-mail:imdw@ce.sejong.ac.kr

Design and Implementation of A Large Scale E-mail Transfer System for Business Environment

Doo-Wook Im*, Jung-Hwan Park*, Seok-Jin Jee*,
Dong-Kyoo Shin*, Dong-Il Shin*
*Dept of Computer Engineering, Sejong University

요약

인터넷을 통하여 광범위한 사람과 기업이 연결되어 있는 현대 정보시스템에서, 전자메일은 그들을 연결하는데 가장 효율적인 도구 중 한가지라는 것은 이제 더 이상 말할 필요가 없을 정도다. 전자메일은 그것을 잘 활용함으로써 생산 효율을 몇 배로 향상시킬 수 있는 반면, 능숙하게 사용하지 못할 경우 수신인과 발신인 상호간의 커뮤니케이션을 혼란시켜 생산성을 저하시킬 위험성을 내포하고 있다. 본 논문에서는 대량의 전자메일을 가입자에게 발송해야 하는 기업환경에서 대량의 메일 전송시에 메일 서버에 걸리는 부하를 3-tier 방식을 적용하여 효율적으로 관리하는 방법에 대하여 고찰하고, 이의 구현에 대하여 논의한다.

1. 서론

인터넷을 통하여 광범위한 사람과 기업이 연결되어 있는 현대 정보시스템에서, 전자메일은 그들을 연결하는데 가장 효율적인 도구 중 한가지라는 것은 이제 더 이상 말할 필요가 없을 정도다. 대부분의 웹사이트들이 회원가입제로 운영되며, 이러한 환경에서 사이트와 고객들간의 커뮤니케이션의 수단으로 전자메일이 이용된다는 것은 이미 잘 알려진 사실이다. 한 예로 최근 업체들 사이에 유행하고 있는 전자메일 뉴스레터는 고객들에게 맞춤 정보를 주면서 충성도를 높이는데 큰 효과를 발휘하고 있다. 그러나 나날이 회원 가입자의 수가 늘어나고, 서비스의 질을 향상시키기 위하여 다양한 컨텐츠의 전자메일을 보내야 하는 경우가 발생하게 되며, 또한 고객들에게 전송하는 전자메일의 양이 많아질수록 메일 전송 중에 메일 서버의 차원에서 문제가 발생할 가능성이 증가한다. 현재 UNIX나 Linux 시스템에서 대표적으로 쓰이는 sendmail이나 Windows 시스템에서의 MS-Mail, Microsoft Exchange와 같은 메일 시스템이 보편적으로 사용되고 있는데, 이러한 일반적인 환경에서 대량의 메일을 전송할 경우, 시스템 메모리의 부족과 같은 경우에 의하여 서버 자체가 멈추는 경우가 자주 발생하고 있으며, 이를 효과적으로 제어할 수 있는 수단이 요구되고 있다. 또한, 기존의 서로 다른 플랫폼을 기반으로 하는 메일 전송 시스템들 사이에서 적절히 부하를 분산시키거나 통합적인 관리를 할 수 있는 방법이 제공되고 있지 않다. 따라서 본 논문의 2장에서는 현재의 메일 시스템의 일반적인 동작구조와 이에 따른 문제점을 알아보고, 3장에서 이에 따른 개선방안을 제안한 후, 4장에서 현재까지의 구현결과를 제시하며, 5장에서는 결론과 향후의 연구방향을 제시한다.

2. 연구배경
2.1 전자메일 시스템의 구성요소

일반적인 전자메일 시스템은 <그림 1>과 같은 구조를 가진다. MUA(Message User Agent)는 전자 메일 시스템의 사용자 인터페이스로, 주로 메일 클라이언트나 메일러 라고 하는 프로그램이다. 사용자는 MUA를 사용함으로써 도착한 메시지를 읽거나 송신할 메시지를 작성할 수 있다. MTA(Message Transfer Agent)는 메시지를 송신할 경우, 작성된 메시지를 특정한 방법을 이용하여 작성자가 희망하는 장소로 메시지를 전달하며, 자신의 주소로 메일이 올 경우 메시지를 수신하는 역할을 한다. Unix계열 시스템에서 주로 사용되는 sendmail이나 Windows 시스템에서 사용되는 Microsoft Exchange가 대표적인 예이다. MTS(Message Transfer System)는 MTA가 네트워 상에서 여러 개가 서로 연계해 최종 수신자까지 메시지를 전송하는 체계를 의미한다.
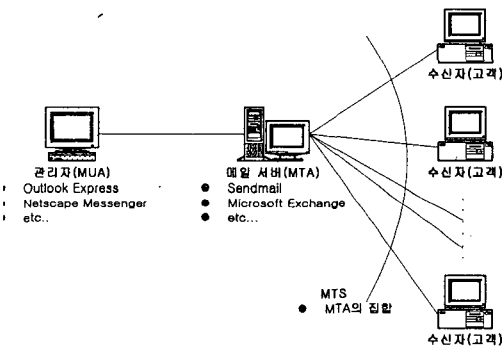

