

멀티미디어 메일 시스템을 위한 동기화 프로토콜

최영수, 장주욱
서강대학교 전자공학과

A Synchronization Protocol for a Multimedia Mail System

Young-su Choi, Ju-wook Jang
Dept. of Electronic Engineering, Sogang Univ.

요약

본 논문에서는 인터넷 멀티미디어 메일 시스템을 위한 동기화 프로토콜을 제안한다. 제안된 동기화 프로토콜은 기존의 멀티미디어 메일 전송 프로토콜인 MIME을 확장시켜 새로운 콘텐츠 타입을 정의하고 MHEG에서 사용되는 가상축 기반 동기화 정보 기술 방식을 도입한 것으로, MIME이 지원하지 않던 메일 메시지 데이터의 동기화 정보를 기술하기 위한 방법을 제공한다. 또한 제안된 프로토콜을 이용하는 멀티미디어 메일 시스템을 구현한 결과 기존의 메일 프로토콜과 호환성을 유지하면서 메일 메시징 내 멀티미디어 데이터간의 시간-공간적 관계를 기술할 수 있기 때문에 보다 효율적인 정보의 전달이 가능함을 확인할 수 있었다.

1. 서론

범 국가적인 네트워크망의 발달과 인터넷의 확대, 보편화로 인해 인터넷을 이용한 전자우편 서비스는 이제 사람들이 네트워크 망을 통해 정보를 주고받을 수 있는 가장 손쉬운 방법으로 정착하고 있다. 초기의 인터넷 전자우편이 단순한 텍스트 위주의 메시지를 주고받았던 반면에, 점차 멀티미디어 시스템이 발달함에 따라 텍스트, 이미지, 비디오, 오디오와 같은 다양한 멀티미디어 데이터를 메일로 주고받도록 하는 멀티미디어 메일 시스템이 개발되었다. 그림 1은 인터넷을 통한 전자우편의 전송 과정을 나타낸 것이다.

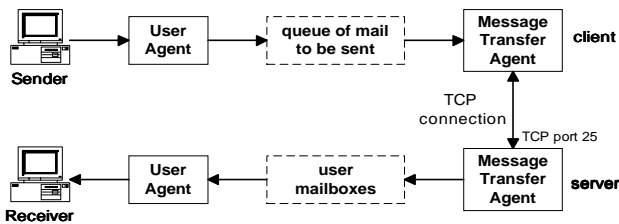


그림 1 인터넷을 이용한 전자우편의 전송

현재 멀티미디어 메일 시스템의 기반이 되는 프로토콜은 MIME (Multipurpose Internet Mail Extension)과 X.400(CCITT Recommendation no. X.400)의 두가지 프로토콜이다. 이 중 X.400은 MIME보다 먼저 멀티미디어 데이터를 고려하여 만들어진 프로토콜이기는 하지만 호환성 등의 문제로 인해 그리 널리 사용되지 않으며 현재 북미를 비롯한 몇몇 곳에서만 부분적으로 사용되고 있다. 반면 MIME은 인터넷 전자우편 전송망의 기반이 되는 SMTP(Simple Mail Transfer

Protocol)과 완벽한 호환성을 보장하기 때문에 현재 인터넷과 연결된 대부분의 멀티미디어 메일 시스템의 기반으로 널리 사용되고 있다.

그러나 MIME 프로토콜에도 몇가지 문제점이 존재한다. 이중 가장 대표적인 것은 기존의 프로토콜과의 호환성을 보장하기 위하여 몇가지 헤더만을 추가하는 방식으로 프로토콜이 만들어져 있기 때문에 데이터의 단순한 전송만이 가능할 뿐 멀티미디어 본연의 특성을 살린 효과적인 정보의 전달이 불가능하다는 점이다. 이를 보완하기 위해서는 우선 메시징 내 멀티미디어 데이터들간의 시간, 공간적 관계를 기술할 수 있도록 하는 동기화 프로토콜의 개발이 요구된다.

본 논문에서는 이렇게 멀티미디어 메일 시스템에 적용될 수 있는 동기화 프로토콜을 제안하고, 이를 직접 구현하여 그 동작을 확인하였다. 제안된 동기화 프로토콜은 MIME 프로토콜을 확장한 개념으로, 동기화 정보를 저장하기 위한 사용자 콘텐츠 타입을 정의하고 여기에서 메시징의 동기화 정보를 MHEG(Multimedia and Hypermedia Information Object Expert Group)에서 사용되는 가상축 기반 동기화 정보 기술 방식으로 기술하는 방법을 사용한다. 따라서 제안된 동기화 프로토콜은 기존 MIME 프로토콜과 호환성을 유지하면서 MIME에서는 제공되지 않던 동기화 정보를 제공할 수 있기 때문에 보다 복잡적이고 효율적인 메시징의 재생이 이루어질 수 있다.

2. 본론

1) 메일 전송 표준 프로토콜 MIME

현재 인터넷에서 사용되고 있는 멀티미디어 메일 전송 표준 프로토콜 MIME은 이전에 사용되던 텍스트 중심 메일 프로토콜 SMTP와 호환성을 유지하면서 단순히 ASCII 텍스트만을 전송할 수 있었던 인터넷 메일 메시지에 멀티미디어 데이터를 포함시킬 수 있는 표준을 제공하기 위하여 1992년 RFC-1521, 1522를 통해 제안되었다. 이를 위하여 MIME은 메일 메시지에 포함된 데이터의 타입과 데이터의 부호화 방식을 나타내기 위한 몇가지 헤더를 SMTP에 추가하는 방식을 사용하였다. 표 1은 MIME에서 사용되는 헤더의 종류와 용도를 나타내고 있다.

헤더 타입	설 명	종 류
MIME-Version	메시지에 사용되는 MIME의 버전을 나타낸다.	현재 버전 1.0을 사용
Content-Type	메시지(혹은 서버메시지)의 데이터 종류를 나타낸다.	text/multipart/message/image/audio/video/application 등
Content-Transfer-Encoding	메시지(혹은 서버메시지)에 데이터에 사용된 부호화 방식을 표시한다.	7bit/quoted-printable/base64/8bit/binary 등
Content-ID	메시지에 고유의 식별자를 부여하거나 추	
Content-Description	가로 설명이 필요할 경우 사용하는 필드.(옵션)	

표 1 MIME 헤더의 종류와 용도 설명

그러나 MIME은 완벽한 프로토콜이 아니다. MIME 프로토콜이 가지고 있는 가장 큰 문제점은 호환성 유지를 위하여 SMTP 프로토콜에 최소한의 변경만을 가하는 방식으로 프로토콜이 만들어졌기 때문에 단순히 데이터의 전송과 수신만이 가능하다는 점이다. 따라서 전달된 데이터의 통합 재생을 위한 동기화 정보를 메일 메시지에 함께 전달하는 것이 불가능하고, 결과적으로 전달된 멀티미디어 데이터가 효율적으로 사용되지 못하게 된다. 이를 보완하기 위해서는 MIME 프로토콜과 호환성을 유지하면서 메일 메시지에 동기화 정보를 함께 전송할 수 있는 방법이 필요하다.

2) MHEG을 이용한 동기화

MHEG(ISO/IEC SC29 WG12)은 서로 다른 어플리케이션이나 서비스간에 최종 표시 형태의 멀티미디어/하이퍼미디어 정보를 교환할 수 있도록 하기 위한 표준으로 만들어졌으며, 이를 위하여 각 정보 객체들을 나타내기 위한 클래스와 부호화 방법을 정의하고 있다. 다시 말해 MHEG은 서로 다른 규격이 적용되는 멀티미디어 시스템 사이에서 각 시스템의 멀티미디어 객체 표현 방법을 교환할 수 있도록 하는 표준을 클래스의 형태로 제공한다. 각 객체의 동작 특성들은 클래스에서 정의된 특성값에 의해 기술될 수 있다.