그림 1. 전자메일 시스템의 구성 요소

2.2 일반적인 전자메일 전송 시스템의 문제점

일반적인 전자메일 전송 시스템의 경우, 요즘과 같이 많은 가입자를 가지는 사이트나 메일링리스트에서의 메일전송 시에 서버에 많은 부하를 주게 된다. UNIX와 Linux와 같은 운영체제에서 일반적으로 사용되는 MTA인 sendmail을 예로 들면, 하나의 메일을 처리하기 위하여 두 개의(오래된 버전에서는 3개) 파일을 쓰고, 삭제하는 조작을 행해야 한다. 더불어 sendmail 프로그램에서는 새로운 처리를 할 경우, 새로운 서브 프로세스를 생성한다. 따라서 다수의 사용자에게 동시에 메일을 전송하게 될 경우에 MTA에서 병목현상이 일어날 수 있다. 이럴 경우에 시스템의 자원 부족 상태를 가져와 관련 프로세스가 비정상적으로 종료될 수 있으며, 최악의 경우 시스템이 다운될 수도 있다. 간혹 특정 사이트에서 중복되는 메일을 받는 경우가 있는데, 이러한 원인에 의

해 일어나는 경우가 많은 수를 차지한다. 또한, 일반적인 메일 시스템에서는 이러한 문제 발생 시에 대처할 수 있는 방법들을 제공하고 있지 않으며, 이러한 방법을 제공하기 위해서는 기존의 MTA를 직접 수정하거나, 새로운 MTA를 개발해야 하는 문제점이 있다. 이럴 경우 새로운 MTA를 개발하기 위한 시간과 비용이 소모되며, 기존의 MTA를 모두 대체해야 한다는 문제점이 발생하게된다.

3. 개발모델 제안
3.1 3-tier 모델

그림 2. 3-tier 모델

앞에서 살펴본 문제점을 본 논문에서는 3-tier 모델을 적용하여 해결하고자 한다. <그림 2>에서 보는 것처럼 3-tier 모델의 경우는, 기존의 서버와 클라이언트의 두 계층으로 나뉘어지는 클라이언트/서버 모델에서 서버와 클라이언트의 사이에 미들웨어라는 계층이 추가된 형태이다. 여기에서 미들웨어는 주로 서버 쪽의 소프트웨어나 하드웨어의 변동에 대응하거나, 서버 쪽의 자원을 안정적으로 사용하기 위하여 존재한다. 본 논문에서는 이러한 특징을 대량의 전자메일 전송 시스템에 적용하였다.

3.2 전자메일 시스템에 3-tier 모델의 적용

<그림 3.3>에서는 다음 앞에서 살펴본 3-tier 모델을 메일 전송 시스템에 적용한 결과를 보여주고 있다. 기존의 메일 서버와 관리자 사이에 메일 서버의 부하를 체크하여 분산시켜주며 시스템에 문제 발생시 이에 대한 기록을 남기는 역할을 하는 메일서버 부하 관리모듈이 위치하게 된다.
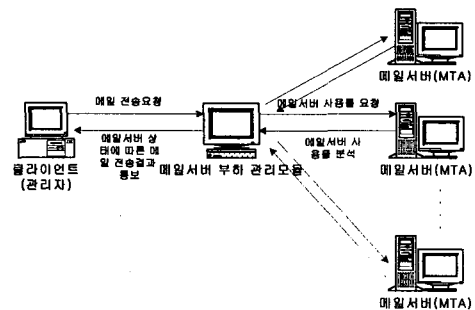

그림 3. 3-tier 모델을 적용한 메일 시스템

4. 구현결과

앞에서 설명한 3-tier 모델을 적용한 시스템에서의 미들웨어의 동작은 다음의 <그림 4>와 같다. 미들웨어가 제공하는 인터페이스를 통하여 메일 수신자의 상태가 저장되어 있는 파일이나 DB의 정보를 메일 발신자로부터 입력받게 되면, 실제 메일 발송을 담당하게될 MTA가 있는 시스템에 접근하여 이 시스템에 이미 설치되어있는 상태체크 모듈로부터 상태를 체크한 후 메일을 전송하게 된다. 이 동작을 정리하면 다음과 같다.