현재 MHEG은 그 개발된 용도에 따라 약간씩 다른 형태로

버전 1에서 5까지 다섯 가지의 표준이 존재한다. 본 논문에서는 그 중 클래스의 구분이 가장 잘 세분화되어 있고 각 객체의 구조가 간단하게 만들어진 MHEG-5를 사용하여 그 동기화 방식을 MIME에 도입시켰다.

멀티미디어 객체의 동작 특성을 기술하고 있는 MHEG은 각 객체간의 동기화 정보를 나타내기 위한 방법을 또한 제공하고 있다. 특히 MHEG에서는 객체간의 동기화 정보를 기술하기 위한 방법으로 가상축 기반의 동기화 정보 기술 방법을 도입하였다. 이것은 가상적인 시간-공간 좌표축을 기준으로 하여 해당 클래스의 데이터가 화면에 표시될 시간, 공간 좌표를 동기화 정보로 제공하는 방식을 말하는 것으로, 데이터를 재생하려는 측에서는 우선 정해진 좌표 공간을 생성한 다음 해당 좌표에 각 객체를 배치하여 데이터를 전송한 측과 동기화를 이룰 수 있게 된다.

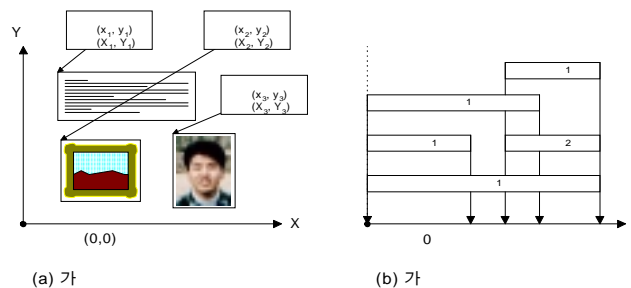


그림 2 MHEG의 가상축 기반 동기화 방식

본 논문에서는 MIME 프로토콜에 동기화 정보를 추가하기 위하여 MIME 표준에서 정의되어 있는 7가지 콘텐츠 타입 이외에 추가로 'X-MHEG-sync'라는 콘텐츠 타입을 정의하여 사용하였다. 이름에서 알 수 있듯이 이 영역에서는 메일 메시지를 구성하고 있는 각 객체들을 MHEG에서 사용하는 표기법을 사용하여 클래스로 구분한 후, 각 클래스에서 자신에 대응되는 메시지에 해당 객체의 시간-공간적 특성을 기술하는 방식을 사용하여 메일 메시지의 동기화 정보를 저장한다. 또한 MIME은 프로토콜 사용자가 자신에게 필요한 새로운 콘텐츠 타입을 정의하여 사용하면 표준 MIME 프로토콜과 호환성을 유지할 수 있도록 'X-' 혹은 'x-'로 시작되는 사용자 정의 콘텐츠 타입의 추가를 허용하므로 이를 이용하면 계속해서 MIME과의 호환성을 유지할 수 있다.

MIME에 동기화 정보를 추가하기 위하여 MHEG의 동기화 방식을 이용하는 것은 다음과 같은 근거를 가진다. 우선, 앞서 설명한 것과 같이 MHEG은 서로 다른 어플리케이션이나 서비스 사이에서 멀티미디어 데이터를 교환할 수 있도록 하기 위하여 만들어진 표준으로, 대상 서비스나 어플리케이션에 관계없이 그 최종적인 화면 표시 형태를 호환성을 위해 정의된 표준 클래스의 객체로 바꾸어 전송, 재생할 수 있도록 만들어졌다. 이렇게 광범위한 멀티미디어 데이터 정보 교환을 위해 만들어진 MHEG이므로, 메일이라는 한정된 어플리케이션에서의 멀티미디어 데이터 교환을 위한 MIME 프로토콜은 결국 MHEG에 흡수, 통합될 것으로 기대된다. 따라서 MHEG에서 사용하는 표기법과 동기화 방식을 MIME에 추가한다는 것은 합리적인 선택으로 생각된다. 또한 MIME 프로토콜에서 정의