① 미들웨어가 제공하는 인터페이스를 통하여 메일 발송에 관계된 정보와 메일 전송명령 수신.
② MTA가 위치한 시스템에 설치된 시스템 상태체크모듈로 접근하여 시스템의 상태를 주고받는 세션 개설.
③ ②에서 개설한 세션을 통하여 시스템의 상태를 체크한 후, 충분한 자원이 있는지의 여부를 판단한 후, 메일 전송 시작
④ ③의 과정을 반복하면서 시스템이 메일을 전송할 수 있다고 판단되는 상태일 경우 메일을 전송하며, 전송할 수 있지 않은 경우에는 특정 시간 이후까지 상태를 체크하며 작업을 대기한다. 만일 이때까지 시스템의 상태가 개선되지 않으면, 에러를 발생하며 종료한다. 또한 종료시 에러에 대한 기록과 전송 중단된 시점과 이때 발송해야 할 사용자에 대한 정보를 기록으로 남겨 서비스 재개시 사용하도록 한다.

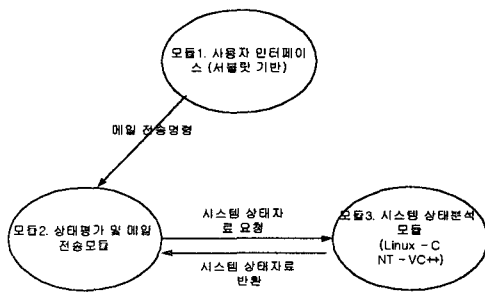이러한 구조를 가지게 됨으로써 기존의 MTA를 그대로 활용할 수 있으며, 미들웨어가 시스템의 상태를 모니터링 함으로 하여 보다 안정적인 사용이 가

그림 4. 시스템 동작도

능하게 되며, 문제가 발생할 경우 보다 효과적인 대응을 가능하게 한다.

5. 결론 및 향후연구

본 논문에서 제안된 시스템은 널리 퍼져있는 기존의 MTA를 최대한 활용하고 안정적인 서비스를 제공하고자 하는 목적을 갖고 설계되었다. 안정적인 대규모 메일서비스를 위한 솔루션은 이미 Sun Microsystems와 같은 회사에서 이미 제공하고 있으나, 이는 MTA를 새로이 구현한 것이며, 플랫폼과 MTA에 종속적인 시스템이다. 이에 반해 본 논문에서 제안한 시스템은 RFC-821(Simple Mail Transfer Protocol)과 RFC-822(Standard for the format of ARPA Internet text message)를 만족하는 MTA를 사용할 경우 플랫폼과 MTA에 독립적으로 동작하는 미들웨어의 형태로 구현된다는 장점이 있다. 향후 메일전송모듈이 메일 전송여부를 결정하는데 사용하게될 알고리즘을 다양한 사양을 갖는 시스템에서 적용 가능한 알고리즘으로 교체하는 작업이 필요하며, 이를 바탕으로 하여 현재보다 많은 사용자에게 동시 메일을 보낼 경우 발생할 수 있는 잠재적인 문제점을 파악하고 보완하는 작업이 필요할 것으로 사료된다.

참고문헌

[1] IETF, RFC-821 (August 1982), "Simple Mail Transfer Protocol."

[2] IETF, RFC-822 (August 13, 1982), "Standard for the format of ARPA Internet text message."

[3] 김기남, 신동일, 신동규, "3-tier 방식을 적용한 분산 네트웍 게임 서버의 설계", '99 춘계 학술발표논문집 1999, 04 v.6, n.1, pp.243-246

[4] 박성진, 문봉교, 손주찬, 백종명, "DCOM을 이용한 3계층 ERP 클라이언트/서버 시스템의 설계 및 구현", '99 춘계 학술발표논문집 1999, 04 v.6, n.1, pp.867-870

[5] 유정준, 서용호, 이동익, "네트워크 부하 감소를 위한 선택적 에이전트의 이동", '99 춘계 학술발표논문집 1999, 04 v.6, n.1, pp.1133-1136

[6] 이광진, 김기창, "중앙 제어식 강결합 다중 프락시 서버의 설계와 DNS 서버를 이용한 로드 분산", '99 춘계 학술발표논문집 1999, 04 v.6, n.1, pp.1009-1002

[7] 박재호, 정찬교, 강성일 "3-tier 웹서버 프로그래밍 올 가이드", 프로그램세계, 2000.3.

[8] Ted G. Lewis, "Where Is Client/Server Software Headed?", Computer, April 1995, pp.49-55

[9] Tim Quinlan, "The Second Generation of Client/Server", Database Programming & Design, May 1995, pp.31-39

[10] Eric Allman, Neil Rickert, "sendmail", O'Reilly & Associates, 1993