하고 있는 표준 콘텐츠 타입과 MHEG에서 정의하고 있는 클래스들을 비교하여 보면 표 2와 같은 유사성이 나타나는 것을 볼 수 있는데, 이것은 결국 MIME에 포함되어 있는 멀티미디어 데이터의 동작 특성은 이미 MHEG의 해당 클래스에서 정의되어 있다는 것을 의미하는 것이다. 따라서 이런 의미에서도 MHEG을 이용하여 MIME에 새로운 동기화 콘텐츠 타입을 추가한다는 것은 합리적이다.

	MIME 콘텐츠 타입	MHEG 클래스
문자	Text	Text
정지 화상	Image	Bitmap
동화상	Video	Video
음성/음향	Audio	Audio

표 2 MIME 콘텐츠 타입과 MHEG 클래스의 유사성

3) MIME에서의 동기화 프로토콜

a. 공간적 동기화

MIME 메일 메시지에 포함된 멀티미디어 데이터 객체들의 동작 특성을 MHEG 클래스 표기 방식으로 기술하기 위해서는 각 데이터의 구분을 위하여 고유의 식별자를 부여할 필요가 있다. 이를 위해서는 MIME 헤더중 그 사용여부가 옵션으로 설정되어 있는 Content-ID 헤더를 이용하였다. 여기서 설정된 식별자와 X-MHEG-sync 내의 각 클래스에 포함되어 있는 식별자를 이용하여 실제 데이터와 동기화 정보간의 매핑이 가능해지는 것이다.

MHEG의 Text, Image, Video 클래스에서 데이터의 공간 동기화 특성을 나타내기 위해 사용되는 클래스 특성값은 다음과 같다.

① OriginalBoxSize

해당 객체를 둘러싸는 상자의 크기를 화면상의 좌표 시스템과 관련해 표시하며, 상자는 X축의 길이인 너비와 Y축의 길이인 높이의 두 개의 정수로 주어진다. 음수는 사용될 수 없지만 화면 좌표의 최대값을 넘는 값도 사용이 가능하며 이 경우는 화면상에 볼 수 있는 영역만 표시가 된다. Textual Notation의 경우 “:OrigBoxSize” 뒤에 두 개의 정수 X, Y값을 써서 표기가 가능하다.

② OriginalPosition

해당 객체를 둘러싸는 상자의 왼쪽 상단 모서리 위치를 나타낸다. 음수나 화면 좌표의 최대값을 넘는 값도 사용될 수 있으며 이 경우에는 화면상에 볼 수 있는 영역만 표시가 된다. 마찬가지로 두 개의 정수로 주어지고, Textual Notation에서는 “:OrigPosition” 뒤에 두 개의 정수 X, Y값을 써서 표기한다.

위의 두가지 특성값을 이용하면 기본적인 데이터의 공간 동기화 정보를 기술할 수 있는데, 이때 화면 좌표의 최대값이라고 하는 것은 메일 메시지를 보여주기 위해 사용할 창

크기를 나타내는 값을 말하며, MHEG의 Scene 클래스 특성값인 SceneCoordinateSystem을 이용해서 표기된다.

예를 들어, 각각 한 개씩의 이미지 파일과 텍스트 파일이 메일 메시지에 첨부되어 있으며 이들을 가로 300, 세로 400 크기의 창에서 보여주기를 원하는 경우의 공간 동기화 정보는 그림 3과 같이 표기될 수 있다.

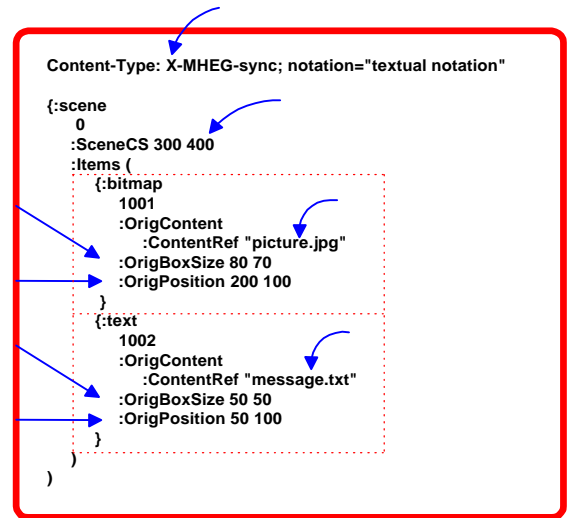


그림 3 공간 동기화 정보의 표기

b. 시간적 동기화

MHEG에서는 멀티미디어 데이터의 최종 표시 형태를 구성하는 각 객체를 구분하고 그 특성을 나타내는 표준화를 위하여 표준 클래스를 정의하고 있다. 특히 MHEG은 각 객체들의 동작(추가, 삭제, 이동, 활성화, 비활성화 등)과 사용자와의 상호작용(사용자 입력, 선택 등)을 지원하기 위하여 별도로 Action 클래스와 Link 클래스를 제공한다. 이것들은 다음과 같이 사용된다.

① Link 클래스

각 객체들의 생성에 필요한 이벤트들과 사용자와의 상호작용에 따른 판단을 조건절로 구성하여 이에 대응하는 동작들을 수행하도록 하기 위하여 사용되는 클래스이며 링크 조건과 Action 클래스로 구성된다. 즉 어떤 이벤트에 대한 동작들을 기술하고 있는 것이 Link 클래스이고, 어떤 이벤트가 발생하면 Link 클래스들의 조건과 비교하여 해당 클래스의 동작이 수행된다.

② Action 클래스

Action 클래스는 다른 클래스를 계승하지 않는 독립적인 클래스로 MHEG에서 각 객체들에 적용되는 동작들의 집합이다. Link 클래스 내에 포함되어 해당 이벤트 발생시 수행되어야 할 동작을 기술하거나, Group 클래스의 OnStartup 특성에 포함되어 프로그램이나 화면의 시작시 필요한 동작들을 기술하기 위하여 사용된다.

Link 클래스와 Action 클래스를 이용하면 어떤 이벤트가 발생하였을 때 이에 따른 각 객체의 동작을 기술하는 것이 가능하다. 시간 동기화 정보는 일정한 시간이 되면 각 객체가 어떤 식으로 동작해야 한다는 것을 나타내는 것이므로, 시간 동기화를 이루기 위해서는 일정한 시간이 되었을 때 특정한 이벤트가 발생하도록 하고 이에 해당하는 동작을 Link, Action 클래스를 이용하여 표기하면 되는 것이다. 이벤트의 발생을 위해서는 타이머가 사용될 수 있다.

즉, 메시지의 재생이 시작되면 시간 동기화 동작에 필요한 타이머들을 특정한 시간으로 설정해 놓고, 정해진 시간이 되어 각 타이머가 종료되고 타이머 종료(TimerFired) 이벤트가 발생하면 그 시간에 수행해야 할 각 객체의 동작들을 해당 타이머와 연결되어 있는 Link, Action 클래스에 기술하였다가 이것이 수행되도록 하여 시간에 따른 각 객체의 변화가 가능하도록 하는 것이다. 그림 4는 네 개의 객체가 포함되어 있는 메시지를 재생하는 경우 3개의 타이머 0, 1, 2를 사용하여 각 객체가 화면에 나타나는 시간을 제어하는 시간 동기화의 예를 나타낸 것이다.

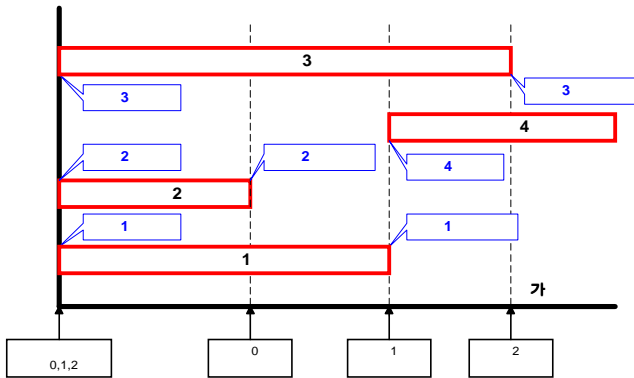


그림 4 타이머 이벤트를 이용한 시간 동기화

이를 X-MHEG-sync 영역내에서 기술하기 위해서는 다음과 같은 과정이 필요하다. 우선 가장 먼저 필요한 작업은 처음부터 화면에 재생되어 나와야 할 객체들을 메시지 재생과 동시에 활성화시키도록 하는 것이다. 이것은 Scene 클래스의 특성중 하나인 OnStartup에 각 클래스들을 활성화시키는 Action 클래스를 포함시켜 이루어질 수 있다. 타이머 값 설정도 마찬가지로 여기에서 Action 클래스중 SetTimer 동작을 통해 이루어진다.

타이머 설정이 끝나면 이후의 동작은 매우 간단하다. 정해진 시간이 되어 타이머의 종료 이벤트가 발생하면 해당 타이머의 타이머 종료 이벤트를 조건절로 가지고 있는 Link 클래스를 찾아 여기에 포함되어 있는 Action 클래스의 동작을 수행하도록 한다. 각 동작은 특정한 객체의 활성화 혹은 비활성화 동작등을 포함할 수 있기 때문에 이를 이용하면 이벤트의 시간 동기화 동작을 기술할 수 있게 된다.

예를 들어 하나의 텍스트와 두 개의 이미지 데이터가 포함되어 있는 메일에서 텍스트와 이미지 하나가 먼저 화면에 10초간 보여지다가 이후에 화면상의 이미지가 메일에 포함되어 있는 다른 메시지로 바뀌어 표시되기를 원하는 경우의 시간 동기화 정보는 그림 5와 같이 표기될 수 있다.

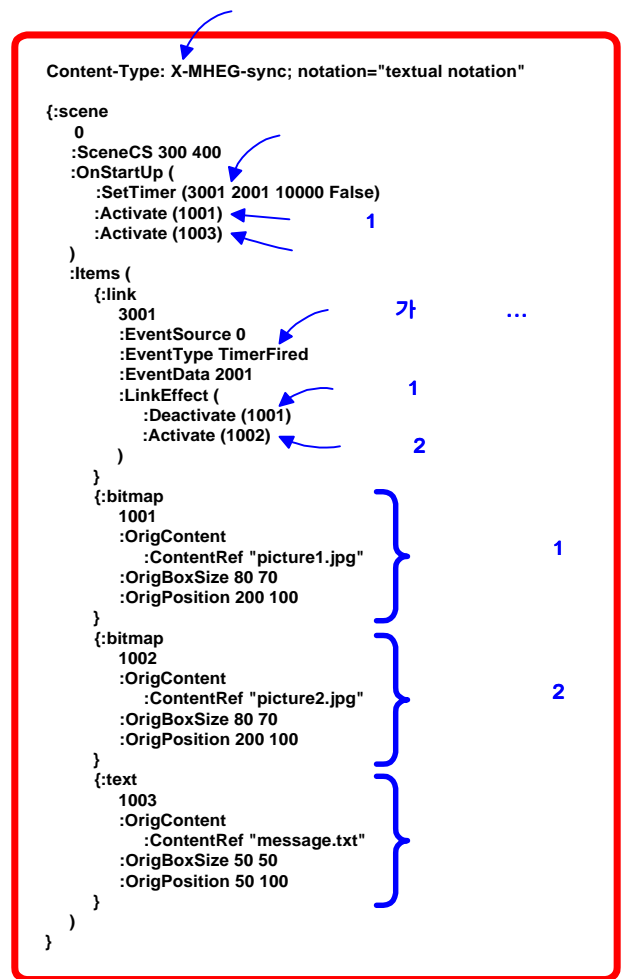


그림 5 시간 동기화 정보의 표기

4) 구현 및 결과

이상의 동기화 프로토콜을 이용하는 멀티미디어 메일 시스템을 실제로 구현하여 그 동작을 확인하였다. 전체 멀티미디어 메일 시스템은 크게 세가지 부분으로 구분될 수 있다. 우선 SMTP 모듈은 메일 표준 프로토콜 SMTP에서 설정되어 있는 규격에 따라 기본적인 텍스트 메시지의 전송을 담당한다. 여기에 MIME을 구현하기 위하여 MIME 부호화/복호화 모듈이 추가되는데 이 모듈은 텍스트만을 지원하는 SMTP 메일 전송망을 통하여 멀티미디어 데이터가 전송될 수 있도록 하기 위한 데이터의 부호화 및 복호화, 헤더의 추가등을 맡게 된다. 이상의 두가지 모듈이 기본적인 멀티미디어 메일 시스템의 구성요소라면, 동기화 프로토콜을 여기에 적용시키기 위해서는 추가적인 모듈이 필요하다. 이것은 통합 저작 도구와 뷰어로서, 전자는 MIME 표준 메시지에 입력받은 동기화 정보를 추가하는 역할을 하고 후자는 전달된 메시지에서 역으로 동기화 정보를 추출하여 일종의 프리젠테이션 시나리오를 만들고 이에 따라 메시지의 데이터들을 보여주는 역할을 하게 된다. 전체적인 시스템의 구성은 그림 6과 같다.

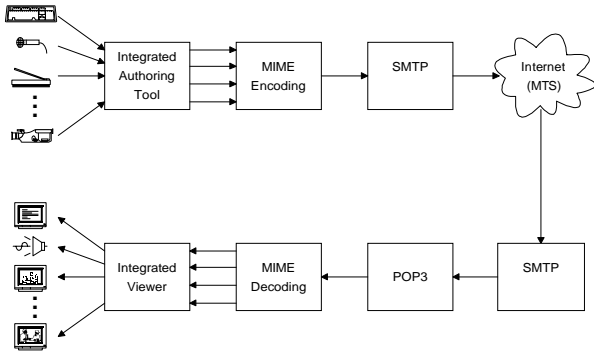


그림 6 멀티미디어 메일 시스템의 구성도

이렇게 구현된 멀티미디어 메일 시스템은 메일 저작 도구가 포함되어 있어 메일 작성을 할 수 있도록 하는 송신 프로그램과 받은 메일을 확인하는 수신 프로그램, 그리고 동기화 프로토콜이 적용된 메시지를 볼 수 있도록 해주는 뷰어 프로그램의 세가지 프로그램으로 구성된다. 그림 7은 구현된 메일 뷰어 프로그램으로 메시지를 재생시킨 예이다.

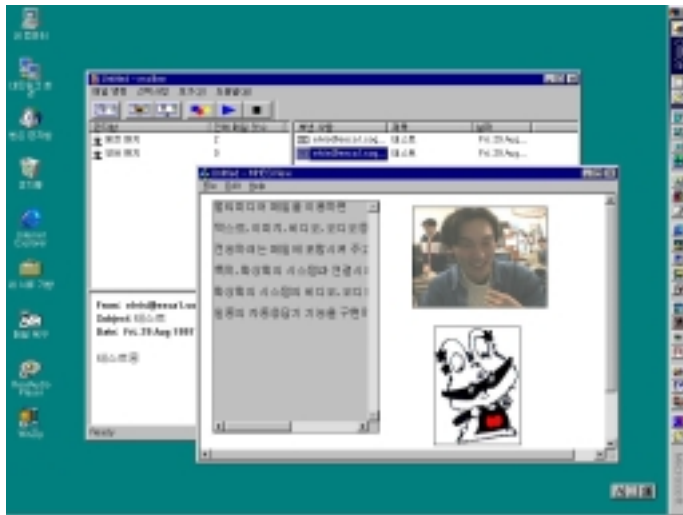


그림 7 뷰어 프로그램의 실행 화면

송신측에서 데이터의 출력 위치 및 크기, 화면에 표시되는 시간등을 지정해주면 수신측의 뷰어 프로그램은 전송된 데이터를 동기화 정보에 따라 구성하여 보여주므로 결과적으로 동기화 프로토콜을 이용하면 시간에 따라 화면에 표시되는 데이터의 위치, 크기, 종류등이 바뀌는 통합 프리젠테이션 형식의 메시지 재생을 할 수 있음을 확인할 수 있었다. 표 3은 현재 메일 전송을 위해 사용되고 있는 프로토콜 SMTP, MIME, X-400의 특징을 본 논문에서 제안한 동기화 프로토콜과 비교한 것이다.

3. 결 론

본 논문에서는 현재 사용되고 있는 인터넷 멀티미디어 메일 전송 표준 프로토콜 MIME의 단점으로 지적되는 동기화 문제를 해결하기 위한 프로토콜을 제안하였다. 또한 제안된

	SMTP	MIME	X-400	본 논문
전송 가능 데이터 타입	ASCII 텍스트만 지원	각종 텍스트 및 이미지, 비디오, 오디오 등의 멀티미디어 데이터 전송 지원	멀티미디어 데이터 전송 지원 (오디오, 비디오 등 일부 제외)	MIME과 동일
인터넷 표준 여부	표준	표준	비표준	표준 (MIME 호환)
메시지 내 데이터간의 동기화	-	지원 안함	지원 안함	시간/공간 동기화 모두 지원
동기화 방식	-	-	-	가상축 기반 동기화
기 타	초기 표준으로 멀티미디어 지원 못함	현재 인터넷 멀티미디어 메일 표준	SMTP, MIME와 호환되지 않는 비표준 프로토콜	동기화 지원으로 통합 멀티미디어 메일 재생 가능

표 3 각종 프로토콜과 동기화 프로토콜의 특성 비교

동기화 프로토콜을 이용하는 멀티미디어 메일 시스템을 실제로 구성하여 그 성능을 평가하였다. 제안된 동기화 프로토콜을 이용하여 구현된 멀티미디어 메일 시스템은 이를 사용할 경우 MIME와 완벽한 호환성을 유지하면서도 메시지내 동기화 정보를 이용한 효과적인 정보의 전달이 가능하다는 것을 보여준다.

참고문헌

- [1] Jonathan B. Postel, "Simple Mail Transfer Protocol," RFC-821, August 1982.
- [2] David H. Crocker, "Standard for the Format of ARPA Internet Text Messages," RFC-822, August 1982.
- [3] N. Borenstein and N. Freed, "MIME (Multipurpose Internet Mail Extensions) Part One: Mechanisms for specifying and Describing the Format of Internet Message Bodies," RFC-1521, September 1993.
- [4] Nathaniel S. Borenstein, "Internet Multimedia Mail with MIME: Emerging Standards for Interoperability," Proc. ULPA '92 Conference, May 1992.
- [5] Gerd Schürmann, "Multimedia mail," Multimedia Systems, vol. 4, pp. 281-295, October 1996.
- [6] Gerold Blakowski and Ralf Steinmetz, "A Media Synchronization Survey: Reference Model, Specification, and Case Studies," IEEE Journal on Selected Areas in Communications, vol. 14, no. 1, pp. 5-35, January 1996.
- [7] Thomas Meyer-Boudnik and Wolfgang Effelsberg, "MHEG Explained," IEEE Multimedia, pp. 26-38, Spring 1995.
- [8] ISO/IEC 13522-5:1997(E), "Information technology-Coding of multimedia and hypermedia information-Part 5: Support for base-level interactive applications," April 15, 1997.